ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ

ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ

ΤΟΜΕΑΣ ΣΥΣΤΗΜΑΤΩΝ ΜΕΤΑΔΟΣΗΣ ΠΛΗΡΟΦΟΡΙΑΣ

ΚΑΙ ΤΕΧΝΟΛΟΓΙΑΣ ΥΛΙΚΩΝ

Μοντέλα Μηχανικής Μάθησης για την Έγκαιρη Πρόβλεψη-Διάγνωση του Σακχαρώδους Διαβήτη Η Πορεία και οι Επιπλοκές της Νόσου

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

του

ΣΕΛΜΑΝ ΣΕΡΙΦ-ΔΑΜΑΔΟΓΛΟΥ

Επιβλέπων : Καθ. Δημήτριος-Διονύσιος Κουτσούρης
Ομότιμος Καθηγητής Ε.Μ.Π.

Συνεπιβλέπουσα : Ουρανία Πετροπούλου
ΕΔΙΠ Ε.Μ.Π.

Αθήνα, Φεβρουάριος 2025

ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ
ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ
ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ
ΤΟΜΕΑΣ ΣΥΣΤΗΜΑΤΩΝ ΜΕΤΑΔΟΣΗΣ ΠΛΗΡΟΦΟΡΙΑΣ
ΚΑΙ ΤΕΧΝΟΛΟΓΙΑΣ ΥΛΙΚΩΝ

Μοντέλα Μηχανικής Μάθησης για την Έγκαιρη Πρόβλεψη-Διάγνωση του Σακχαρώδους Διαβήτη Η Πορεία και οι Επιπλοκές της Νόσου

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

του

ΣΕΛΜΑΝ ΣΕΡΙΦ-ΔΑΜΑΔΟΓΛΟΥ

Επιβλέπων : Καθ. Δημήτριος-Διονύσιος Κουτσούρης
Ομότιμος Καθηγητής Ε.Μ.Π.
Συνεπιβλέπουσα : Ουρανία Πετροπούλου
ΕΔΙΠ Ε.Μ.Π.

Εγκρίθηκε από την τριμελή εξεταστική επιτροπή την 18η Φεβρουαρίου 2025.

(Υπογραφή)	*(Υπογραφή)*	*(Υπογραφή)*
.....	.....	.....
Καθ. Κουτσούρης Δ. Δ.	Καθ. Ματσόπουλος Γ.	Καθ. Τσανάκας Π.

Αθήνα, Φεβρουάριος 2025

(Υπογραφή)

.....

ΣΕΛΜΑΝ ΣΕΡΙΦ-ΔΑΜΑΔΟΓΛΟΥ

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

Ευχαριστίες

Η παρούσα διπλωματική εργασία εκπονήθηκε στα πλαίσια των δραστηριοτήτων του Εργαστηρίου Βιοϊατρικής Τεχνολογίας του τομέα Συστημάτων Μετάδοσης Πληροφορίας και Τεχνολογίας Υλικών της Σχολής Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών του Εθνικού Μετσόβιου Πολυτεχνείου, υπό την επίβλεψη του κύριου Δημήτριου-Διονύσιου Κουτσούρη, Ομότιμου Καθηγητή Ε.Μ.Π, τον οποίο θέλω να ευχαριστήσω θερμά για την εμπιστοσύνη που μου έδειξε και την ευκαιρία που μου έδωσε να εκπονήσω αυτή τη διπλωματική.

Επίσης, θέλω να ευχαριστήσω την κυρία Ράνια Πετροπούλου, Ε.ΔΙ.Π. Ε.Μ.Π. και τον κύριο Σταύρο Πιτόγλου, Διδάκτορα Ε.Μ.Π. για την υποστήριξη και την καθοδήγησή τους, καθώς και για τις πολύτιμες συμβουλές τους, που συνετέλεσαν καθοριστικά στην επιτυχή διεκπεραίωση της παρούσας διπλωματικής εργασίας.

Θα ήθελα να κάνω ειδική αναφορά και να ευχαριστήσω με όλη μου την καρδιά την οικογένειά μου, ιδιαιτέρως τον πατέρα μου Σιουκρή και την μητέρα μου Αϊσέ, τον αδερφό μου Μερβάν και τις αδερφές μου Χουμεϊρά και Σουμεϊρά, που στάθηκαν δίπλα μου σε κάθε μου βήμα, με ενθάρρυναν και μου έδειξαν την αξία της υπομονής και της προσπάθειας. Η αμέριστη αγάπη και η ανιδιοτελής υποστήριξή τους είναι καθοριστική στην πραγματοποίηση όλων μου των στόχων, ένας εκ των οποίων ήταν οι σπουδές μου στη Σχολή Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών του Εθνικού Μετσόβιου Πολυτεχνείου. Το δίπλωμα από το Πολυτεχνείο είναι αφιερωμένο σε αυτούς.

Και τέλος, πρέπει να αναφέρω και να ευχαριστήσω ιδιαιτέρως τους πολύ καλούς φίλους μου, που κατά τη διάρκεια των σπουδών μου, η υποστήριξή τους δεν μπορεί να εκφραστεί με λέξεις.

Σελμάν Σερίφ-Δαμάδογλου

Περίληψη

Ο σακχαρώδης διαβήτης είναι μια χρόνια και περίπλοκη ασθένεια, η οποία εμφανίζεται όταν ο οργανισμός δεν μπορεί να ελέγξει τα επίπεδα γλυκόζης στο αίμα. Υπάρχουν τρία είδη διαβήτη: ο σακχαρώδης διαβήτης τύπου 1, ο σακχαρώδης διαβήτης τύπου 2 και ο διαβήτης κύησης. Η έγκαιρη διάγνωση του σακχαρώδους διαβήτη είναι απαραίτητη για την διαχείριση της νόσου, αλλά και για την πρόληψη των σοβαρών επιπλοκών που μπορεί να δημιουργήσει (καρδιαγγειακές παθήσεις, νεφροπάθεια, τύφλωση κτλ.).

Ένα από τα πιο αποτελεσματικά και πολλά υποσχόμενα εργαλεία για την έγκαιρη διάγνωση της ασθένειας είναι η μηχανική μάθηση. Με τους κατάλληλους αλγορίθμους, η μηχανική μάθηση δύναται να διαδραματίσει βασικό ρόλο στην ανάπτυξη μοντέλων για τον γρήγορο και αποτελεσματικό χειρισμό των ιατρικών φακέλων και εξετάσεων, την πρόβλεψη αλλά και τη θεραπεία των χρόνιων νοσημάτων, όπως είναι και ο σακχαρώδης διαβήτης.

Η συγκεκριμένη διπλωματική εργασία αφορά την ανάπτυξη του θεωρητικού υπόβαθρου των αλγορίθμων μηχανικής μάθησης, τη βιβλιογραφική αναδρομή και μελέτη ερευνών που έχουν πραγματοποιηθεί και την αξιολόγηση των επιδόσεων των αλγορίθμων για τη διάγνωση του σακχαρώδη διαβήτη μέσω συνόλων δεδομένων όπως είναι η βάση δεδομένων PIMA Indians Diabetes Database.

Με την βοήθεια της γλώσσας προγραμματισμού Python και τη χρήση κατάλληλων βιβλιοθηκών μηχανικής μάθησης (scikit-learn, pandas κτλ), αναπτύχθηκε πρόγραμμα που προσομοιώνει και παράγει μετρήσεις με τη χρήση διαφόρων μοντέλων μηχανικής μάθησης, συγκρίνει τα αποτελέσματα μέσω γραφικών παραστάσεων και συμπεραίνει την αποτελεσματικότητά τους.

Λέξεις κλειδιά: Σακχαρώδης Διαβήτης, Μηχανική Μάθηση, Επιβλεπόμενη Μάθηση, Μη Επιβλεπόμενη Μάθηση, Ενισχυτική Μάθηση, Linear Regression, Logistic Regression, Decision Trees, Random Forests, SVM, k-NN, Naive Bayes, K-Means, Q-Learning, PIMA Indians, Python

Abstract

Diabetes mellitus is a chronic and complex disease that occurs when the body cannot control blood glucose levels. There are three types of diabetes: type 1 diabetes mellitus, type 2 diabetes mellitus and gestational diabetes. Early diagnosis of diabetes mellitus is essential to manage the disease and to prevent the serious complications it can cause (cardiovascular disease, kidney disease, blindness, etc.).

One of the most effective and promising tools for early diagnosis of the disease is machine learning. With appropriate algorithms, machine learning can play a key role in developing models for fast and efficient handling of medical records and tests, prediction and treatment of chronic diseases, such as diabetes mellitus.

This thesis is about developing the theoretical background of machine learning algorithms, literature review and study of researches that have been carried out and evaluation of the performance of the algorithms for diagnosis of diabetes mellitus using datasets such as PIMA Indians Diabetes Database.

With the help of Python programming language and using appropriate machine learning libraries (scikit-learn, pandas etc.), a program was developed to simulate and generate measurements using various machine learning models, compare the results through graphs and infer their effectiveness.

Keywords: Diabetes mellitus, Machine Learning, Supervised Learning, Unsupervised Learning, Reinforcement Learning, Linear Regression, Logistic Regression, Decision Trees, Random Forests, SVM, k-NN, Naive Bayes, K-Means, Q-Learning, PIMA Indians, Python

Περιεχόμενα

Ευρετήριο Εικόνων

Ευρετήριο Γραφημάτων

1. Εισαγωγή

Εκατομμύρια άτομα παγκοσμίως πάσχουν από σακχαρώδη διαβήτη, μια χρόνια και περίπλοκη ασθένεια. Η διαταραχή αυτή εμφανίζεται όταν ο οργανισμός αδυνατεί να ελέγξει επαρκώς τα επίπεδα γλυκόζης (σακχάρου) στο αίμα, γεγονός που οδηγεί σε επικίνδυνα υψηλά επίπεδα για την υγεία του ασθενούς. Τα κύτταρά μας χρησιμοποιούν τη γλυκόζη, η οποία λαμβάνεται από τις τροφές που τρώμε, ως κύρια πηγή ενέργειας. Το πάγκρεας παράγει την ορμόνη ινσουλίνη, η οποία είναι απαραίτητη για την είσοδο της γλυκόζης στα κύτταρα. Ο σακχαρώδης διαβήτης προκαλεί υπερβολικά υψηλά επίπεδα γλυκόζης στο αίμα με σημαντικές επιπτώσεις στην υγεία, επειδή είτε ο οργανισμός δεν μπορεί να χρησιμοποιήσει επαρκώς την ινσουλίνη (όπως στον τύπο 2) είτε δεν δημιουργείται σε κατάλληλες ποσότητες (όπως στον τύπο 1).

1.1 Τύποι Σακχαρώδους Διαβήτη

Ο διαβήτης τύπου 1, ο διαβήτης τύπου 2 και ο διαβήτης κύησης είναι μεταξύ των διαφόρων μορφών σακχαρώδη διαβήτη που συζητήθηκαν προηγουμένως. Παρακάτω γίνεται ανάλυση αυτών των τύπων διαβητών.

Διαβήτης Τύπου 1

Ο σακχαρώδης διαβήτης τύπου 1 είναι μια αυτοάνοση πάθηση, όπου το ανοσοποιητικό σύστημα του οργανισμού επιτίθεται και καταστρέφει τα β-κύτταρα του παγκρέατος, τα οποία είναι υπεύθυνα για την παραγωγή της ινσουλίνης. Αυτό σημαίνει ότι ο οργανισμός δεν μπορεί να παράγει την ινσουλίνη που απαιτείται για τη ρύθμιση των επιπέδων σακχάρου στο αίμα. Ο διαβήτης τύπου 1 μπορεί να αναπτυχθεί σε οποιαδήποτε ηλικία, ωστόσο συνήθως εκδηλώνεται στην παιδική ή εφηβική ηλικία. Η υπερβολική δίψα, η συχνή ούρηση, η εξάντληση, η θολή όραση και η ανεξήγητη απώλεια βάρους είναι μερικά από τα συμπτώματα [1]. Προκειμένου να διατηρηθεί ο έλεγχος της γλυκόζης, οι ασθενείς πρέπει να λαμβάνουν καθημερινά ινσουλίνη μέσω ενέσεων ή αντλιών ινσουλίνης, επειδή ο οργανισμός δεν είναι σε θέση να την παράγει [2].

Διαβήτης Τύπου 2

Ο πιο διαδεδομένος τύπος διαβήτη, ο διαβήτης τύπου 2, προκαλείται είτε από την αδυναμία του οργανισμού να χρησιμοποιήσει κατάλληλα την ινσουλίνη είτε από την αδυναμία του παγκρέατος να δημιουργήσει αρκετή ινσουλίνη για να καλύψει τις ανάγκες του οργανισμού. Αυτή η διαταραχή αναφέρεται ως «αντίσταση στην ινσουλίνη» και συνήθως συνδέεται με κακές διατροφικές συνήθειες, παχυσαρκία και αδράνεια [1]. Η βελτιωμένη διατροφή, η άσκηση και τα φάρμακα αποτελούν μέρος της διαχείρισης του διαβήτη τύπου 2. Σε πιο σοβαρές περιπτώσεις, μπορεί να είναι απαραίτητες ενέσεις ινσουλίνης [2].

Διαβήτης Κύησης

Ο διαβήτης κύησης εμφανίζεται κατά τη διάρκεια της εγκυμοσύνης και συχνά υποχωρεί μετά τη γέννα, αν και μπορεί να αυξήσει τον κίνδυνο ανάπτυξης διαβήτη τύπου 2 αργότερα στη ζωή τους, αλλά η πάθηση συνήθως υποχωρεί μετά τη γέννηση [1].

Τα υψηλά επίπεδα γλυκόζης στο αίμα προκύπτουν από την αδυναμία του οργανισμού να παράγει αρκετή ινσουλίνη για να ικανοποιήσει τις αυξανόμενες απαιτήσεις της εγκυμοσύνης [2].

1.2 Επιπλοκές του Σακχαρώδους Διαβήτη

Η ανεπαρκής φροντίδα και η μη τήρηση των οδηγιών των ιατρών για τον σακχαρώδη διαβήτη μπορεί να έχει σοβαρές συνέπειες που επηρεάζουν τη συνολική υγεία. Οι συνέπειες αυτές είναι μερικές φορές ύπουλες και αρχικά ανεπαίσθητες, ωστόσο μπορεί τελικά να οδηγήσουν σε μόνιμη βλάβη με την πάροδο του χρόνου. Οι πιθανές σοβαρές επιπλοκές περιλαμβάνουν:

- **Καρδιαγγειακές Παθήσεις**: Ο διαβήτης αυξάνει τον κίνδυνο καρδιαγγειακών διαταραχών, συμπεριλαμβανομένης της στεφανιαίας νόσου και των εγκεφαλικών επεισοδίων, μέσω της αθηροσκλήρωσης, η οποία περιλαμβάνει βλάβες στις μεγάλες και μεσαίες αρτηρίες [3].
- **Νευροπάθεια**: Τα αυξημένα επίπεδα γλυκόζης μπορεί να οδηγήσουν σε τραυματισμό των νεύρων, προκαλώντας μούδιασμα, αίσθημα καύσης ή πόνου, ιδιαίτερα στα άκρα [2].
- **Νεφροπάθεια**: Τα αυξημένα επίπεδα γλυκόζης επηρεάζουν αρνητικά τη φυσιολογική λειτουργία των νεφρών, οδηγώντας σε χρόνια νεφρική ανεπάρκεια [4].
- **Διαβητική Αμφιβληστροειδοπάθεια**: Η διαβητική αμφιβληστροειδοπάθεια επηρεάζει τα αιμοφόρα αγγεία του αμφιβληστροειδούς, οδηγώντας σε διαταραχές της όρασης και, σε ορισμένες περιπτώσεις, σε τύφλωση [5].
- **Διαβητικό Πόδι**: Η αλληλεπίδραση της μειωμένης κυκλοφορίας και της νευροπάθειας μπορεί να οδηγήσει σε έλκη και ακρωτηριασμούς [2].

2. Η Σημασία της Έγκαιρης Διάγνωσης και Διαχείρισης

Η έγκαιρη διάγνωση του σακχαρώδους διαβήτη είναι απαραίτητη για την αποτελεσματική διαχείριση της νόσου και την πρόληψη των επιπλοκών. Η διάγνωση του διαβήτη τύπου 2 μπορεί να αναβληθεί λόγω της ενίοτε ανεπαίσθητης ή δυσδιάκριτης φύσης των συμπτωμάτων του. Παρ' όλα αυτά, η έγκαιρη διάγνωση και οι τροποποιήσεις στον τρόπο ζωής, συμπεριλαμβανομένων των θρεπτικών διατροφικών πρακτικών, της συνεπούς σωματικής δραστηριότητας και των φαρμακολογικών παρεμβάσεων, μπορούν να βελτιώσουν τα επίπεδα γλυκόζης στο αίμα και να μετριάσουν τα προβλήματα [1].

Η έγκαιρη διάγνωση παρέχει στους ασθενείς την ευκαιρία να εφαρμόσουν στρατηγικές προληπτικής θεραπείας, να αξιολογούν με συνέπεια την κατάσταση της υγείας τους και να υιοθετήσουν υγιεινές πρακτικές, μειώνοντας έτσι το άγχος και την αγωνία σχετικά με τις μακροπρόθεσμες επιπτώσεις της νόσου.

2.1 Εισαγωγή

Η Μηχανική Μάθηση (Machine Learning - ML) αποτελεί έναν κλάδο της Τεχνητής Νοημοσύνης (Artificial Intelligence - AI), η οποία επικεντρώνεται στη δημιουργία μαθηματικών μοντέλων και αλγορίθμων που επιτρέπουν στους υπολογιστές να «μαθαίνουν» από τα δεδομένα και να αποδίδουν καλύτερα σε διάφορες εργασίες χωρίς να χρειάζεται να διδάσκονται κάθε σενάριο [8]. Οι υπολογιστές είναι χρήσιμοι επειδή μπορούν να μάθουν να εντοπίζουν μοτίβα, να προβλέπουν αποτελέσματα ή να κάνουν κρίσεις με βάση πληροφορίες.

Η μηχανική μάθηση διαχωρίζεται σε διακριτές κατηγορίες με βάση το είδος της μάθησης που χρησιμοποιείται και τον τύπο των δεδομένων που χρησιμοποιούνται για την εκπαίδευση του μοντέλου. Η ενισχυτική μάθηση (Reinforcement learning), η μάθηση με επίβλεψη (supervised learning) και η μάθηση χωρίς επίβλεψη (unsupervised learning) είναι οι τρεις βασικές κατηγορίες. Κάθε κατηγορία έχει μοναδικές μεθόδους και χρήσεις που την καθιστούν κατάλληλη για συγκεκριμένα ζητήματα και πληροφορίες.

2.2 Κατηγορίες Μηχανικής Μάθησης

2.2.1 Επιβλεπόμενη Μάθηση (Supervised Learning)

Η Επιβλεπόμενη Μάθηση είναι η πιο διαδεδομένη κατηγορία της Μηχανικής Μάθησης και περιλαμβάνει τη διαδικασία εκπαίδευσης ενός μοντέλου χρησιμοποιώντας δεδομένα που περιλαμβάνουν τόσο εισόδους (features) όσο και αντίστοιχες εξόδους (labels).

Το μοντέλο εκπαιδεύεται ώστε να αναγνωρίζει την συσχέτιση μεταξύ αυτών των δύο και να χρησιμοποιεί αυτή τη γνώση για να κάνει προβλέψεις ή να κατηγοριοποιήσει νέα δεδομένα. Προκειμένου να μειωθεί η απόκλιση μεταξύ των προβλέψεων του μοντέλου και των

πραγματικών ετικετών εξόδου, οι παράμετροι του μοντέλου προσαρμόζονται κατά τη φάση της εκπαίδευσης [7].

Η Επιβλεπόμενη Μάθηση χρησιμοποιείται κυρίως για προβλήματα πρόβλεψης και ταξινόμησης, τα οποία περιλαμβάνουν:

- **Παλινδρόμηση (Regression):** Σε θέματα παλινδρόμησης, το μοντέλο καλείται να προβλέψει μια αριθμητική τιμή και η έξοδος είναι συνεχής. Η πρόβλεψη της τιμής ενός ακινήτου ή της χρηματιστηριακής τιμής μιας εταιρείας με βάση έναν αριθμό μεταβλητών είναι μερικές από τις περιπτώσεις αυτών των θεμάτων. Η γραμμική παλινδρόμηση (linear regression) είναι το πιο ευρέως χρησιμοποιούμενο μοντέλο για αυτού του είδους τα ζητήματα.
- **Ταξινόμηση (Classification):** Στην ταξινόμηση, το μοντέλο προσπαθεί να ομαδοποιήσει τα δεδομένα σε προκαθορισμένες κατηγορίες και παράγει μια κατηγορική έξοδο. Για παράδειγμα, το μοντέλο μπορεί να προσπαθήσει να προσδιορίσει αν μια ιατρική εικόνα περιέχει καρκινικά κύτταρα ή αν ένα μήνυμα ηλεκτρονικού ταχυδρομείου είναι ανεπιθύμητο. Στην ταξινόμηση χρησιμοποιούνται αλγόριθμοι όπως τα δέντρα απόφασης (decision trees), τα νευρωνικά δίκτυα (neural networks) και οι αλγόριθμοι υποστήριξης διανυσμάτων (support vector machines, SVM).

Η Επιβλεπόμενη Μάθηση έχει ευρεία εφαρμογή σε τομείς όπως η υγειονομική περίθαλψη, τα χρηματοοικονομικά και η ανάλυση δεδομένων. Για παράδειγμα, χρησιμοποιείται για την πρόβλεψη ασθενειών όπως ο καρκίνος, την ανάλυση του χρηματοπιστωτικού κινδύνου και την αναγνώριση φωνής σε συστήματα όπως οι ψηφιακοί βοηθοί.

2.2.2 Μη Επιβλεπόμενη Μάθηση (Unsupervised Learning)

Στην Μη Επιβλεπόμενη Μάθηση, τα δεδομένα που χρησιμοποιούνται για την εκπαίδευση του μοντέλου δεν περιλαμβάνουν τις ετικέτες εξόδου. Το μοντέλο καλείται να εντοπίσει κρυφές δομές ή μοτίβα μέσα στα δεδομένα χωρίς την καθοδήγηση επισημασμένων εξόδων.Ο στόχος είναι ο εντοπισμός θεμελιωδών προτύπων, όπως ομάδες (clusters) ή τάσεις, που μπορούν να χρησιμοποιηθούν για περαιτέρω ανάλυση ή λήψη αποφάσεων [6].

Ορισμένες βασικές εφαρμογές της Μη Επιβλεπόμενης Μάθησης περιλαμβάνουν:

- **Ομαδοποίηση (Clustering):** Το clustering χρησιμοποιείται για την κατηγοριοποίηση ανάλογων δεδομένων σε ομάδες ή κλάσεις. Τα δεδομένα εντός της ίδιας ομάδας παρουσιάζουν μεγαλύτερη ομοιότητα σε σύγκριση με δεδομένα από άλλες ομάδες. Η ομαδοποίηση χρησιμοποιείται σε εφαρμογές όπως η τμηματοποίηση πελατών για στοχευμένη διαφήμιση και η ανάλυση προτύπων σε εκτεταμένα σύνολα δεδομένων.
- **Μείωση διάστασης (Dimensionality Reduction):** Η μείωση διάστασης περιλαμβάνει την μείωση της ποσότητας των χαρακτηριστικών στο σύνολο δεδομένων, διατηρώντας παράλληλα τις βασικές πληροφορίες.

  Η ανάλυση κύριων συνιστωσών ((Principal Component Analysis - PCA) και αλγόριθμοι όπως οι αυτοκωδικοποιητές (autoencoders) συγκαταλέγονται μεταξύ των πιο διαδεδομένων τεχνικών μείωσης διαστάσεων.

Η Μη Επιβλεπόμενη Μάθηση χρησιμοποιείται σε εφαρμογές που απαιτούν αναγνώριση προτύπων χωρίς επίβλεψη, όπως η ανίχνευση απάτης στις χρηματοπιστωτικές συναλλαγές, οι συστάσεις προϊόντων στο ηλεκτρονικό εμπόριο και η ανάλυση κοινωνικών δικτύων.

2.2.3 Ενισχυτική Μάθηση (Reinforcement Learning)

Σε αντίθεση με την επιβλεπόμενη και τη μη επιβλεπόμενη μάθηση, η Ενισχυτική Μάθηση είναι ένα εξειδικευμένο υποσύνολο της μηχανικής μάθησης στο οποίο ο πράκτορας (agent) διδάσκεται με βάση τις αλληλεπιδράσεις του με το περιβάλλον και τις ανταμοιβές (rewards) ή τις ποινές (penalties) που λαμβάνει και όχι με βάση συγκεκριμένα δεδομένα εισόδου και εξόδου. Καθώς αλληλεπιδρά με το περιβάλλον, ο πράκτορας μαθαίνει κάνοντας λάθη και προσπαθεί να βελτιστοποιήσει τη συνολική ανταμοιβή του [9].

Κάποιες από τις βασικές εφαρμογές της Ενισχυτικής Μάθησης περιλαμβάνουν:

- **Αυτόνομη Οδήγηση:** Στην αυτόνομη οδήγηση, τα οχήματα χρησιμοποιούν Ενισχυτική Μάθηση για να μάθουν να πλοηγούνται σε περίπλοκα περιβάλλοντα, παίρνοντας αποφάσεις σε πραγματικό χρόνο για να αποφύγουν εμπόδια και να βελτιώσουν τη συνολική τους απόδοση.
- **Ρομποτική:** Η Ενισχυτική Μάθηση χρησιμοποιείται για την εκπαίδευση των ρομπότ έτσι ώστε να μάθουν πώς να αλληλεπιδρούν με το περιβάλλον τους, να εκπαιδευτούν να εκτελούν δύσκολες εργασίες, όπως το να σηκώνουν πράγματα ή να μεταφέρουν βάρη.
- **Παιχνίδια και Στρατηγική:** Η Ενισχυτική Μάθηση έχει χρησιμοποιηθεί για την ανάπτυξη αλγορίθμων που επιτυγχάνουν υψηλές επιδόσεις σε πολύπλοκα στρατηγικά παιχνίδια, όπως το Go ή το σκάκι, όπου οι πράκτορες (agents) αναπτύσσουν στρατηγικές βασισμένες στην ανταμοιβή (reward) που κερδίζουν με κάθε κίνηση.

Η Ενισχυτική Μάθηση είναι ιδιαίτερα χρήσιμη για την επίλυση προβλημάτων σε δυναμικά και ασαφή περιβάλλοντα, όπου οι ενέργειες του πράκτορα (agent) δεν επηρεάζουν άμεσα το αποτέλεσμα, αλλά επηρεάζονται από την αλληλεπίδραση με το περιβάλλον.

2.2.4 Ημι-επιβλεπόμενη Μάθηση (Semi-Supervised Learning)

Η Ημι-επιβλεπόμενη Μάθηση είναι ένας ενδιάμεσος τύπος μάθησης που συνδυάζει στοιχεία από την Επιβλεπόμενη και τη Μη Επιβλεπόμενη Μάθηση. Στην Ημι-επιβλεπόμενη Μάθηση, το μοντέλο εκπαιδεύεται με ένα μικρό ποσοστό επισημασμένων δεδομένων και μεγάλο ποσοστό μη επισημασμένων δεδομένων, γεγονός που καθιστά την εκπαίδευση πιο αποδοτική, ειδικά όταν η συλλογή επισημασμένων δεδομένων είναι δαπανηρή ή χρονοβόρα [11].

Αυτή η μέθοδος χρησιμοποιείται σε τομείς όπως:

- **Επεξεργασία Φυσικής Γλώσσας:** Όταν οι ετικέτες (όπως η κατηγοριοποίηση κειμένου ή η ανάλυση συναισθήματος) δεν είναι πάντα προσβάσιμες για όλα τα δεδομένα, η ανάλυση κειμένου γίνεται με τη χρήση Ημι-επιβλεπόμενης Μάθησης.
- **Ανάλυση Εικόνας:** Στην υπολογιστική όραση, η Ημι-επιβλεπόμενη Μάθηση χρησιμοποιείται για την αναγνώριση αντικειμένων σε εικόνες όπου οι ετικέτες είναι μερικώς διαθέσιμες.

2.2.5 Μάθηση Συνόλου (Ensemble Learning)

Η Μάθηση Συνόλου (Ensemble Learning) Η μάθηση συνόλου είναι μια τεχνική μηχανικής μάθησης που ενσωματώνει πολλά μοντέλα για να βελτιώσει την απόδοση πέρα από εκείνη ενός μεμονωμένου μοντέλου. Η τεχνική αυτή βασίζεται στην παραδοχή ότι η συγχώνευση πολλών μοντέλων μετριάζει τις ελλείψεις των μεμονωμένων μοντέλων και βελτιώνει τη συνολική αποτελεσματικότητα του συστήματος [13]. Η μάθηση συνόλου χρησιμοποιείται για να βελτιώσει την ακρίβεια, τη σταθερότητα και τη γενίκευση του μοντέλου, άρα να μετριάσει τα λάθη και τα προβλήματα υπερπροσαρμογής (overfitting).

- **Βασικές Μέθοδοι Μάθησης Συνόλου**

Bagging (Bootstrap Aggregating)

Μια από τις πιο συχνά χρησιμοποιούμενες στρατηγικές μάθησης συνόλου είναι το bagging, η οποία προσπαθεί να αυξήσει την ανθεκτικότητα του μοντέλου στα σφάλματα εκπαίδευσης μειώνοντας τη διακύμανσή του. Ο όρος «bagging» περιγράφει την τεχνική bootstrap aggregating, η οποία δημιουργεί διάφορα υποσύνολα των δεδομένων εκπαίδευσης με επαναλαμβανόμενη τυχαία δειγματοληψία των δεδομένων με αντικατάσταση. Ένα μοντέλο εκπαιδεύεται σε κάθε υποσύνολο και οι προβλέψεις του μοντέλου αθροίζονται στη συνέχεια είτε με μέσο όρο (για παλινδρόμηση) είτε με ψηφοφορία (για ταξινόμηση). Τα δέντρα αποφάσεων και άλλοι αλγόριθμοι υψηλής διακύμανσης επωφελούνται σε μεγάλο βαθμό από αυτή την τεχνική. Ένα από τα πιο δημοφιλή μοντέλα, το Random Forest, είναι μια παραλλαγή bagging στην οποία το σύνολο δημιουργείται με τη χρήση δέντρων απόφασης [12].

Boosting

Μια άλλη μέθοδος μάθησης συνόλου που διαφοροποιείται από το bagging ονομάζεται «boosting», η οποία αποσκοπεί στην ενίσχυση ενός φτωχού μοντέλου με τη σταδιακή εκπαίδευση νέων μοντέλων που επικεντρώνονται στα λάθη του παλιού. Η θεμελιώδης ιδέα πίσω από το boosting είναι ότι χτίζει ένα ισχυρότερο μοντέλο συνδυάζοντας απλά μοντέλα, τα οποία είναι συνήθως αδύναμα, όπως τα δέντρα αποφάσεων ενός επιπέδου. Το αλγοριθμικό μοντέλο αναγκάζει το επόμενο μοντέλο να επικεντρωθεί στις πιο δύσκολες καταστάσεις αυξάνοντας τα βάρη των λανθασμένων παραδειγμάτων σε κάθε στάδιο της διαδικασίας [14]. Το AdaBoost και το Gradient Boosting είναι γνωστές περιπτώσεις της προσέγγισης μάθησης ενίσχυσης.

Έχει αποδειχθεί ότι οι τεχνικές Boosting λειτουργούν ιδιαίτερα καλά για δεδομένα υψηλών διαστάσεων και δύσκολα ζητήματα.

Stacking (Stacked Generalization)

Μια διαφορετική προσέγγιση για το συνδυασμό μοντέλων προτείνεται από τη μέθοδο Stacking. Το stacking χρησιμοποιεί έναν δεύτερο «μετα-αλγόριθμο» για την επιλογή του πιο αποτελεσματικού τρόπου συγκέντρωσης των προβλέψεων από τα πολλά μοντέλα, σε αντίθεση με την ψηφοφορία ή τον μέσο όρο. Το stacking είναι μια τεχνική όπου ένα δεύτερο μοντέλο εκπαιδεύεται χρησιμοποιώντας τις εξόδους των κύριων μοντέλων (π.χ.

δέντρα αποφάσεων, νευρωνικά δίκτυα) και στη συνέχεια συνδυάζει τις προβλέψεις των πρώτων μοντέλων για να δημιουργήσει το τελικό συμπέρασμα. Όταν τα κύρια μοντέλα προσφέρουν διαφορετικές «προοπτικές» για το ίδιο θέμα, το stacking έχει δείξει ιδιαίτερα ισχυρά αποτελέσματα [15].

- **Τύποι Συνδυασμού Μοντέλων**

Ψηφοφορία (Voting)

Η τεχνική της **ψηφοφορίας** εφαρμόζεται κυρίως σε προβλήματα ταξινόμησης, όπου η πλειοψηφία των ψήφων καθορίζει το τελικό αποτέλεσμα. Κάθε μοντέλο στο σύνολο κάνει μια πρόβλεψη για την κατηγορία του παραδείγματος. Κάθε μοντέλο δίνει μια ψήφο για μια κατηγορία στο σενάριο πλειοψηφικής ψηφοφορίας και η κατηγορία με τις περισσότερες ψήφους γίνεται η τελική πρόβλεψη.

Μέσος Όρος (Averaging)

Τα ζητήματα παλινδρόμησης, όπου τα μοντέλα παράγουν συνεχείς τιμές, αποτελούν την κύρια εφαρμογή για τη μέθοδο του μέσου όρου. Η τελική πρόβλεψη για το παράδειγμα καθορίζεται με τη μέση τιμή των προβλέψεων που γίνονται από διάφορα μοντέλα. Αυτή η τεχνική βελτιώνει την ακρίβεια του συστήματος και μειώνει τα σφάλματα.

- **Πλεονεκτήματα της Μάθησης Συνόλου**

Η χρήση στρατηγικών ομαδικής μάθησης έχει πολλά σημαντικά οφέλη, όπως:

1. **Βελτίωση της ακρίβειας**: Μια πιο ακριβής πρόβλεψη μπορεί να επιτευχθεί με το συνδυασμό πολλαπλών μοντέλων αντί της χρήσης ενός μόνο.
2. **Μείωση του overfitting**: Το bagging και άλλες στρατηγικές μάθησης συνόλου μειώνουν την πιθανότητα τα μοντέλα να προσαρμόζονται υπερβολικά στο σύνολο εκπαίδευσης.
3. **Αύξηση της σταθερότητας**: Η τελική πρόβλεψη μπορεί να γίνει πιο αξιόπιστη με τη χρήση των άλλων μοντέλων για τη διόρθωση ανακριβών ή αναξιόπιστων προβλέψεων από ένα μοντέλο.

2.2.6 Μεταφορά Μάθησης (Transfer Learning)

Η **Μεταφορά Μάθησης (Transfer Learning)** είναι μια πολύ σημαντική και αναπτυσσόμενη μέθοδος που επιτρέπει την εκπαίδευση μοντέλων σε νέα θέματα χρησιμοποιώντας μοντέλα που έχουν εκπαιδευτεί στο παρελθόν σε σχετικά θέματα. Η θεμελιώδης αρχή της μεταφορικής μάθησης είναι η ανακύκλωση της γνώσης που αποκτήθηκε από μια δραστηριότητα και η εφαρμογή της σε μια διαφορετική, αλλά γενικά συναφή εργασία [16]. Όταν δεν υπάρχουν επαρκή δεδομένα για τη διδασκαλία ενός νέου μοντέλου αλλά αρκετά δεδομένα για την τρέχουσα εργασία, η μάθηση μεταφοράς γίνεται πιο κρίσιμη.

- **Στάδια της Μεταφοράς Μάθησης**

Παρακάτω παρουσιάζονται τα στάδια της τεχνικής της Μεταφοράς Μάθησης.

Προ-εκπαίδευση (Pre-training)

Η προ-εκπαίδευση του μοντέλου σε ένα μεγάλο σύνολο δεδομένων από μια σχετική εργασία ή περιοχή σηματοδοτεί το αρχικό στάδιο της μεταφοράς μάθησης. Το στάδιο αυτό αποσκοπεί στην απόκτηση γενικών δεξιοτήτων που εφαρμόζονται στην καινούρια εργασία. Στα μοντέλα επεξεργασίας εικόνας, για παράδειγμα, ένα δίκτυο μπορεί να διδαχθεί να επιλέγει σημαντικά χαρακτηριστικά από τεράστιες συλλογές εικόνων (όπως το ImageNet).

Μεταφορά (Transfer)

Η εφαρμογή των χαρακτηριστικών που αποκτήθηκαν στην προηγούμενη φάση βοηθά το μοντέλο να εκτελέσει τη νέα εργασία μετά την προ-εκπαίδευση. Συνήθως, το μοντέλο προσαρμόζεται με την εκπαίδευσή του σε δεδομένα από το νέο πρόβλημα, μειώνοντας έτσι την ανάγκη πολλών δεδομένων.

Εξειδίκευση (Fine-tuning)

Η **εξειδίκευση** είναι η πρόσθετη εκπαίδευση ενός προ-εκπαιδευμένου μοντέλου με βάση δεδομένα από μια νέα εργασία. Συνήθως εκπαιδεύοντας μόνο λίγα στρώματα του δικτύου ή εκπαιδεύοντας τις πιο «επιφανειακές» παραμέτρους, τα βάρη του μοντέλου επαναπροσδιορίζονται και οι παράμετροι του αλλάζουν κατά τη λεπτή ρύθμιση ώστε να ταιριάζουν στις απαιτήσεις της νέας εργασίας [29].

- **Τύποι Μεταφοράς Μάθησης**

Υπάρχουν κάποιοι διαφορετικοί τύποι μεταφοράς μοντέλων όπως θα αναφερθούν παρακάτω.

Μέθοδος εξαγωγής χαρακτηριστικών (Feature Extraction)

Αυτή η μέθοδος εκπαιδεύει ένα νέο μοντέλο σε δεδομένα του νέου τομέα ή προβλήματος χρησιμοποιώντας τα χαρακτηριστικά που έχουν μάθει από το προ-εκπαιδευμένο μοντέλο. Όταν οι κοινές δομές χαρακτηριστικών μεταξύ των δύο τομέων είναι ισχυρές, η μέθοδος αυτή είναι ιδιαίτερα χρήσιμη.

Μεταφορά Επαγγελματικής Γνώσης (Domain Adaptation)

Η **μεταφορά επαγγελματικής γνώσης** σχετίζεται με τη γενική γνώση που αποκτάται, ακόμη και αν επικεντρώνεται στην περίπτωση που τα δεδομένα του νέου έργου διαφέρουν από εκείνα του αρχικού έργου. Μία από τις βασικές δυσκολίες είναι η προσαρμογή των μοντέλων ώστε να λειτουργούν με επιτυχία σε διάφορους τομείς χωρίς πλήρη καθοδήγηση.

Μεταφορά Εργασίας (Task Transfer)

Η **μεταφορά εργασίας** είναι η εφαρμογή της γνώσης που αποκτήθηκε από μια εργασία σε μια άλλη, αλλά συνδεδεμένη, εργασία. Τα μοντέλα μπορούν να εφαρμοστούν μεταξύ διαφόρων εργασιών με συναφείς δομές ή απαιτήσεις.

- **Εφαρμογές της Μεταφοράς Μάθησης**

Η μεταφορά μάθησης είναι εξαιρετικά χρήσιμη σε εφαρμογές και περιοχές όπως:

- **Αναγνώριση Εικόνας**: Η αναγνώριση εικόνας είναι η ικανότητα προ-εκπαιδευμένων μοντέλων να αναγνωρίζουν πρόσωπα ή αντικείμενα σε φρέσκες φωτογραφίες με μικρή είσοδο.
- **Επεξεργασία Φυσικής Γλώσσας (NLP)**: Τα μοντέλα επεξεργασίας φυσικής γλώσσας (NLP), συμπεριλαμβανομένων των BERT και GPT, μπορούν να προσαρμοστούν σε συγκεκριμένες εργασίες, συμπεριλαμβανομένης της ανάλυσης συναισθήματος ή της μετάφρασης κειμένου, αφού εκπαιδευτούν σε μαζικά σύνολα δεδομένων κειμένου.

2.3. Αλγόριθμοι Μηχανικής Μάθησης στην Επιβλεπόμενη Μάθηση

Ένα από τα πιο θεμελιώδη υποπεδία της μηχανικής μάθησης, η επιβλεπόμενη μάθηση (supervised learning) χρησιμοποιεί αλγορίθμους που εκπαιδεύονται σε ένα σύνολο δεδομένων στο οποίο κάθε είσοδος συνδέεται με ένα αντίστοιχο επιδιωκόμενο αποτέλεσμα-έξοδο. Αυτές οι μέθοδοι επιδιώκουν να μάθουν τη σύνδεση μεταξύ εισόδων και αποτελεσμάτων-εξόδων, επιτρέποντας έτσι την πρόβλεψη των αποτελεσμάτων για νέες εισόδους που δεν είναι γνωστές από την προηγούμενη εμπειρία [6].

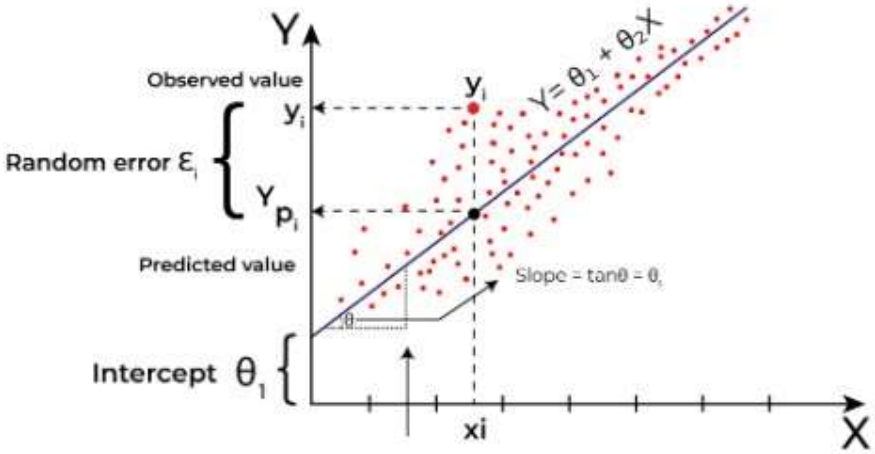
Από την ανάλυση δεδομένων έως την ιατρική διάγνωση, την ανίχνευση απάτης και την ομαδοποίηση καταναλωτών [17], ορισμένοι από τους πιο γνωστούς και σημαντικούς αλγορίθμους που εφαρμόζονται στην επιβλεπόμενη μάθηση και θα αναλυθούν παρακάτω είναι η Γραμμική Παλινδρόμηση, η Λογιστική Παλινδρόμηση, τα Δέντρα Αποφάσεων και τα Τυχαία Δάση.

2.3.1 Γραμμική Παλινδρόμηση (Linear Regression)

Με βάση μία ή περισσότερες ανεξάρτητες μεταβλητές, η γραμμική παλινδρόμηση -ένας από τους θεμελιώδεις και πιο συχνά χρησιμοποιούμενους αλγορίθμους μηχανικής μάθησης- προβλέπει μια συνεχή εξαρτημένη μεταβλητή [19]. Η γραμμική παλινδρόμηση ουσιαστικά αναζητά τη γραμμική σχέση μεταξύ της εξαρτημένης μεταβλητής (έξοδος) και των χαρακτηριστικών (είσοδοι). Το μοντέλο δηλώνεται χρησιμοποιώντας τη μαθηματική εξίσωση:

$$Y=\beta_0+\beta_1X_1+\beta_2X_2+...+\beta_nX_n+\varepsilon$$

όπου Y είναι η εξαρτημένη μεταβλητή, β_0 είναι η σταθερά (το σημείο τομής με τον άξονα Y) και β_1, β_2, ..., β_n είναι οι συντελεστές των ανεξάρτητων μεταβλητών X_1, X_2, ..., X_n, και ε είναι το σφάλμα της εκτίμησης.



Εικόνα 1: Απεικόνιση της Γραμμικής Παλινδρόμησης. Η εικόνα παρουσιάζει τη γραφική παράσταση με τις τυχαίες τιμές, αλλά και τις εξισώσεις της γραμμικής παλινδρόμησης. [20]

Η εύρεση των καλύτερων συντελεστών που μειώνουν το άθροισμα των τετραγώνων των αποκλίσεων μεταξύ των πραγματικών τιμών και των προβλεπόμενων τιμών εκπαιδεύει το μοντέλο. Η προσέγγιση αυτή, που χρησιμοποιείται ευρέως και αναγνωρίζεται στην ανάλυση δεδομένων και τη στατιστική, είναι γνωστή ως "Μέθοδος Ελαχίστων Τετραγώνων" (Least Squares Method) [24].

Η αξιολόγηση του μοντέλου γίνεται συνήθως με τη βοήθεια του Συντελεστή Προσδιορισμού R^2, ο οποίος δείχνει το ποσοστό της διακύμανσης της εξαρτημένης μεταβλητής που εξηγείται από το μοντέλο. Άλλες μετρικές αξιολόγησης περιλαμβάνουν την Μέση Τετραγωνική Απόκλιση (MSE), την Ρίζα της Μέσης Τετραγωνικής Απόκλισης (RMSE), και τη Μέση Απόλυτη Απόκλιση (MAE) [19].

Πλεονεκτήματα και Μειονεκτήματα

Όπως θα καλυφθεί εδώ, η Γραμμική Παλινδρόμηση έχει αρκετά πλεονεκτήματα αλλά και μειονεκτήματα.

Πλεονεκτήματα:

Κάποια από τα πλεονεκτήματα του αλγορίθμου της Γραμμικής Παλινδρόμησης είναι ότι ο συγκεκριμένος αλγόριθμος είναι:

- Απλός και κατανοητός.
- Εκπαιδεύεται γρήγορα, άρα είναι κατάλληλος για μεγάλα δεδομένα.
- Εφαρμόζεται κυρίως στις κοινωνικές, ιατρικές και οικονομικές επιστήμες.

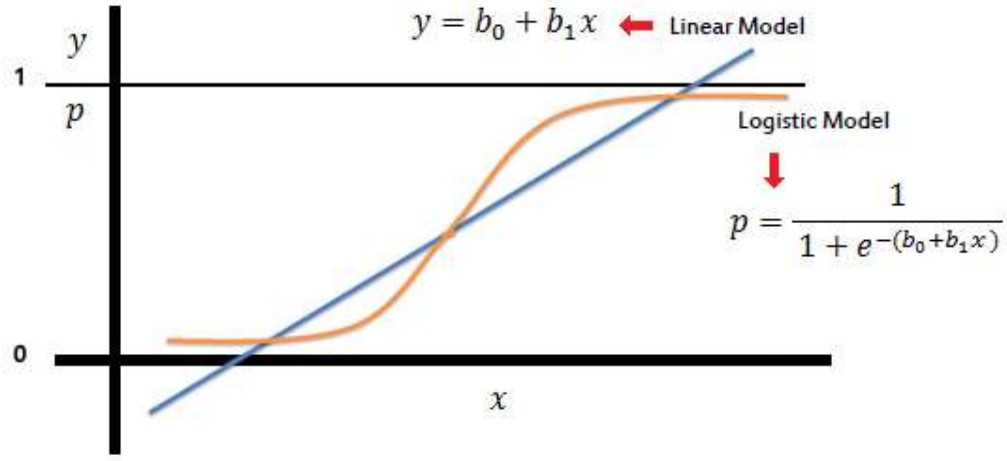
Μειονεκτήματα:

Όπως και άλλοι αλγόριθμοι, έχει ορισμένα μειονεκτήματα- μερικά από αυτά απαριθμούνται εδώ:

- Προϋποθέτει μια γραμμική σύνδεση μεταξύ μεταβλητών.
- Μπορεί να προκαλέσει προβλήματα σε περιπτώσεις πολλαπλής συγγραμικότητας - δηλαδή όταν οι ανεξάρτητες μεταβλητές συνδέονται έντονα μεταξύ τους.
- Δυσκολεύεται με μη γραμμικά δεδομένα ή με δεδομένα που περιλαμβάνουν ακραίες τιμές (outliers).

2.3.2 Λογιστική Παλινδρόμηση (Logistic Regression)

Παρά το γεγονός ότι αναφέρεται ως «παλινδρόμηση», η λογιστική παλινδρόμηση χρησιμοποιείται κυρίως για ταξινόμηση και όχι για προβλέψεις συνεχών μεταβλητών. Η πιθανότητα μια παρατήρηση να εμπίπτει σε μια από τις δύο κατηγορίες -για παράδειγμα, θετική ή αρνητική, 0 ή 1-προβλέπεται με τη χρήση της λογιστικής παλινδρόμησης [22].



Εικόνα 2: Απεικόνιση της Λογιστικής Παλινδρόμησης. Η εικόνα παρουσιάζει τη γραφική παράσταση μαζί με τις εξισώσεις της λογιστικής παλινδρόμησης. [23]

Η κύρια διάκριση μεταξύ της λογιστικής παλινδρόμησης και της γραμμικής παλινδρόμησης είναι ότι η έξοδος του μοντέλου είναι μια πιθανότητα μεταξύ 0 και 1 και όχι ένας συνεχής αριθμός. Η λογιστική συνάρτηση, συχνά γνωστή ως σιγμοειδής συνάρτηση (sigmoid function), χρησιμοποιείται για τον προσδιορισμό αυτής της πιθανότητας και είναι η:

$$p = \sigma(z) = \frac{1}{1+e^{-z}},$$

όπου $z=\beta_0+\beta_1X_1+\beta_2X_2+...+\beta_nX_n$ είναι ο γραμμικός συνδυασμός των χαρακτηριστικών και των συντελεστών, και $\sigma(z)$ είναι η λογιστική συνάρτηση που περιορίζει το αποτέλεσμα μεταξύ 0 και 1.

Η εκπαίδευση του μοντέλου λογιστικής παλινδρόμησης γίνεται με τη μέθοδο της μέγιστης πιθανοφάνειας (Maximum Likelihood Estimation), η οποία επιδιώκει τη μεγιστοποίηση της πιθανότητας των παρατηρούμενων δεδομένων [21].

Η αξιολόγηση του μοντέλου πραγματοποιείται με διαφορετικού είδους μετρήσεων, όπως η ακρίβεια (accuracy), η ανάκληση (recall), η ακρίβεια θετικών προγνώσεων (precision), το F1 score, και η περιοχή κάτω από την καμπύλη λειτουργίας δέκτη (AUC-ROC) [25].

Πλεονεκτήματα και Μειονεκτήματα

Όπως θα καλυφθεί παρακάτω, υπάρχουν ορισμένα πλεονεκτήματα και μειονεκτήματα από την υιοθέτηση της λογιστικής παλινδρόμησης.

Πλεονεκτήματα:

Ακολουθούν ορισμένα από τα σημαντικότερα οφέλη του αλγορίθμου λογιστικής παλινδρόμησης:

- Απλή και εύκολη στην εφαρμογή.
- Παρέχει πιθανολογικές εκτιμήσεις, χρήσιμες σε πολλές εφαρμογές (π.χ., ανάλυση κινδύνου).
- Κατάλληλη για δυαδική ταξινόμηση.

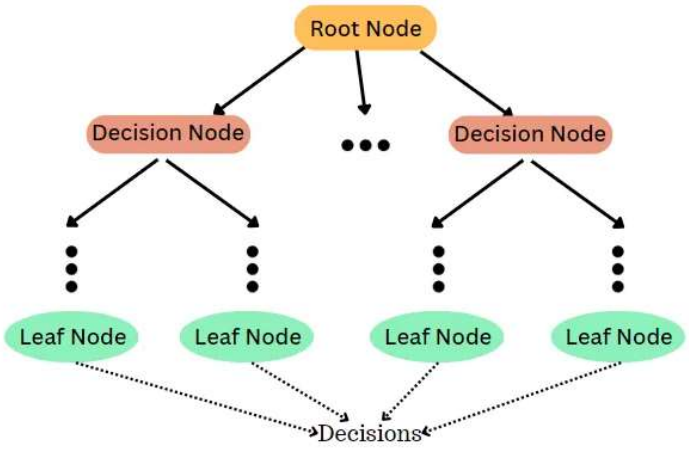
Μειονεκτήματα:

Κάποια από τα κυριότερα μειονεκτήματα του αλγορίθμου της Λογιστικής Παλινδρόμησης είναι ότι:

- Υποθέτει ότι τα χαρακτηριστικά και η πιθανότητα έχουν γραμμική σχέση.
- Δεν είναι ιδανική για πολύπλοκες, μη γραμμικές σχέσεις.

2.3.3 Δέντρα Αποφάσεων (Decision Trees)

Τα δέντρα αποφάσεων είναι από τις πλέον αναγνωρισμένες και εκτενώς χρησιμοποιούμενες μεθόδους ταξινόμησης και παλινδρόμησης [18]. Ένα δέντρο απόφασης χρησιμοποιεί ένα ιεραρχικό πλαίσιο για τις προβλέψεις, με κάθε κόμβο να αντιπροσωπεύει μια διάσπαση δεδομένων και τα φύλλα να περιέχουν τις τελικές προβλέψεις.



Εικόνα 3: Απεικόνιση των Δέντρων αποφάσεων. Η εικόνα παρουσιάζει ένα ιεραρχικό πλαίσιο ενός δέντρου απόφασης μαζί με τα φύλλα του. [26]

Η κατασκευή του δέντρου αποφάσεων ξεκινά από τη ρίζα, η οποία περιέχει το σύνολο των δεδομένων. Στη συνέχεια, το δέντρο χωρίζει τα δεδομένα σε υποομάδες σύμφωνα με συγκεκριμένα χαρακτηριστικά ή κριτήρια, χρησιμοποιώντας κριτήρια διαχωρισμού όπως το κέρδος πληροφορίας (information gain) ή ο δείκτης Gini (Gini index). Η διαδικασία αυτή εκτελείται αναδρομικά σε κάθε κόμβο του δέντρου μέχρι να ικανοποιηθούν συγκεκριμένες απαιτήσεις, όπως το μέγιστο βάθος του δέντρου ή το ελάχιστο πλήθος παρατηρήσεων σε έναν κόμβο [27].

Πλεονεκτήματα και Μειονεκτήματα

Υπάρχουν κάποια πλεονεκτήματα αλλά και μειονεκτήματα χρήσης του αλγορίθμου των Δέντρων Αποφάσεων, όπως θα αναφερθούν παρακάτω.

Πλεονεκτήματα:

Κάποια από τα πλεονεκτήματα του αλγορίθμου των Δέντρων Αποφάσεων είναι ότι:

- Είναι πολύ κατανοητά και ερμηνεύσιμα μοντέλα.
- Μπορούν να χειριστούν τόσο αριθμητικά όσο και κατηγορικά δεδομένα.
- Δεν απαιτούν κλιμάκωση ή κανονικοποίηση των δεδομένων.

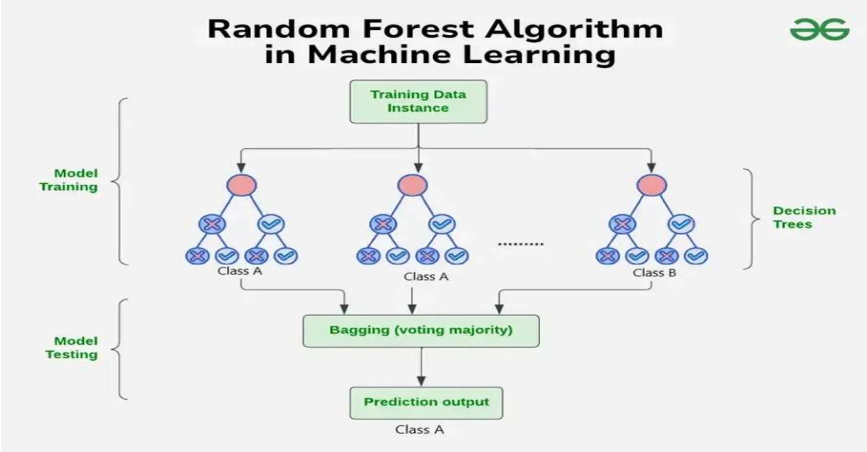
Μειονεκτήματα:

Πέρα από τα πλεονεκτήματα, ο αλγόριθμος έχει και κάποια μειονεκτήματα όπως:

- Ευαισθησία στην υπερπροσαρμογή (overfitting).
- Δυσκολία στην ακριβή εκτίμηση του μοντέλου για μεγάλα δεδομένα με πολλά χαρακτηριστικά.

2.3.4 Τυχαία Δάση (Random Forests)

Τα Τυχαία Δάση (Random Forests) είναι μια τεχνική μηχανικής μάθησης που χρησιμοποιεί ένα σύνολο δέντρων αποφάσεων για να δημιουργήσει ένα ισχυρό και σταθερό μοντέλο [12]. Κάθε δέντρο στο δάσος εκπαιδεύεται σε ένα ξεχωριστό υποσύνολο των δεδομένων μέσω bootstrapping και επιλέγεται ένα τυχαίο υποσύνολο χαρακτηριστικών σε κάθε κόμβο ενός δέντρου. Αυτός ο τυχαίος συνδυασμός μειώνει τη συσχέτιση μεταξύ των δέντρων και ενισχύει τη συνολική απόδοση του μοντέλου.



Εικόνα 4: Απεικόνιση του Random Forest. Η εικόνα παρουσιάζει το ιεραρχικό πλαίσιο μαζί με τους κόμβους και τα δέντρα απόφασης και την διαδικασία υλοποίηση του αλγορίθμου. [28]

Η πρόβλεψη για ένα δείγμα καθορίζεται είτε με ψηφοφορία (voting) είτε με μέσο όρο (mean) των προβλέψεων των δέντρων του δάσους, ανάλογα με το αν πρόκειται για ταξινόμηση ή παλινδρόμηση, αντίστοιχα.

Πλεονεκτήματα και Μειονεκτήματα

Η μέθοδος Random Forest παρουσιάζει πλεονεκτήματα και μειονεκτήματα, τα οποία θα αναλυθούν παρακάτω.

Πλεονεκτήματα:

Κάποια από τα πλεονεκτήματα του αλγορίθμου Random Forest είναι ότι:

- Έχουμε υψηλή ακρίβεια και σταθερότητα.
- Διαθέτει την ικανότητα διαχείρισης πολύπλοκων δεδομένων και μεγάλης διαστατικότητας.
- Μειώνει την πιθανότητα υπερπροσαρμογής.

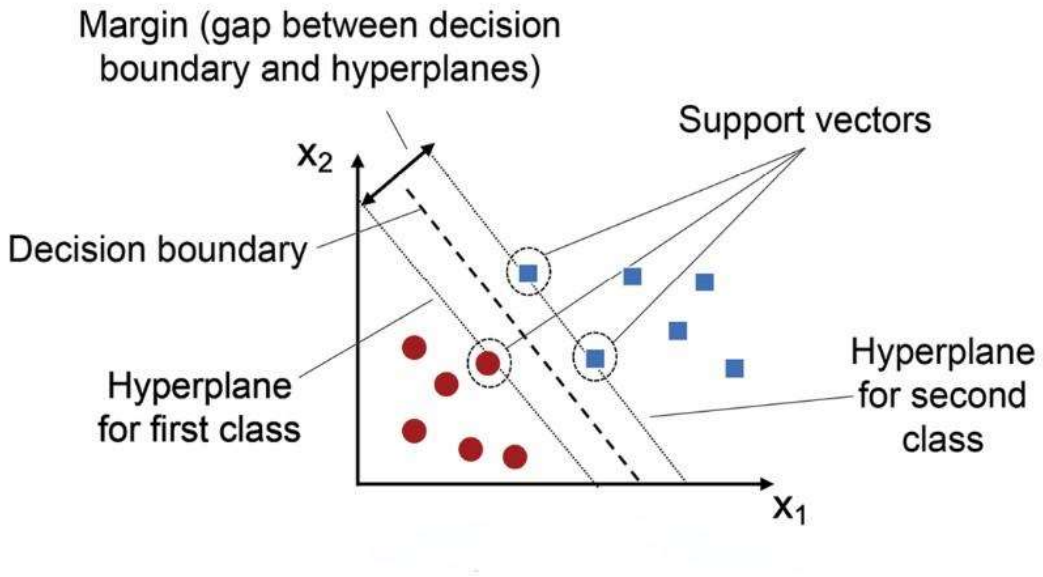
Μειονεκτήματα:

Μερικά από τα μειονεκτήματα αυτής της τεχνικής μηχανικής μάθησης είναι ότι:

- Γενικά είναι πολύπλοκα μοντέλα που μπορεί να είναι δύσκολα στην ερμηνεία.
- Απαιτούν υψηλούς υπολογιστικούς πόρους, ιδιαίτερα με μεγάλες ποσότητες δεδομένων.

2.3.5. Μηχανές Διανυσμάτων Υποστήριξης (Support Vector Machines - SVM)

Οι Μηχανές Διανυσμάτων Υποστήριξης (SVM) είναι ένας αλγόριθμος μηχανικής μάθησης που χρησιμοποιείται ευρέως για την επίλυση προβλημάτων ταξινόμησης και παλινδρόμησης. Η ικανότητά τους να διαχειρίζονται δεδομένα υψηλών διαστάσεων και η αποτελεσματική τους ανάλυση πολύπλοκων γεωμετρικών δομών, τέτοιων μη γραμμικά διαχωρίσιμων δεδομένων, τις κάνουν να ξεχωρίζουν [33]. Η εύρεση του υπερεπιπέδου (ή των υπερεπιπέδων στην περίπτωση της ταξινόμησης πολλαπλών κλάσεων) που διαχωρίζει διάφορες κλάσεις δεδομένων με το μεγαλύτερο εφικτό περιθώριο είναι ο πρωταρχικός στόχος των SVM.



Εικόνα 5: Απεικόνιση του Support Vector Machines. Η εικόνα παρουσιάζει τη γραφική παράσταση με τις τυχαίες τιμές και την υλοποίηση του αλγορίθμου. [31]

Βασικές Έννοιες

Κάποιες από τις βασικές έννοιες που είναι σημαντικές στον αλγόριθμο είναι:

- **Το Υπερεπίπεδο**: Ένας n-διάστατος χώρος διαχωρίζεται από ένα (n-1)-διάστατο υπερεπίπεδο. Για παράδειγμα, σε δύο διαστάσεις, το υπερεπίπεδο είναι μια γραμμή. Οι SVM στοχεύουν πρωτίστως στον εντοπισμό του υπερεπιπέδου που διαχωρίζει τις κλάσεις με το μεγαλύτερο εφικτό περιθώριο, μειώνοντας έτσι τα σφάλματα ταξινόμησης.

- **Το Περιθώριο**: Η απόσταση μεταξύ του υπερεπίπεδου που διαχωρίζει τις κλάσεις και των πλησιέστερων δειγμάτων από κάθε κλάση (γνωστά και ως υποστηρικτικά διανύσματα). Οι SVM επιδιώκουν να μεγιστοποιήσουν το περιθώριο για να ενισχύσουν τη γενίκευση του μοντέλου, με στόχο τη μείωση των σφαλμάτων ταξινόμησης σε μη εμφανιζόμενα δεδομένα [33].

- **Διανύσματα Υποστήριξης**: Ο προσδιορισμός της επικάλυψης και του περιθωρίου εξαρτάται σε μεγάλο βαθμό από αυτά τα σημεία δεδομένων, άρα από τα διανύσματα υποστήριξης. Εξακριβώνουν τη θέση του υπερεπιπέδου διαχωρισμού και δείχνουν τα σημεία που βρίσκονται πλησιέστερα σε αυτό.

Λειτουργία

Τα κυριότερα χαρακτηριστικά της λειτουργίας του αλγορίθμου απαριθμούνται παρακάτω.

1. **Εκπαίδευση**: Κατά την εκπαίδευση, ο αλγόριθμος SVM προσπαθεί να βρει το υπερεπίπεδο που διαχωρίζει τις κλάσεις δεδομένων με το μέγιστο δυνατό περιθώριο. Η διαδικασία αυτή περιλαμβάνει τη βελτιστοποίηση ενός συναρτησιακού στόχου υπό περιορισμούς που εξασφαλίζουν τον σωστό διαχωρισμό των κλάσεων [30].
2. **Μη Γραμμική Ταξινόμηση**: Σε περιπτώσεις όπου τα δεδομένα δεν είναι γραμμικά διαχωρίσιμα, οι SVM χρησιμοποιούν **πυρηνικές συναρτήσεις** (kernel functions), που θα αναλυθούν παρακάτω, για να μεταφέρουν τα δεδομένα σε έναν υψηλότερο διαστατικό χώρο, όπου μπορεί να εφαρμοστεί γραμμικός διαχωρισμός. Αυτή η τεχνική επιτρέπει στο SVM να διαχειριστεί μη-γραμμικά διαχωρίσιμα δεδομένα αποτελεσματικά [33].
3. **Πρόβλεψη**: Μετά την εκπαίδευση, ο αλγόριθμος μπορεί να προβλέψει την κλάση ενός νέου δείγματος με βάση τη θέση του στο χώρο σε σχέση με το υπερεπίπεδο. Εάν το δείγμα βρίσκεται στη μία πλευρά του υπερεπιπέδου, κατατάσσεται σε μία κλάση, και στην άλλη πλευρά σε άλλη κλάση.

Συναρτήσεις Πυρήνα (Kernel Functions)

Η ικανότητα των SVM να επεξεργάζονται μη γραμμικά διαχωρίσιμα δεδομένα εξαρτάται από τις πυρηνικές συναρτήσεις [33]. Χρησιμοποιούν την ίδια προσέγγιση ταξινόμησης σε έναν χώρο υψηλότερων διαστάσεων στον οποίο τα δεδομένα γίνονται τελικά γραμμικά διαχωρίσιμα. Οι πιο συνηθισμένες πυρηνικές συναρτήσεις περιλαμβάνουν:

- **Την Γραμμική**: Όταν τα δεδομένα είναι ήδη γραμμικά διαχωρίσιμα.
- **Την Πολυωνυμική**: Εξαιρετική για δεδομένα με πολυωνυμική δομή.
- **Την Ραβδωτή Βάση (Radial Basis Function - RBF)**: Χρησιμοποιείται συχνά σε περιπτώσεις όπου τα δεδομένα δεν είναι γραμμικά διαχωρίσιμα.
- **Τη Σιγμοειδής**: Εφαρμόζεται για δεδομένα που απαιτούν σιγμοειδή μετασχηματισμό για να καταστούν γραμμικά διαχωρίσιμα.

Πλεονεκτήματα και Μειονεκτήματα

Υπάρχουν κάποια πλεονεκτήματα αλλά και μειονεκτήματα χρήσης του αλγορίθμου Support Vector Machines, όπως θα αναφερθούν παρακάτω.

Πλεονεκτήματα:

Η μέθοδος προσφέρει ορισμένα από τα ακόλουθα κύρια οφέλη:

- **Η Εξαιρετική Απόδοση σε Υψηλο-διάστατα Δεδομένα**: Ιδανικό για προβλήματα όταν ο αριθμός των διαστάσεων υπερβαίνει τον αριθμό των δειγμάτων.
- **Η Καλή Γενίκευση**: Ικανό να αποδώσει αποτελεσματικά ακόμη και σε μικρά σύνολα δεδομένων.
- **Η Αντοχή σε Υπερπροσαρμογή**: Όταν το περιθώριο μεγιστοποιείται, το μοντέλο παρουσιάζει υψηλή ικανότητα γενίκευσης σε αντίθεση με την υπερπροσαρμογή.

Μειονεκτήματα:

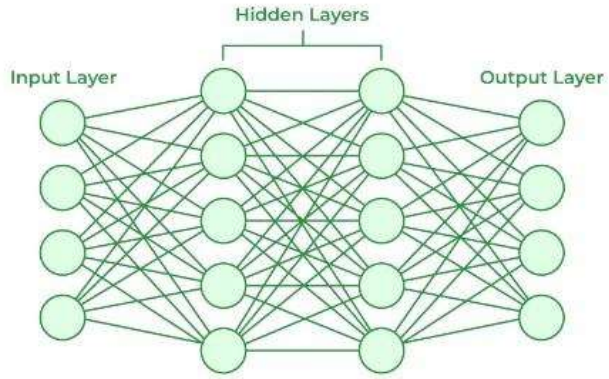
Υπάρχουν και κάποια μειονεκτήματα με τα πιο σημαντικά από αυτά να είναι:

  - **Η Υπολογιστική Δυσκολία**: Ο αλγόριθμος μπορεί να είναι απαιτητικός σε υπολογιστικούς πόρους, ιδιαίτερα σε μεγάλα σύνολα δεδομένων.
  - **Η Ανεπάρκεια στην Επιλογή Πυρηνικών Συναρτήσεων**: Η επιλογή της κατάλληλης πυρηνικής συνάρτησης και των παραμέτρων της μπορεί να είναι δύσκολη και να απαιτεί πολλές δοκιμές.
  - **Η Ευαισθησία στην Υπερπροσαρμογή σε Πολύ Υψηλές Διαστάσεις**: Εάν το περιθώριο δεν επιλεγεί σωστά, τα SVM ενδέχεται να υπερπροσαρμοστούν σε τόσο υψηλές διαστάσεις.

Οι Support Vector Machines παραμένουν μια από τις πιο ισχυρές και δημοφιλείς τεχνικές μηχανικής μάθησης λόγω της ικανότητάς τους να παράγουν ακριβή μοντέλα και να γενικεύουν αποτελεσματικά, ακόμη και για περίπλοκα ή λιγότερο κατανοητά δεδομένα [33]. Για πολλές εφαρμογές, συμπεριλαμβανομένης της ανάλυσης κειμένου, της αναγνώρισης προτύπων και της ανάλυσης εικόνων, η ευελιξία τους στον χειρισμό υψηλών διαστάσεων και η χρήση συναρτήσεων πυρήνα τις καθιστά τέλειες παρά τους περιορισμούς τους.

2.3.6 Νευρωνικά Δίκτυα (Neural Networks)

Εμπνευσμένα από ιδέες που προέρχονται από τη δομή και τη λειτουργία του ανθρώπινου εγκεφάλου, τα νευρωνικά δίκτυα συγκαταλέγονται στην πιο ισχυρή και προσαρμόσιμη κατηγορία τεχνικών μηχανικής μάθησης. Μεταξύ των διαφόρων χρήσεων αυτών των αλγορίθμων είναι η ταξινόμηση, η παλινδρόμηση, η εξόρυξη δεδομένων και οι εργασίες αναγνώρισης προτύπων [6]. Πολλοί τομείς της τεχνολογίας εξαρτώνται από τα νευρωνικά δίκτυα λόγω της ικανότητάς τους να επεξεργάζονται και να αξιολογούν δεδομένα με περίπλοκες σχέσεις.



Εικόνα 6: Απεικόνιση Νευρωνικού Δικτύου. Η εικόνα παρουσιάζει ένα νευρωνικό δίκτυο μαζί με την είσοδο, την έξοδο και τα κρυφά στρώματα. [32]

Βασικές Έννοιες

Τα νευρωνικά δίκτυα έχουν διάφορα θεμελιώδη και σημαντικά χαρακτηριστικά όπως τα ακόλουθα:

1. **Ο Νευρώνας (Neuron)**: Θεμελιώδης υπολογιστική μονάδα ενός νευρωνικού δικτύου είναι ο νευρώνας. Παράγει μια έξοδο χρησιμοποιώντας μια συνάρτηση ενεργοποίησης αφού λάβει είσοδο από άλλους νευρώνες ή το εξωτερικό περιβάλλον, επεξεργαζόμενος το σύνολο της εισόδου ανάλογα με τα βάρη των συνδέσεων [38].
2. **Τα Στρώματα (Layers)**: Ένα νευρωνικό δίκτυο είναι οργανωμένο σε διαδοχικά στρώματα νευρώνων. Τα λεγόμενα «κρυφά στρώματα» βρίσκονται μεταξύ των στρωμάτων εισόδου και εξόδου- το πρώτο στρώμα είναι το στρώμα εισόδου. Η δομή των στρωμάτων καθορίζει την ικανότητα του δικτύου να αντιλαμβάνεται και να επεξεργάζεται περίπλοκα μοτίβα [37], συνεπώς όσο περισσότερα στρώματα, τόσο μεγαλύτερη η ικανότητα επεξεργασίας πιο περίπλοκων μοτίβων.
3. **Τα Βάρη (Weights) και οι Τιμές (Biases)**: Τα βάρη καθορίζουν τη σημασία κάθε εισόδου για την παραγωγή της εξόδου ενός νευρώνα, ενώ οι τιμές παρέχουν σταθερές παραμέτρους που επιτρέπουν μεγαλύτερη ευελιξία στο μοντέλο [44].
4. **Η Συνάρτηση Ενεργοποίησης (Activation Function)**: Η συνάρτηση ενεργοποίησης φέρνει τη μη γραμμικότητα στο μοντέλο, επιτρέποντας έτσι στα νευρωνικά δίκτυα να μαθαίνουν περίπλοκες και μη γραμμικές συσχετίσεις [43]. Μεταξύ των πιο συχνά χρησιμοποιούμενων και γνωστών συναρτήσεων ενεργοποίησης είναι η Sigmoid, η ReLU και η Tanh.
5. **Τα Πλήρως Συνδεδεμένα Νευρωνικά Δίκτυα (Fully Connected Neural Networks - Feedforward Networks)**: Στα πλήρως συνδεδεμένα νευρωνικά δίκτυα, κάθε νευρώνας ενός στρώματος συνδέεται με όλους τους νευρώνες του επόμενου στρώματος, επιτρέποντας την πλήρη ανταλλαγή πληροφοριών μεταξύ των στρωμάτων [80]. Όπως φαίνεται και στην προηγούμενη εικόνα, έχουμε ένα τέτοιο νευρωνικό δίκτυο.
6. **Η Ενεργοποίηση (Activation)**: Η διαδικασία ενεργοποίησης είναι η εφαρμογή της συνάρτησης ενεργοποίησης στις εισόδους του νευρώνα παράγοντας έτσι την έξοδό του [37].

7. **Τα Αναδρομικά Νευρωνικά Δίκτυα (Recurrent Neural Networks - RNNs)**: Σε αντίθεση με τα feedforward δίκτυα, τα αναδρομικά νευρωνικά δίκτυα επιτρέπουν τη ροή των δεδομένων να διακινείται και προς τα πίσω, δημιουργώντας έτσι μια δομή που μπορεί να αποθηκεύσει τη γνώση του παρελθόντος και επομένως να είναι κατάλληλη για την επεξεργασία χρονοσειρών και δεδομένων ακολουθίας [45].

Λειτουργία

Σε γενικές γραμμές η λειτουργία των νευρωνικών δικτύων είναι όπως αναφέρεται παρακάτω:

1. **Εκπαίδευση (Training)**: Στη διαδικασία εκπαίδευσης, τα δεδομένα εισόδου προωθούνται μέσα από τα στρώματα του δικτύου. Η έξοδος του δικτύου συγκρίνεται με τις πραγματικές τιμές, και οι παράμετροι του δικτύου (βάρη και τιμές) ενημερώνονται ώστε να μειωθεί το σφάλμα [44]. Αυτός ο μηχανισμός μάθησης βασίζεται στη βελτιστοποίηση μέσω αλγορίθμων όπως η μέθοδος της κλίσης καθόδου (gradient descent).
2. **Οπισθοδρόμηση (Backpropagation)**: Η μέθοδος της οπισθοδρόμησης χρησιμοποιείται για τη διάδοση του σφάλματος πίσω στο δίκτυο, επιτρέποντας τη βελτίωση των παραμέτρων μέσω του υπολογισμού των παραγώγων του σφάλματος ως προς τα βάρη και τις τιμές του δικτύου [39].
3. **Συνάρτηση Κόστους (Cost Function)**: Η συνάρτηση κόστους μετρά τη διαφορά μεταξύ των προβλέψεων του δικτύου και των πραγματικών τιμών και αποτελεί τον στόχο της διαδικασίας εκπαίδευσης. Οι συνήθεις συναρτήσεις κόστους περιλαμβάνουν την μέση τετραγωνική απόκλιση (MSE) για τα προβλήματα παλινδρόμησης και την εφαπτομενική λογιστική συνάρτηση (cross-entropy) για τα προβλήματα ταξινόμησης [6].

Είδη Νευρωνικών Δικτύων

Ακολουθεί ένας κατάλογος μερικών από τις σημαντικότερες μορφές νευρωνικών δικτύων:

1. **Δίκτυα Ταξινόμησης (Classification Networks)**: Τα δεδομένα κατηγοριοποιούνται διακριτά με τη χρήση αυτών των δικτύων. Η αναγνώριση εικόνων είναι το σύνηθες πεδίο εφαρμογής τους [37].
2. **Νευρωνικά Δίκτυα Παλινδρόμησης (Regression Networks)**: Χρησιμοποιούνται για την πρόβλεψη συνεχών τιμών και αξιοποιούνται σε εφαρμογές όπως η πρόβλεψη χρηματοοικονομικών δεδομένων ή καιρικών φαινομένων [42].
3. **Αυτόματοι κωδικοποιητές (Autoencoders)**: Εφαρμόζονται για την εκμάθηση συμπιεσμένων αναπαραστάσεων δεδομένων, η τεχνική αυτή εξυπηρετεί είτε τη μείωση των διαστάσεων είτε την αναγνώριση χαρακτηριστικών [41].
4. **Συνελικτικά Νευρωνικά Δίκτυα (Convolutional Neural Networks - CNNs)**: Σχεδιασμένα ειδικά για την ανάλυση εικόνων και την αναγνώριση προτύπων, τα CNN χρησιμοποιούν συνελικτικά στρώματα που εφαρμόζουν φίλτρα στις εισόδους για την εξαγωγή χαρακτηριστικών [42], επομένως χρησιμοποιούνται στην ανάλυση δορυφορικών εικόνων κ.λπ.
5. **Μακροπρόθεσμη Μνήμη Κυττάρων (Long Short-Term Memory - LSTM) και Κύτταρα Δικτύων Μνήμης (Gated Recurrent Unit - GRU)**: Χρησιμοποιούνται για την επεξεργασία ακολουθιών δεδομένων, αντιμετωπίζοντας τα προβλήματα που σχετίζονται με την εξαφάνιση και την εκρηκτική ανάπτυξη του βαθμού κατά τη διάρκεια της εκπαίδευσης [36][40].

Πλεονεκτήματα και Μειονεκτήματα

Υπάρχουν κάποια πλεονεκτήματα αλλά και μειονεκτήματα χρήσης των Νευρωνικών Δικτύων, όπως θα αναφερθούν παρακάτω.

Πλεονεκτήματα:

Κάποια από τα πλεονεκτήματα των νευρωνικών δικτύων αναφέρονται παρακάτω.

- Είναι δυνατή η εκμάθηση πολύπλοκων συναρτήσεων και μη γραμμικών σχέσεων.
- Η ευέλικτη αρχιτεκτονική που μπορεί να προσαρμοστεί σε διάφορους τύπους προβλημάτων.
- Η ικανότητα αυτόματης εξαγωγής χαρακτηριστικών από μη επεξεργασμένα δεδομένα [37].

Μειονεκτήματα:

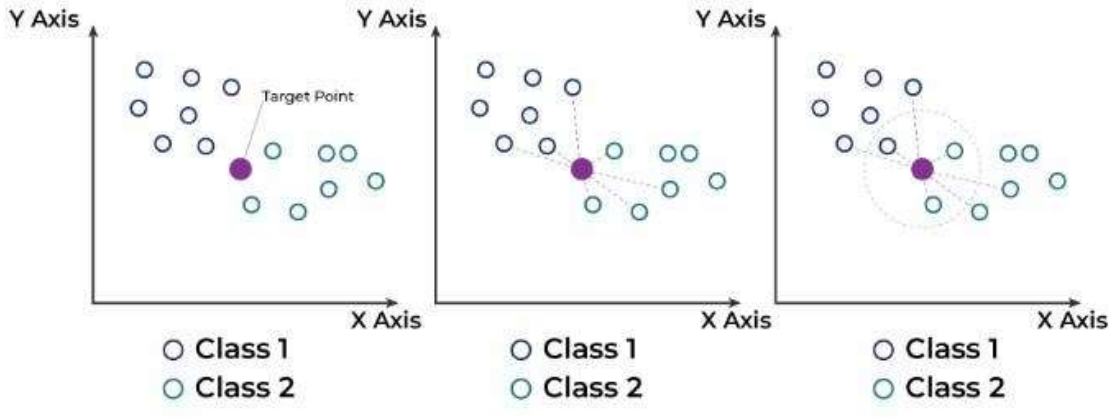
Υπάρχουν και κάποια μειονεκτήματα, με τα πιο σημαντικά να είναι:

- Απαιτούν μεγάλο όγκο δεδομένων για αποτελεσματική εκπαίδευση, καθώς και υψηλές υπολογιστικές δυνατότητες.
- Δεδομένης της φύσης των νευρωνικών δικτύων ως «μαύρα κουτιά», η ερμηνεία των αποτελεσμάτων μπορεί να είναι δύσκολη.
- Ειδικά όταν τα δεδομένα εκπαίδευσης είναι μικρά, ελλοχεύει ο κίνδυνος της υπερπροσαρμογής [6].

Σε γενικές γραμμές, η μηχανική μάθηση και η τεχνητή νοημοσύνη εξαρτώνται από τα νευρωνικά δίκτυα. Έχουν επίσης συμβάλει σημαντικά σε τομείς όπως η αναγνώριση εικόνας, η επεξεργασία φυσικής γλώσσας και η αυτόνομη οδήγηση. Ο τύπος του προβλήματος, ο όγκος και ο χαρακτήρας των δεδομένων, καθώς και τα κριτήρια ακρίβειας και απόδοσης της εφαρμογής επηρεάζουν τον κατάλληλο αλγόριθμο που πρέπει να επιλεγεί.

2.3.7. Αλγόριθμος k-Nearest Neighbors (k-NN)

Ο αλγόριθμος k-Nearest Neighbors (k-NN) εφαρμόζεται τόσο για την ταξινόμηση όσο και για την παλινδρόμηση και είναι από τις πιο συχνά χρησιμοποιούμενες τεχνικές μηχανικής μάθησης. Η μέθοδος αυτή βασίζεται στην ιδέα ότι στο χώρο χαρακτηριστικών (features), τα στενά διατεταγμένα μεταξύ τους σημεία δεδομένων παρουσιάζουν συγκρίσιμες ιδιότητες ή τιμές. Κατά συνέπεια, οι k-κοντινότεροι γείτονες ενός νέου σημείου καθορίζουν είτε την πρόβλεψη είτε την ταξινόμηση της τιμής του.



Εικόνα 7: Απεικόνιση του k-Nearest Neighbors (k-NN). Η εικόνα παρουσιάζει την διαδικασία υλοποίησης του αλγορίθμου με την επιλογή k-κοντινότερων τιμών. [46]

Στρατηγική Υλοποίησης

Παρακάτω περιγράφεται βήμα βήμα η υλοποίηση του αλγορίθμου των K-Nearest Neighbors.

1. Καθορισμός του k

Η απόδοση της μεθόδου εξαρτάται σε μεγάλο βαθμό από τον αριθμό k των πλησιέστερων γειτόνων, ο οποίος επηρεάζει επίσης την ακρίβεια ταξινόμησης ή πρόβλεψης. Αρκετά κρίσιμα στοιχεία καθοδηγούν την επιλογή του αριθμού k:

- **Μικρός αριθμός k:** Δεδομένου ότι το αποτέλεσμα της μεθόδου θα εξαρτάται μόνο από έναν περιορισμένο αριθμό γειτόνων, οι οποίοι ενδέχεται να μην αντικατοπτρίζουν τη γενική τάση των δεδομένων, η προσέγγιση είναι πιο ευαίσθητη στο θόρυβο και τις ακραίες τιμές όταν ο αριθμός k είναι μικρός [6]. Για παράδειγμα, εάν k=1 ο αλγόριθμος προσδιορίζει την κατηγορία ή την τιμή του νέου σημείου μόνο με βάση το πλησιέστερο γειτονικό σημείο, οπότε σε συνθήκες θορύβου μπορεί να προκύψουν λανθασμένα αποτελέσματα.
- **Μεγάλος αριθμός k:** Η μέθοδος χάνει την ευαισθησία της στο θόρυβο και γίνεται πιο γενική όταν ο αριθμός k είναι μεγάλος. Η συμπερίληψη περισσότερων γειτόνων, ωστόσο, μπορεί να βοηθήσει στην εξομάλυνση της επιλογής και να μειώσει την ακρίβεια, καθώς μπορεί να συμπεριλάβει δεδομένα από απομακρυσμένες τοποθεσίες που δεν είναι τόσο ενδεικτικά για την κατηγορία ή την αξία του νέου σημείου [19].

2. Υπολογισμός Αποστάσεων

Ο υπολογισμός των αποστάσεων μεταξύ του νέου σημείου δεδομένων και κάθε σημείου στο σύνολο εκπαίδευσης βοηθάει στον προσδιορισμό των πλησιέστερων γειτόνων. Ο τύπος των δεδομένων και η φύση της εφαρμογής θα καθορίσουν τους πολυάριθμους τρόπους με τους οποίους μπορεί κανείς να υπολογίσει αυτές τις αποστάσεις. Οι συνήθεις μορφές των αποστάσεων είναι οι εξής:

- **Ευκλείδεια Απόσταση:** Η ευκλείδεια απόσταση είναι ο πιο συνηθισμένος τρόπος υπολογισμού της απόστασης μεταξύ δύο σημείων και υπολογίζεται με τον ακόλουθο τύπο:

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

όπου x_i και y_i είναι οι τιμές των χαρακτηριστικών για τα δύο σημεία x και y, και n είναι ο αριθμός των χαρακτηριστικών [49]. Η ευκλείδεια απόσταση είναι ιδανική όταν τα δεδομένα είναι συνεχόμενα και η διάταξη των σημείων στο χώρο είναι ομοιόμορφη.

- **Απόσταση Μανχάταν:** Η απόσταση Μανχάταν είναι το άθροισμα των απόλυτων διαφορών μεταξύ των χαρακτηριστικών σημείων:

$$\sum_{i=1}^{n}|x_i - y_i|^2$$

Αυτή η μέθοδος χρησιμοποιείται κυρίως όταν τα δεδομένα περιλαμβάνουν χαρακτηριστικά που δεν είναι συνεχόμενα ή όταν η διάταξη των σημείων είναι τέτοια ώστε οι ευθείες γραμμές δεν είναι οι καλύτεροι δείκτες για το πόσο κοντά είναι τα σημεία [6].

- **Συντελεστής Cosine:** Ο συντελεστής Cosine χρησιμοποιείται κυρίως για υπολογισμούς ομοιότητας μεταξύ διανυσμάτων σε προβλήματα κειμένου ή δεδομένων με υψηλή διάσταση. Υπολογίζεται ως:

$$\text{cosine(x,y)} = \frac{x \cdot y}{\|x\|\|y\|}$$

όπου x και y είναι τα διανύσματα των χαρακτηριστικών και ‖x‖ και ‖y‖ είναι οι νορμές των διανυσμάτων [50]. Ο συντελεστής Cosine μετράει ομοιότητα σε γωνίες μεταξύ των διανυσμάτων, γεγονός που τον καθιστά κατάλληλο για δεδομένα όπως το κείμενο.

3. Επιλογή Κοντινών Γειτόνων

Μετά τον υπολογισμό των αποστάσεων, επιλέγονται οι k πλησιέστεροι γείτονες ανάλογα με τις μικρότερες αποστάσεις. Για την αποτελεσματικότερη αναζήτηση των πλησιέστερων γειτόνων εφαρμόζονται μέθοδοι όπως KD-Tree Structure ή Cost-Based Search. Σχεδιασμένη ως ιεραρχική δομή δεδομένων, η δομή KD-Tree διατάσσει τα σημεία σε ένα δέντρο διευκολύνοντας έτσι την αποτελεσματική αναζήτηση των γειτονικών σημείων [47].

4. Απόφαση Κατηγορίας ή Τιμής

Η απόφαση για την κατηγορία ή την τιμή του νέου σημείου λαμβάνεται με βάση τους πλησιέστερους γείτονες:

- **Ταξινόμηση:** Με βάση την πλειοψηφία των k πλησιέστερων γειτόνων, το νέο σημείο εντάσσεται σε μια κατηγορία ταξινόμησης. Η σταθμισμένη ψηφοφορία μπορεί να εφαρμοστεί σε περιπτώσεις ισοψηφίας, σύμφωνα με την οποία οι ψήφοι των γειτόνων σταθμίζονται ανάλογα με την απόστασή τους από τη νέα θέση [48]. Συμβάλλοντας στην αποτροπή της ισοψηφίας, η σταθμισμένη ψηφοφορία εγγυάται ότι οι πλησιέστεροι γείτονες έχουν μεγαλύτερο αντίκτυπο στην τελική επιλογή.
- **Παλινδρόμηση:** Για την παλινδρόμηση η τιμή του νέου σημείου υπολογίζεται ως ο μέσος όρος των τιμών των k πλησιέστερων γειτόνων.

  Σε περιπτώσεις όπου οι τιμές των γειτόνων έχουν διαφορετικές κλίμακες ή μονάδες μέτρησης, χρησιμοποιείται ο **σταθμισμένος μέσος όρος**, όπου οι κοντινότεροι γείτονες έχουν μεγαλύτερο αντίκτυπο στην τελική πρόβλεψη [19].

Υλοποίηση και Παράδειγμα

Για να κατανοήσουμε καλύτερα τον αλγόριθμο, ας εξετάσουμε ένα παράδειγμα ταξινόμησης:

1. **Επιλογή k:** Επιλέγουμε έναν αριθμό k, π.χ. k=3.
2. **Υπολογισμός αποστάσεων:** Υπολογίζουμε την απόσταση μεταξύ του νέου σημείου και όλων των σημείων του συνόλου εκπαίδευσης.
3. **Αναγνώριση k πλησιέστερων γειτόνων:** Επιλέγουμε τους τρεις πλησιέστερους γείτονες.
4. **Απόφαση κατηγορίας:** Η κατηγορία του νέου σημείου καθορίζεται από την πλειοψηφία των κατηγοριών των 3 γειτόνων.

Για παράδειγμα, αν το νέο σημείο βρίσκεται κοντά σε 3 σημεία που ανήκουν στις κατηγορίες "Α" και "Β", η απόφαση μπορεί να γίνει με βάση την πλειοψηφία των κατηγοριών αυτών.

Ας δούμε αναλυτικά και με αριθμούς:

Δεδομένα Εκπαίδευσης:

Σημείο	Χαρακτηριστικό 1	Χαρακτηριστικό 2	Κατηγορία
A	1.0	2.0	Κόκκινο
B	1.5	1.5	Κόκκινο
C	5.0	6.0	Μπλε
D	6.0	5.5	Μπλε

Νέο Σημείο:

Χαρακτηριστικό 1	Χαρακτηριστικό 2
2.0	3.0

1. Υπολογισμός αποστάσεων:

- Απόσταση από Α: $\sqrt{(2.0 - 1.0)^2 + (3.0 - 2.0)^2} = \sqrt{1 + 1} = \sqrt{2} \approx 1.41$

- Απόσταση από B: $\sqrt{(2.0-1.5)^2+(3.0-1.5)^2} = \sqrt{0.25+2.25} = \sqrt{2.5} \approx 1.58$

- Απόσταση από C: $\sqrt{(2.0-5.0)^2+(3.0-6.0)^2} = \sqrt{9+9} = \sqrt{18} \approx 4.24$

- Απόσταση από D: $\sqrt{(2.0-6.0)^2+(3.0-5.5)^2} = \sqrt{16+6.25} = \sqrt{22.25} \approx 4.72$

2. Επιλογή k = 2 πλησιέστερων γειτόνων: A και B.

3. Απόφαση κατηγορίας: Και οι δύο γείτονες είναι της κατηγορίας "Κόκκινο". Επομένως, το νέο σημείο κατηγορείται ως "Κόκκινο".

Πλεονεκτήματα και Μειονεκτήματα

Όπως ισχύει σε όλους τους αλγορίθμους, έτσι και στο K-Nearest Neighbors, έχουμε πλεονεκτήματα και μειονεκτήματα, κάποια από τα οποία θα αναφερθούν παρακάτω.

Πλεονεκτήματα:

Κάποια από τα πλεονεκτήματα του αλγορίθμου είναι:

- **Απλότητα**: Η μέθοδος είναι «ελκυστική» καθώς είναι απλή στην κατανόηση και την εφαρμογή της χωρίς την ανάγκη δύσκολων διαδικασιών εκπαίδευσης.
- **Δυναμική Αντιμετώπιση**: Τα δεδομένα παραμένουν στη φυσική τους κατάσταση συνεπώς δεν είναι απαραίτητη η προ-εκπαίδευση [48].
- **Ευελιξία**: Μπορεί να χρησιμοποιηθεί τόσο για ταξινόμηση όσο και για παλινδρόμηση, γεγονός που τον καθιστά πολύ χρήσιμο σε ποικιλία εφαρμογών.

Μειονεκτήματα:

Μερικά από τα κυριότερα μειονεκτήματα του αλγορίθμου είναι:

- **Χρόνος Υπολογισμού**: Η αναζήτηση για τους πλησιέστερους γείτονες μπορεί να είναι χρονοβόρα, ειδικά σε μεγάλα σύνολα δεδομένων, καθώς συνεπάγεται τον υπολογισμό της απόστασης μεταξύ του νέου σημείου και κάθε σημείου εκπαίδευσης [51].
- **Προβλήματα Μνήμης**: Η μέθοδος αποθηκεύει ολόκληρο το σύνολο εκπαίδευσης, οπότε καταναλώνει πολύ μνήμη.
- **Ευαισθησία σε Θόρυβο**: Δεδομένα με θόρυβο ή ακραίες τιμές (outliers) μπορεί να επηρεάσουν σημαντικά τη μέθοδο.
- **Κλίμακες Χαρακτηριστικών**: Αν τα χαρακτηριστικά έχουν διαφορετικές κλίμακες, πρέπει να εφαρμοστεί κανονικοποίηση, ώστε να αποφευχθεί η κυριαρχία ορισμένων χαρακτηριστικών λόγω της μεγαλύτερης κλίμακας τους [52].

Με μεγάλη χρήση σε τομείς όπως η ανάλυση εικόνας, η ιατρική διάγνωση και η επεξεργασία φυσικής γλώσσας, η μέθοδος k-κοντινότερων γειτόνων είναι ένα ισχυρό και προσαρμόσιμο εργαλείο στον τομέα της μηχανικής μάθησης. Παρ' όλα αυτά, η αποτελεσματικότητά της εξαρτάται σε μεγάλο βαθμό από την ποιότητα των δεδομένων και την κατάλληλη επιλογή του k, καθώς και από τη σωστή επεξεργασία των χαρακτηριστικών.

2.3.8 Αλγόριθμος Naive Bayes

Ο αλγόριθμος Naive Bayes είναι μια ευρέως χρησιμοποιούμενη μέθοδος ταξινόμησης στον τομέα της μηχανικής μάθησης. Βασίζεται στο θεώρημα του Bayes, το οποίο συσχετίζει την πιθανότητα μιας κλάσης δεδομένων χαρακτηριστικών και χρησιμοποιεί την υπόθεση της ανεξαρτησίας μεταξύ των χαρακτηριστικών. Παρά την απλότητά του, ο αλγόριθμος Naive Bayes είναι ιδιαίτερα αποδοτικός και χρησιμοποιείται σε πολυάριθμες εφαρμογές, όπως η ανάλυση κειμένου, η επεξεργασία φυσικής γλώσσας και η ιατρική διάγνωση [54].

Θεώρημα του Bayes

Όπως προαναφέραμε, αλγόριθμος του Naive Bayes βασίζεται στο θεώρημα του Bayes, το οποίο παρέχει έναν τρόπο υπολογισμού της πιθανότητας μιας κατηγορίας **C** δεδομένων των χαρακτηριστικών **X**. Εκφράζεται ως:

$$P(C|X)= \frac{P(X|C)\cdot P(C)}{P(X)}$$

Όπου:

- P(C|X)) είναι η πιθανότητα της κλάσης C δεδομένων των χαρακτηριστικών X,
- P(X|C) είναι η πιθανότητα των χαρακτηριστικών X δεδομένης της κλάσης C,
- P(C) είναι η πιθανότητα της κλάσης C,
- P(X) είναι η πιθανότητα των χαρακτηριστικών X.

Αυτός ο τύπος υπολογίζει την **posterior** πιθανότητα της κλάσης C, η οποία επηρεάζεται από τις **prior** πιθανότητες και τις πιθανότητες των χαρακτηριστικών σε κάθε κατηγορία.

Υπόθεση Ανεξαρτησίας

Ο αλγόριθμος Naive Bayes βασίζεται στην υπόθεση ότι τα χαρακτηριστικά είναι **ανεξάρτητα** μεταξύ τους, δεδομένης της κατηγορίας. Αυτό σημαίνει ότι:

$$P(X|C) = P(x_1,x_2,\ldots,x_n|C)= \prod_{i=1}^{n} P(x_i|C)$$

όπου $\mathbf{X_i}$ είναι τα επιμέρους χαρακτηριστικά του δείγματος X και C είναι η κατηγορία.

Αυτή η υπόθεση μειώνει δραστικά την υπολογιστική πολυπλοκότητα, καθώς δεν χρειάζεται να υπολογίσουμε την αλληλεξάρτηση μεταξύ των χαρακτηριστικών [53]. Ωστόσο, αυτή η υπόθεση **είναι συνήθως υπερβολική** και μπορεί να μειώσει την ακρίβεια του αλγόριθμου, ειδικά όταν τα χαρακτηριστικά είναι έχουν σχέσεις μεταξύ τους και δεν είναι ανεξάρτητες.

Διαδικασία Εκπαίδευσης και Κατηγοριοποίησης

Η διαδικασία εκπαίδευσης του αλγορίθμου Naive Bayes περιλαμβάνει δύο βασικά βήματα:

1. **Υπολογισμός Πιθανοτήτων Κλάσης**: Υπολογίζεται η αρχική πιθανότητα για κάθε κλάση C. Η πιθανότητα καθορίζεται από την αναλογία των δειγμάτων που ταξινομούνται ως C προς το συνολικό αριθμό των δειγμάτων [54].
2. **Υπολογισμός Συνθηκών Πιθανοτήτων**: Για κάθε χαρακτηριστικό xi, υπολογίζεται η πιθανότητα του να εμφανιστεί το χαρακτηριστικό αυτό δεδομένης της κλάσης C. Αν τα χαρακτηριστικά είναι συνεχείς, χρησιμοποιούνται κατανομές, όπως η κανονική κατανομή, για την εκτίμηση αυτών των πιθανοτήτων.

Για την κατηγοριοποίηση ενός νέου δείγματος $X=(X_1,X_2,\ldots,X_n)$ υπολογίζεται η πιθανότητα κάθε κατηγορίας C με βάση το θεώρημα του Bayes:

$$P(C \mid X) = \frac{P(C) \cdot \prod_{i=1}^{n} P(x_i|C))}{P(X)}$$

Καθώς η P(X) είναι κοινή για όλες τις κατηγορίες, συνήθως παραλείπεται και η τελική κατηγοριοποίηση γίνεται με βάση την κατηγορία που δίνει τη μέγιστη πιθανότητα:

$$\hat{C} = \arg\max P(C) \cdot \prod_{i=1}^{n} P(x_i|C)$$

Παράδειγμα

Ας εξετάσουμε το παράδειγμα ταξινόμησης μηνυμάτων email ως "spam" ή "not spam" με βάση τις λέξεις που περιέχουν:

1. **Εκπαίδευση**:
   - Υπολογισμός Πιθανοτήτων Κλάσης: Αν έχουμε 100 email, από τα οποία τα 30 είναι "spam" και τα 70 "not spam", τότε:

     $$P(\text{spam}) = \frac{30}{100} = 0.3$$

     $$P(\text{not spam}) = \frac{70}{100} = 0.7$$

   - Υπολογισμός Συνθηκών Πιθανοτήτων: Αν η λέξη "προσφορά" εμφανίζεται σε 15 από τα συνολικά 30 email που είναι στην κατηγορία "spam" και σε 5 από τα 70 email που είναι στην κατηγορία "not spam", τότε:

     $$P(\text{προσφορά} \mid \text{spam}) = \frac{15}{30} = 0.5$$

     $$P(\text{προσφορά} \mid \text{not spam}) = \frac{5}{70} \approx 0.071$$

2. **Κατηγοριοποίηση**: Για ένα νέο email που περιέχει τη λέξη "προσφορά":

- Υπολογίζουμε τις παρακάτω πιθανότητες:
  $P(\text{spam}|\text{προσφορά}) \propto P(\text{spam}) \cdot P(\text{προσφορά}|\text{spam}) = 0.3 \cdot 0.5 = 0.15$
  $P(\text{not spam}|\text{προσφορά}) \propto P(\text{not spam}) \cdot P(\text{προσφορά}|\text{not spam}) = 0.7 \cdot 0.071 \approx 0.050$
- Η πιθανότητα ότι το email ανήκει στην κατηγορία "spam" είναι μεγαλύτερη, οπότε το email ταξινομείται ως "spam".

Πλεονεκτήματα και Μειονεκτήματα

Η μέθοδος Naive Bayes έχει αρκετά πλεονεκτήματα και μειονεκτήματα, τα σημαντικότερα από τα οποία απαριθμούνται παρακάτω.

Πλεονεκτήματα:

Κάποια πλεονεκτήματα του αλγορίθμου είναι:

- **Η Απλότητα**: Ο αλγόριθμος είναι απλός στην κατανόηση και την εκτέλεση [53].
- **Η Αποτελεσματικότητα**: Παρότι βασίζεται σε απλές υποθέσεις, μπορεί να αποδώσει καλά ακόμα και με μικρές ποσότητες δεδομένων [54].
- **Η Επεκτασιμότητα**: Ο Naive Bayes μπορεί να επεκταθεί εύκολα για να συμπεριλάβει πρόσθετες κατηγορίες και χαρακτηριστικά χωρίς να απαιτούνται περίπλοκοι υπολογισμοί [54].

Μειονεκτήματα:

Κάποια μειονεκτήματα του αλγορίθμου είναι:

- **Η Υπόθεση Ανεξαρτησίας**: Η υπόθεση της ανεξαρτησίας υποστηρίζει ότι τα χαρακτηριστικά σπάνια είναι ανεξάρτητα, μια πραγματικότητα που μπορεί να επηρεάσει την ακρίβεια του αλγορίθμου [53].
- **Η Περιορισμένη Ευελιξία**: Ο αλγόριθμος έχει μικρή ευελιξία, καθώς δεν λαμβάνει υπόψη τις αλληλεπιδράσεις μεταξύ των χαρακτηριστικών [54].

Ο Naive Bayes είναι ιδιαίτερα πλεονεκτικός σε εφαρμογές όπως η ανάλυση κειμένου, η ιατρική διάγνωση, η ανίχνευση απάτης και άλλα σενάρια όπου η ταχύτητα και η αποτελεσματικότητα είναι κρίσιμα χαρακτηριστικά [53].

2.3.9 Αλγόριθμος Gradient Boosting Machine (GBM)

Ο αλγόριθμος **Gradient Boosting Machine (GBM)** αποτελεί έναν ισχυρό μηχανισμό μηχανικής μάθησης που εντάσσεται στις μεθόδους **ενίσχυσης (boosting)**. Χρησιμοποιώντας συνήθως δέντρα αποφάσεων, η μέθοδος αυτή συγκεντρώνει διάφορα όχι και τόσο ισχυρά μοντέλα για να δημιουργήσει ένα ισχυρό μοντέλο ικανό να επιλύει αποτελεσματικά ζητήματα ταξινόμησης και παλινδρόμησης. Μέσω μιας διαδικασίας επαναληπτικής μάθησης, κατά την οποία κάθε νέο «υπομοντέλο» προσπαθεί να διορθώσει τα λάθη του προηγούμενου «υπομοντέλου» και ούτω καθεξής με έναν αλυσιδωτό τρόπο, ο πρωταρχικός στόχος της GBM είναι η βελτίωση των προβλέψεων [55].

Βασικές Αρχές Λειτουργίας του GBM

Η διαδικασία εκπαίδευσης του αλγορίθμου Gradient Boosting παίρνοντας τα βασικά στάδια λειτουργίας μπορεί να περιγραφεί όπως περιγράφεται παρακάτω:

1. Αρχική Πρόβλεψη (Initial Prediction)

Ο αλγόριθμος ξεκινά με την εκτίμηση μιας αρχικής πρόβλεψης, η οποία είναι συνήθως η μέση τιμή για τα προβλήματα παλινδρόμησης ή η πιο συχνή κατηγορία για τα προβλήματα ταξινόμησης. Η «ελάχιστη» διαφορά από τα πραγματικά δεδομένα εξακριβώνεται με τη χρήση της πρώτης πρόβλεψης. Αυτή η εκτίμηση αποτελεί τη βάση για τον χειρισμό των υπόλοιπων λαθών που θα διορθωθούν σε μεταγενέστερα στάδια από τα επόμενα υπομοντέλα [55].

2. Υπολογισμός Υπολειμμάτων (Residuals)

Στην επόμενη φάση, υπολογίζονται τα **υπολείμματα**, τα οποία είναι οι διαφορές μεταξύ των πραγματικών τιμών των εξαρτημένων μεταβλητών και των προβλέψεων του αρχικού μοντέλου. Ο υπολογισμός του υπολείμματος για κάθε δείγμα γίνεται ως εξής:

$$r_i = y_i - \hat{y}_i$$

όπου:

- r_i είναι το υπόλοιπο για το i-οστό δείγμα,
- y_i είναι η πραγματική τιμή του i-οστού δείγματος,
- $\hat{y}_i$ είναι η αρχική πρόβλεψη του i-οστού δείγματος.

Αυτή η διαδικασία αποκαλύπτει τα λάθη στις πρώτες προβλέψεις, τα οποία πρέπει να διορθωθούν στις επόμενες φάσεις, αν θέλουμε να έχουμε έναν πιο ισχυρό αλγόριθμο.

3. Εκπαίδευση Νέου Υπομοντέλου (Training a New Weak Learner)

Ένα νέο υπομοντέλο, συνήθως ένα **δέντρο απόφασης**, εκπαιδεύεται έτσι ώστε να προβλέπει τα υπολείμματα (τα σφάλματα) που υπολογίστηκαν στο προηγούμενο βήμα [55]. Αυτή η εκπαίδευση επιτρέπει στο νέο υπομοντέλο να επικεντρωθεί στα μέρη του μοντέλου με τα μεγαλύτερα λάθη. Κάθε υπομοντέλο προσπαθεί να διορθώσει τα λάθη και τις ελλείψεις του προηγούμενου.

4. Ενημέρωση Προβλέψεων (Updating Predictions)

Μετά την εκπαίδευση του νέου υπομοντέλου, οι προβλέψεις του συνολικού μοντέλου ενημερώνονται προσθέτοντας τις προβλέψεις του νέου υπομοντέλου. Οι νέες προβλέψεις του μοντέλου είναι το άθροισμα των προβλέψεων όλων των προηγούμενων υπομοντέλων. Ο ρυθμός μάθησης η, είναι ένας υπερπαράμετρος που ελέγχει τη συνεισφορά κάθε νέου υπομοντέλου στη συνολική πρόβλεψη [55]. Το τελικό μοντέλο, μετά από πολλαπλές επαναλήψεις, υπολογίζεται και εκφράζεται με την παρακάτω εξίσωση:

$$\hat{y}_{\text{final}} = \hat{y}_0 + \eta * \sum_{m=1}^{M} h_m(x)$$

όπου $\hat{y}_0$ είναι η αρχική πρόβλεψη, η είναι ο ρυθμός μάθησης, $h_m(x)$ είναι οι προβλέψεις του m-οστού υπομοντέλου, και M είναι ο συνολικός αριθμός των υπομοντέλων.

5. Επανάληψη (Iteration)

Η διαδικασία επαναλαμβάνεται για έναν προκαθορισμένο αριθμό βημάτων M ή μέχρι τα υπολείμματα (σφάλματα) να μειωθούν σημαντικά, δηλαδή μέχρι η βελτίωση στις προβλέψεις να γίνει αμελητέα. Κάθε επανάληψη επικεντρώνεται στη διόρθωση των λαθών που διαπράττουν τα προηγούμενα υπομοντέλα.

Υπερπαράμετροι του GBM

Η εκτέλεση του αλγορίθμου και η αποτελεσματικότητα του μοντέλου εξαρτώνται σε μεγάλο βαθμό από τις υπερπαραμέτρους [55]. Μεταξύ των πιο κρίσιμων υπερπαραμέτρων είναι αυτές όπως:

1. **Ρυθμός Μάθησης (Learning Rate)**: Η ένταση με την οποία κάθε νέο υπομοντέλο επηρεάζει τη γενική πρόβλεψη καθορίζεται από τον ρυθμό μάθησης **η**. Ενώ ένας χαμηλότερος ρυθμός μάθησης βελτιώνει τη γενίκευση, απαιτεί περισσότερες επαναλήψεις εκπαίδευσης για την επίτευξη των στόχων.
2. **Αριθμός Δέντρων (Number of Trees)**: Η υπερπαράμετρος ελέγχει την ποσότητα των δέντρων απόφασης που θα δημιουργήσει ο αλγόριθμος κατά τη διάρκεια της μάθησης. Αν και αυξάνουν τον κίνδυνο υπερπροσαρμογής, τα περισσότερα δέντρα συνήθως βελτιώνουν την απόδοση.
3. **Μέγιστο Βάθος Δέντρων (Tree Depth)**: Το βάθος των δέντρων καθορίζει το βαθμό πολυπλοκότητας κάθε δέντρου. Ενώ το μικρότερο βάθος οδηγεί συνήθως σε πιο γενικευμένα μοντέλα, το μεγαλύτερο βάθος μπορεί να προκαλέσει υπερπροσαρμογή.
4. **Ελάχιστος Αριθμός Δειγμάτων για Φύλλο (Minimum Samples per Leaf)**: Το ελάχιστο των δειγμάτων που απαιτούνται για την παραγωγή ενός φύλλου στο δέντρο απόφασης αποφασίζεται από αυτήν την υπερπαράμετρο. Αυτό επηρεάζει την ικανότητα γενίκευσης των δέντρων.

5. **Ελάχιστος Αριθμός Δειγμάτων για Διακλάδωση (Minimum Samples Split)**: Καθορίζει τον ελάχιστο απαιτούμενο αριθμό δειγμάτων για τη διάσπαση ενός κόμβου. Η κατάλληλη τιμή αυτής της παραμέτρου συμβάλλει στην αποφυγή υπερβολικής προσαρμογής.
6. **Υποδείγματα (Subsample)**: Ορίζει το ποσοστό των δειγμάτων που χρησιμοποιούνται για την εκπαίδευση κάθε δέντρου. Χρησιμοποιώντας μια υποομάδα των δεδομένων, αποφεύγεται η υπερπροσαρμογή και βελτιώνεται η γενίκευση του μοντέλου.

Πλεονεκτήματα και Αδυναμίες του GBM

Όπως όλοι οι αλγόριθμοι, υπάρχουν πλεονεκτήματα και μειονεκτήματα, ορισμένα από αυτά παρουσιάζονται παρακάτω.

Πλεονεκτήματα:

- **Υψηλή Απόδοση**: Η μέθοδος χρησιμοποιεί μια διαδικασία σταδιακής βελτίωσης που επικεντρώνεται στην εξάλειψη των λαθών των προηγούμενων μοντέλων, οπότε συνήθως αποδίδει εξαιρετικά αποτελέσματα για μια σειρά καταστάσεων.
- **Ευελιξία**: Μπορεί να κάνει χρήση διαφόρων ειδών μη ισχυρών μοντέλων και να διαχειριστεί τόσο δυσκολίες ταξινόμησης όσο και παλινδρόμησης.
- **Ανθεκτικότητα στα Χαμένα Δεδομένα**: Η μέθοδος GBM μπορεί να λειτουργήσει με ελλιπή σύνολα δεδομένων, δεδομένου ότι τα δέντρα αποφάσεων είναι ισχυρά στα δεδομένα που λείπουν.

Μειονεκτήματα:

- **Αργή Εκπαίδευση**: Η διαδικασία εκπαίδευσης μπορεί να είναι χρονοβόρα, ειδικά για μεγάλα σύνολα δεδομένων ή όταν ο αριθμός των δέντρων είναι μεγάλος.
- **Ευαισθησία στην Υπερπροσαρμογή**: Το μοντέλο μπορεί να υπερπροσαρμοστεί, αν δεν έχουμε σωστή ρύθμιση των υπερπαραμέτρων.
- **Δυσκολία στην Ερμηνεία**: Η ερμηνεία της πολυπλοκότητας των μοντέλων και η χρήση πολυάριθμων δέντρων καθιστά τα αποτελέσματα δύσκολα.

Εφαρμογές του GBM

Ο αλγόριθμος GBM χρησιμοποιείται σε πολλούς τομείς και έχει εφαρμογή σε πολλά πεδία, όπως:

- **Χρηματοοικονομικές Προβλέψεις**: Στην ανάλυση πιστωτικών κινδύνων, την πρόβλεψη μετοχών και τη χρηματοοικονομική ανάλυση [56].
- **Ανάλυση Κινδύνου και Ασφαλιστικά Σενάρια**: Στην εκτίμηση κινδύνου για ασφαλιστικές εταιρείες.
- **Επεξεργασία Κειμένου**: Στην ανάλυση συναισθημάτων και την ταξινόμηση κειμένων.
- **Ιατρική**: Στην πρόγνωση ασθενειών και τη διάγνωση με βάση ιατρικά δεδομένα.

2.3.10 Αλγόριθμος AdaBoost (Adaptive Boosting)

Ο αλγόριθμος AdaBoost (Adaptive Boosting) είναι μια ευρέως χρησιμοποιούμενη και αποτελεσματική μέθοδος για την ενίσχυση της απόδοσης αλγορίθμων ταξινόμησης, η οποία βασίζεται στην έννοια της ολοκληρωμένης μάθησης μέσω πολλών αδύναμων μαθητών. Η κύρια καινοτομία του AdaBoost είναι η ικανότητά του να ενισχύει την αποτελεσματικότητα ενός συστήματος ταξινόμησης δίνοντας δυναμικά έμφαση σε περιπτώσεις που είναι δύσκολο να εντοπιστούν με ακρίβεια, αυξάνοντας έτσι τη σημασία τους στην εκπαίδευση κάθε διαδοχικού μοντέλου. Η εξέλιξη αυτών των βημάτων μπορεί να αποδώσει έναν ισχυρό ταξινομητή, ακόμη και όταν κάθε αδύναμος ταξινομητής μεμονωμένα παρουσιάζει υποδεέστερη απόδοση [57].

Βασική Ιδέα του AdaBoost

Το AdaBoost έχει ως θεμελιώδη στόχο τη σταδιακή βελτίωση του μοντέλου με την ενσωμάτωση πολλών αδύναμων μοντέλων που εκπαιδεύονται με επαναληπτικό και προσαρμοστικό τρόπο. Ουσιαστικά, κάθε επόμενο μοντέλο επικεντρώνεται στα δεδομένα που τα προηγούμενα μοντέλα θεωρούν δύσκολα [59]. Ο αλγόριθμος AdaBoost τροποποιεί τα βάρη των δειγμάτων εκπαίδευσης, αυξάνοντας τη σημασία των λανθασμένα ταξινομημένων περιπτώσεων σε κάθε επόμενο στάδιο.

Αν και χρησιμοποιείται κυρίως για ταξινόμηση, η τεχνική εφαρμόζεται επίσης σε εφαρμογές όπως η πρόβλεψη και η ανάλυση σημάτων.

Βήματα του AdaBoost

Κάποια από τα βασικά βήματα του αλγορίθμου είναι περιγράφονται παρακάτω.

1. **Αρχικοποίηση Βαρών Δεδομένων**: Ο αλγόριθμος ξεκινά με το σύνολο δεδομένων $\{(x_1,y_1),(x_2,y_2),...,(x_N,y_N)\}$, όπου τα x_i είναι τα χαρακτηριστικά του κάθε παραδείγματος και τα yi οι ετικέτες (ταξινομήσεις). Στην πρώτη επανάληψη, όλα τα παραδείγματα έχουν το ίδιο βάρος:

$$w_i^{(1)}=\frac{1}{N}$$

   όπου $w_i^{(1)}$ είναι το βάρος του i-στου παραδείγματος κατά την πρώτη επανάληψη και N το συνολικό πλήθος των παραδειγμάτων.

2. **Εκπαίδευση Αδύναμου Μοντέλου**: Σε κάθε επανάληψη t του αλγορίθμου, εκπαιδεύεται ένας αδύναμος ταξινομητής h_t χρησιμοποιώντας τα δεδομένα και τα βάρη $w_i^{(t)}$. Στο σύνηθες παράδειγμα του AdaBoost, οι αδύναμοι ταξινομητές είναι συνήθως δέντρα απόφασης με περιορισμένο βάθος (γνωστά και ως "stumps"). Αυτοί οι ταξινομητές συνήθως είναι απλοί και έχουν απόδοση ελαφρώς καλύτερη από την τυχαία εικασία.

3. **Υπολογισμός Σφάλματος**: Στη συνέχεια, υπολογίζεται το σφάλμα του αδύναμου ταξινομητή h_t, το οποίο μετράει την αναλογία των παραδειγμάτων που ταξινομήθηκαν λανθασμένα:

$$\epsilon_t=\frac{\sum_{i:h_t(x_i)\neq y_i} w_i^{(t)}}{\sum_i w_i^{(t)}}$$

   Εδώ, το ϵ_t είναι το ποσοστό των παραδειγμάτων που ταξινομήθηκαν λανθασμένα από το h_t, και το κλάσμα εκφράζει την αναλογία των παραδειγμάτων με λάθος ταξινόμηση σε σχέση με τα συνολικά βάρη των παραδειγμάτων.

4. **Υπολογισμός Βάρους Μοντέλου**: Ο αδύναμος ταξινομητής h_t αποκτά ένα βάρος α_t, το οποίο εξαρτάται από το σφάλμα του ταξινομητή:

$$\alpha_t = \frac{1}{2} ln \left(\frac{1-\epsilon_t}{\epsilon_t}\right)$$

Αν το ϵ_t είναι μικρό, τότε το α_t είναι μεγάλο, πράγμα που σημαίνει ότι ο ταξινομητής έχει καλή απόδοση και έχει μεγαλύτερη επιρροή στο τελικό μοντέλο. Αντίθετα, αν το ϵ_t είναι κοντά στο 0.5, το α_t είναι μικρό, υποδηλώνοντας ότι ο ταξινομητής έχει χαμηλή απόδοση και η επιρροή του στο τελικό μοντέλο είναι περιορισμένη.

5. **Ενημέρωση Βαρών Δεδομένων**: Τα βάρη των παραδειγμάτων ενημερώνονται για την επόμενη επανάληψη, αυξάνοντας τα βάρη των παραδειγμάτων που ταξινομήθηκαν λανθασμένα και μειώνοντας τα βάρη των σωστά ταξινομημένων παραδειγμάτων:

$$w_i^{(t+1)} = w_i^{(t)} \cdot \exp(\alpha_t \cdot I(h_t(x_i) \neq y_i))$$

όπου $I(h_t(x_i) \neq y_i)$ είναι η ένδειξη (0 ή 1) που σηματοδοτεί αν το x_i ταξινομήθηκε σωστά ή λανθασμένα (0 αν ταξινομήθηκε σωστά και 1 αν ταξινομήθηκε λάθος). Στη συνέχεια, τα βάρη κανονικοποιούνται έτσι ώστε το άθροισμα τους να ισούται με 1, διασφαλίζοντας ότι η διαδικασία παραμένει σταθερή κατά τη διάρκεια των επαναλήψεων.

6. **Συνδυασμός Μοντέλων**: Στο τέλος, αφού ολοκληρωθούν όλες οι επαναλήψεις, οι αδύναμοι ταξινομητές συνδυάζονται για να δημιουργήσουν το τελικό μοντέλο. Η τελική απόφαση για ένα νέο παράδειγμα x γίνεται μέσω μιας βαρύτητας, η οποία βασίζεται στην ψήφο των ταξινομητών h_t, ζυγισμένη με τα βάρη τους:

$$H(x) = \text{sign}\left(\sum_{t=1}^{T} a_t h_t(x)\right)$$

όπου η συνάρτηση sign επιστρέφει την πρόβλεψη +1 ή −1 ανάλογα με το πρόσημο του αθροίσματος.

Βασικές Έννοιες

Κάποιες από τις πιο βασικές έννοιες που συναντάμε στον αλγόριθμο είναι τα παρακάτω.

- **Αδύναμο Μοντέλο (Weak Learner)**: Πρόκειται για έναν ταξινομητή που είναι ελαφρώς καλύτερος από την τυχαία εικασία. Στο AdaBoost, συνήθως χρησιμοποιούνται δέντρα απόφασης με περιορισμένο βάθος, γνωστά και ως "stumps".
- **Σφάλμα (ϵ_t)**: Ο λόγος των βαρών των λανθασμένα ταξινομημένων παραδειγμάτων προς το συνολικό βάρος όλων των παραδειγμάτων.
- **Βάρος Μοντέλου (α_t)**: Το μέτρο της σημασίας του κάθε αδύναμου ταξινομητή, που εξαρτάται από το σφάλμα του.
- **Βάρη Δεδομένων (w_i)**: Βάρη που χρησιμοποιούνται για την προσαρμογή της σημασίας των παραδειγμάτων κατά τη διάρκεια της εκπαίδευσης.

Πλεονεκτήματα και Μειονεκτήματα

Υπάρχουν κάποια πλεονεκτήματα αλλά και μειονεκτήματα του αλγορίθμου που πρέπει να αναφερθούν.

Πλεονεκτήματα:

- **Ευκολία Εφαρμογής**: Ο AdaBoost μπορεί να χρησιμοποιηθεί με διάφορους αδύναμους ταξινομητές (συμπεριλαμβανομένων των δέντρων απόφασης, Naive Bayes κ.λπ.) και δεν απαιτεί εξειδικευμένη διαμόρφωση.
- **Βελτίωση Απόδοσης**: Συνήθως ενισχύει σημαντικά την απόδοση του αλγορίθμου σε σύγκριση με μεμονωμένους ασθενείς ταξινομητές, καθιστώντας τον κατάλληλο για δύσκολα ή μη γραμμικά ζητήματα δεδομένων.

Μειονεκτήματα:

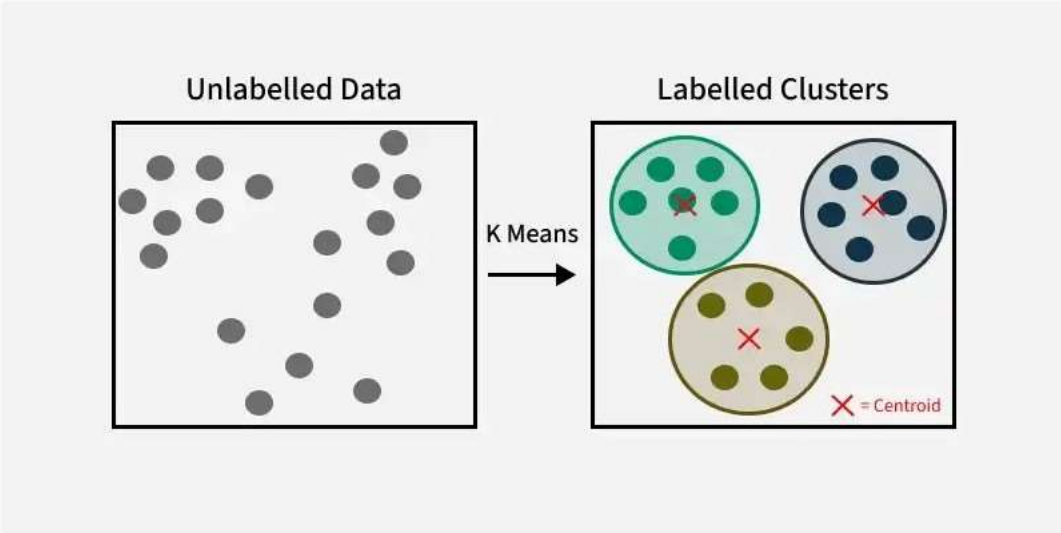
- **Ευαισθησία σε Θόρυβο**: Ο AdaBoost παρουσιάζει ευαισθησία σε θορυβώδη δεδομένα, καθώς η μέθοδος προσαρμογής βάρους μπορεί να οδηγήσει σε υπερπροσαρμογή όταν το σύνολο δεδομένων περιλαμβάνει πολλές λανθασμένες ή σοβαρές περιπτώσεις [58].

2.4 Αλγόριθμοι Μηχανικής Μάθησης στην Μη Επιβλεπόμενη Μάθηση

Η μη επιβλεπόμενη μάθηση (unsupervised learning) αποτελεί έναν σημαντικό τομέα της μηχανικής μάθησης, στον οποίο οι αλγόριθμοι εκπαιδεύονται χρησιμοποιώντας δεδομένα χωρίς ετικέτες (labels). Ο κύριος στόχος αυτής της προσέγγισης είναι να ανακαλυφθούν υποκείμενα μοτίβα ή δομές στα δεδομένα, χωρίς την ανάγκη καθοδήγησης από ανθρώπινο παράγοντα. Ο αλγόριθμος πρέπει να αναγνωρίσει και να εκμεταλλευτεί τις κρυμμένες σχέσεις ή ομαδοποιήσεις των δεδομένων. Χρησιμοποιείται σε πλήθος εφαρμογών, όπως η ομαδοποίηση (clustering), η μείωση διαστάσεων (dimensionality reduction), και η εκμάθηση παραγόντων (factorization) [19]. Παρακάτω, αναφέρονται και αναλύονται οι πιο γνωστοί και ευρέως χρησιμοποιούμενοι αλγόριθμοι μη επιβλεπόμενης μάθησης.

2.4.1 Αλγορίθμος K-Means

Ο αλγόριθμος K-Means είναι από τους πιο συχνά χρησιμοποιούμενους για την ομαδοποίηση δεδομένων, καθώς οργανώνει ένα σύνολο δεδομένων σε K ομάδες (clusters) ανάλογα με την ομοιότητα των παρατηρήσεων μέσα σε κάθε ομάδα. Η τεχνική επιδιώκει να μειώσει την απόσταση μεταξύ των δεδομένων και των κέντρων των ομάδων τους, έτσι ώστε τα δεδομένα σε κάθε ομάδα να είναι όσο το δυνατόν πιο συγκρίσιμα [63]. Η μέθοδος ως επί το πλείστον μετρά την ομοιότητα μεταξύ των δεδομένων και των κέντρων των ομάδων μέσω της ευκλείδειας απόστασης. Η ευκλείδεια απόσταση υπολογίζεται ως η ρίζα του αθροίσματος των τετραγώνων των επιμέρους μεταβολών των συντεταγμένων μεταξύ των σημείων δεδομένων.



Εικόνα 8: Απεικόνιση του αλγορίθμου k-Means. Η εικόνα παρουσιάζει την διαδικασία υλοποίησης του αλγορίθμου με την ομαδοποίηση των τιμών με ομοιότητες. [64]

Μέχρι να επιτευχθεί ένα κριτήριο τερματισμού -όπως η σύγκλιση των κέντρων της ομάδας ή η επίτευξη ενός καθορισμένου αριθμού επαναλήψεων- ο αλγόριθμος K-Means αποτελείται από πολυάριθμα βήματα που επαναλαμβάνονται μέχρι να επιτευχθεί το εν λόγω κριτήριο. Στη συνέχεια περιγράφονται οι φάσεις αυτές:

1. **Επιλογή Αρχικών Κέντρων (Initialization):** Στο πρώτο βήμα του αλγορίθμου, επιλέγονται τυχαία ή με κάποιον άλλο βελτιστοποιημένο τρόπο τα αρχικά κέντρα για τις ομάδες (centroids). Ο τρόπος επιλογής των αρχικών κέντρων μπορεί να επηρεάσει την απόδοση του αλγορίθμου, καθώς κακή επιλογή μπορεί να οδηγήσει σε τοπικά ελάχιστα ή αργή σύγκλιση. Μια πιο εξελιγμένη μέθοδος για την επιλογή των αρχικών κέντρων είναι η μέθοδος *K-Means*++, η οποία προσπαθεί να διαλέξει τα αρχικά κέντρα με τρόπο που να ελαχιστοποιεί την πιθανότητα κακής σύγκλισης [60].
2. **Ανάθεση σε Ομάδα (Assignment):** Κατά την ανάθεση, κάθε σημείο δεδομένων ανατίθεται στην ομάδα της οποίας το κέντρο είναι το πλησιέστερο, βάσει της ευκλείδειας απόστασης. Κάθε δεδομένο, επομένως, ανήκει στην ομάδα που έχει το κοντινότερο κέντρο, με βάση αυτή τη μέτρηση απόστασης. Η ανάθεση αυτή επαναλαμβάνεται για όλα τα σημεία δεδομένων.
3. **Ενημέρωση Κέντρων (Update):** Στο βήμα της ενημέρωσης, τα κέντρα των ομάδων ανανεώνονται. Κάθε κέντρο αντικαθίσταται από το μέσο όρο των δεδομένων που του ανατέθηκαν. Με αυτόν τον τρόπο, το κέντρο κάθε ομάδας προσαρμόζεται ώστε να αντικατοπτρίζει καλύτερα την "κεντρική" τάση των δεδομένων στην ομάδα του.
4. **Επανάληψη (Iteration):** Τα βήματα 2 και 3 επαναλαμβάνονται πολλές φορές. Ο αλγόριθμος συνεχίζει να αναθέτει τα δεδομένα στις ομάδες τους και να ενημερώνει τα κέντρα των ομάδων μέχρι να πληρωθεί κάποιο κριτήριο τερματισμού. Τα κριτήρια τερματισμού περιλαμβάνουν τη σύγκλιση των κέντρων ή την επίτευξη ενός προκαθορισμένου αριθμού επαναλήψεων.

Κριτήρια Τερματισμού

Ο αλγόριθμος K-Means διακόπτει τη λειτουργία του με την ικανοποίηση των ακόλουθων απαιτήσεων:

1. **Σύγκλιση (Convergence):** Όταν οι θέσεις των κέντρων των ομάδων δεν αλλάζουν πλέον ουσιαστικά από την προηγούμενη επανάληψη, δηλαδή τα κέντρα παραμένουν σταθερά. Αυτό υποδηλώνει ότι η μέθοδος έχει ανακαλύψει μια λύση με την οποία τα δεδομένα κατανέμονται μεταξύ των πιο σταθερών εφικτών κέντρων.
2. **Αριθμός Επαναλήψεων (Number of Iterations):** Ο αλγόριθμος μπορεί να έχει ένα προκαθορισμένο μέγιστο αριθμό επαναλήψεων, πέρα από τον οποίο σταματά, ανεξαρτήτως του αν έχει επιτευχθεί σύγκλιση. Συνήθως, αυτός ο περιορισμός βοηθάει στην αποφυγή υπερβολικά μεγάλων υπολογιστικών επιβαρύνσεων σε μεγάλες βάσεις δεδομένων.
3. **Αλλαγή στο Κριτήριο (Change in Criterion):** Όταν η αλλαγή στις θέσεις των κέντρων από την προηγούμενη επανάληψη είναι αμελητέα (π.χ. η μεταβολή είναι μικρότερη από κάποιο καθορισμένο κατώφλι), ο αλγόριθμος σταματά γιατί έχουμε μια σύγκλιση, υποδεικνύοντας ότι οι ομάδες είναι πλέον σταθερές.

Πλεονεκτήματα και Μειονεκτήματα

Ο αλγόριθμος K-Means έχει αρκετά πλεονεκτήματα, αλλά και μειονεκτήματα, κάποια από τα οποία αναφέρονται παρακάτω.

Πλεονεκτήματα:

1. **Απλότητα και Ευκολία Κατανόησης:** Ο αλγόριθμος είναι απλός και εύκολος στην κατανόηση, γεγονός που τον καθιστά δημοφιλή επιλογή για εφαρμογές όπου η κατηγοριοποίηση δεδομένων απαιτεί μια γρήγορη και αποτελεσματική λύση [61].
2. **Ταχύτητα Σύγκλισης:** Ιδιαίτερα χρήσιμη σε περιπτώσεις όπου οι υπολογιστικοί πόροι είναι περιορισμένοι, ο αλγόριθμος K-Means συγκλίνει γρήγορα τις περισσότερες φορές, ακόμη και σε μεγάλα σύνολα δεδομένων.
3. **Εφαρμοσιμότητα σε Μεγάλα Δεδομένα:** Η χαμηλή υπολογιστική πολυπλοκότητα της μεθόδου πληροί τις προϋποθέσεις για τεράστια σύνολα δεδομένων [62].

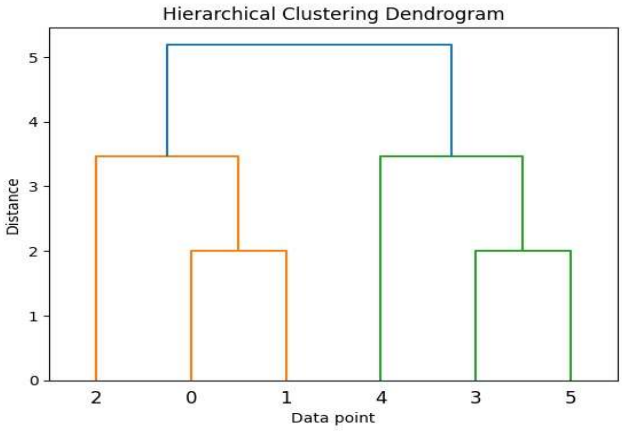
Μειονεκτήματα:

1. **Ευαισθησία στις Αρχικές Επιλογές:** Ο αλγόριθμος είναι ιδιαίτερα ευαίσθητος στην επιλογή των αρχικών κέντρων. Σε περίπτωση που τα αρχικά κέντρα είναι κακώς επιλεγμένα, μπορεί να ακολουθήσουν τοπικά ελάχιστα ή κακή σύγκλιση [60].
2. **Μη Ανθεκτικός σε Ανομοιόμορφες Ομάδες:** Ο αλγόριθμος δεν είναι ανθεκτικός σε ομάδες με πολύ διαφορετικά σχήματα ή πυκνότητες. Μπορεί να αντιμετωπίσει δυσκολίες όταν παράδειγμα οι ομάδες έχουν ανομοιογενή σχήματα ή όταν υπάρχουν πολλοί θόρυβοι στα δεδομένα [6].
3. **Απαιτεί Προκαθορισμένο Αριθμό Ομάδων:** Ο αλγόριθμος απαιτεί να γνωρίζουμε εκ των προτέρων τον αριθμό των ομάδων *K*, κάτι που μπορεί να είναι δύσκολο σε περιπτώσεις όπου δεν υπάρχει σαφής υπόθεση για τον αριθμό των κρυμμένων ομάδων στα δεδομένα.

Ιδιαίτερα στις περιπτώσεις που ο αριθμός των ομάδων είναι γνωστός, η απλότητα και η αποτελεσματικότητα του αλγορίθμου K-Means εξακολουθούν να αποτελούν παράγοντες δημοτικότητας μεταξύ των μεθόδων ομαδοποίησης δεδομένων. Παρόλα αυτά, η δομή των δεδομένων και η επιλογή των κέντρων εκκίνησης θα καθορίσουν κυρίως την αποτελεσματικότητά του. Άλλες τεχνικές ομαδοποίησης, όπως η DBSCAN, μπορεί να αποδώσουν ανώτερα αποτελέσματα σε περιπτώσεις που τα σχήματα των ομάδων είναι περίπλοκα ή οι πυκνότητες των ομάδων ποικίλλουν.

2.4.2 Ιεραρχική Ομαδοποίηση (Hierarchical Clustering)

Η Ιεραρχική Ομαδοποίηση (Hierarchical Clustering) είναι μια τεχνική μη-επιβλεπόμενης μάθησης που χρησιμοποιείται για την ανάλυση και κατηγοριοποίηση δεδομένων σε ομάδες, με σκοπό την αποκάλυψη της ιεραρχικής δομής τους. Η ιεραρχική ομαδοποίηση προσφέρει πιο ευέλικτες και πλούσιες αναπαραστάσεις δεδομένων από άλλες τεχνικές όπως ο αλγόριθμος K-Means, καθώς δεν απαιτεί μια προκαθορισμένη επιλογή του αριθμού των ομάδων. Όταν είναι άγνωστο πόσες ομάδες υπάρχουν στα δεδομένα ή όταν οι σχέσεις μεταξύ των ομάδων είναι ιεραρχικές, αυτό την καθιστά πολύ χρήσιμη [65].



Εικόνα 9: Απεικόνιση της Ιεραρχικής Ομαδοποίησης. Η εικόνα παρουσιάζει την υλοποίηση του αλγορίθμου με την κατηγοριοποίηση των δεδομένων σε ομάδες.[66]

Η διαδικασία της Ιεραρχικής Ομαδοποίησης περιλαμβάνει την κατασκευή ενός δέντρου ομαδοποίησης (dendrogram), το οποίο απεικονίζει την ιεραρχία των ομάδων. Τα σημεία δεδομένων που αρχικά εμφανίζονται ως μεμονωμένες ομάδες στην Ιεραρχική Ομαδοποίηση συνδυάζονται σταδιακά μέχρι να προκύψει μια ενιαία ομάδα. Αυτό βοηθά κάποιον να βρει τις συνδέσεις μεταξύ των δεδομένων σε διάφορα επίπεδα.

Οι αποφάσεις σχετικά με τον αριθμό των τελικών ομαδοποιήσεων (clusters) μπορούν να ληφθούν χρησιμοποιώντας αυτό το ιεραρχικό πλαίσιο [69].

Βασικές Έννοιες

1. **Μετρική Απόστασης**: Η μετρική απόστασης είναι ένας παράγοντας κλειδί για την Ιεραρχική Ομαδοποίηση, δεδομένου ότι ελέγχει τη μέτρηση της ομοιότητας ή της διαφοράς μεταξύ δύο σημείων δεδομένων. Εφαρμόζονται διάφορα μέτρα απόστασης με βάση τον τύπο των δεδομένων και τις ιδιότητές τους. Τα μέτρα απόστασης που χρησιμοποιούνται συνήθως είναι τα εξής:

    - **Η Ευκλείδεια Απόσταση**: Πρόκειται για τη «συνηθισμένη» απόσταση που χρησιμοποιείται για τη μέτρηση της απόστασης μεταξύ δύο σημείων σε έναν Ευκλείδιο χώρο. Είναι η ρίζα του αθροίσματος των τετραγώνων των διαφορών των συντεταγμένων των σημείων.

    - **Η Απόσταση Μανχάταν**: Λαμβάνοντας υπόψη μόνο την αθροιστική απόσταση στις οριζόντιες και κάθετες διαστάσεις, η απόσταση Μανχάταν υπολογίζει την «αντίστροφη» απόσταση μεταξύ δύο σημείων αντί της ευκλείδειας απόστασης.

    - **Η Ομοιότητα συνημιτόνου (Cosine Similarity)**: Χρησιμοποιείται συχνά σε εφαρμογές που σχετίζονται με την ανάλυση κειμένου, η ομοιότητα συνημιτόνου προσδιορίζει, στο πολυδιάστατο χώρο, την ομοιότητα μεταξύ δύο διανυσμάτων ανάλογα με τη γωνία τους.

2. **Κριτήριο Σύνδεσης (Linkage Criterion)**: Καθορίζει τον τρόπο υπολογισμού της απόστασης μεταξύ δύο ομάδων καθ' όλη τη διάρκεια της διαδικασίας συγχώνευσης μέσω ενός κριτηρίου σύνδεσης. Τα κριτήρια σύνδεσης έχουν πολλές μορφές και καθορίζουν τη «σχέση» μεταξύ των ομάδων.

    - **Ενιαία σύνδεση (Single Linkage)**: Βρίσκει την απόσταση που χωρίζει τα πλησιέστερα σημεία δύο ομάδων χρησιμοποιώντας απλή σύνδεση. Δύο ομάδες συγχωνεύονται όταν το πλησιέστερο ζεύγος των σημείων τους έχει τη μικρότερη απόσταση.

    - **Πλήρης σύνδεση (Complete Linkage)**: Υπολογίζει την απόσταση μεταξύ των πιο απομακρυσμένων σημείων δύο ομάδων σε πλήρη σύνδεση. Αυτό το είδος σχέσης δημιουργεί συνήθως ομαδοποιήσεις με ομοιόμορφα και πιο συμπαγή όρια.

    - **Μέση σύνδεση (Average Linkage)**: Υπολογίζει τη μέση απόσταση μεταξύ όλων των ζευγών σημείων από δύο ομάδες. Αυτή η προσέγγιση αποσκοπεί στο να συμπεριλάβει τις αποστάσεις κάθε σημείου και όχι μόνο των πλησιέστερων ή των πιο απομακρυσμένων [70].

Διαδικασία

Η διαδικασία της Ιεραρχικής Ομαδοποίησης περιλαμβάνει διάφορα βήματα που καθορίζουν πώς οι ομάδες δημιουργούνται και συγχωνεύονται:

1. **Αρχικοποίηση**: Κάθε παρατήρηση στο σύνολο δεδομένων θεωρείται αρχικά ως ξεχωριστή ομάδα.
2. **Υπολογισμός Αποστάσεων**: Η επιλεγμένη μετρική απόστασης καθορίζει τις αποστάσεις μεταξύ όλων των ομαδοποιήσεων.
3. **Συγχώνευση Ομάδων**: Δύο ομάδες που βρίσκονται πιο κοντά η μία στην άλλη συνδυάζονται σε μια μεγαλύτερη ομάδα. Για κάθε ομάδα, η διαδικασία αυτή επαναλαμβάνεται- κάθε φορά υπολογίζονται οι αποστάσεις για τις επόμενες ομάδες.
4. **Επανάληψη**: Η διαδικασία συγχώνευσης και επανυπολογισμού συνεχίζεται έως ότου παραχθεί μια ομάδα που περιλαμβάνει όλα τα δεδομένα.
5. **Δημιουργία Δέντρου Ομαδοποίησης (Dendrogram)**: Η ιεραρχία των ομάδων καταγράφεται σε ένα dendrogram (όπως φαίνεται και στην παραπάνω εικόνα), το οποίο αναπαριστά την ακολουθία συγχώνευσης των ομάδων. Το dendrogram παρέχει τη δυνατότητα στους χρήστες να «κόψουν» το δέντρο σε διαφορετικά επίπεδα για να επιλέξουν τον αριθμό των τελικών ομάδων που επιθυμούν να εξετάσουν [68].

Κριτήρια Τερματισμού

Η διαδικασία της Ιεραρχικής Ομαδοποίησης μπορεί να τερματίσει με διάφορους τρόπους, ανάλογα με τις απαιτήσεις της ανάλυσης:

1. **Συγκεκριμένος Αριθμός Ομάδων**: Η διαδικασία μπορεί να σταματήσει όταν φθάσει ένας προκαθορισμένος αριθμός ομάδων. Αυτό επιτρέπει στους χρήστες να αποφασίσουν τον απαιτούμενο αριθμό ομάδων για την εξέταση των πληροφοριών.
2. **Κατώφλι Απόστασης**: Ο τερματισμός της διαδικασίας μπορεί να προκύψει από την απόσταση μεταξύ των ομάδων που υπερβαίνει ένα καθορισμένο όριο. Αυτό επιτρέπει την αποφυγή υπερβολικής συγχώνευσης.
3. **Δημιουργία Δέντρου Σύνδεσης**: Η διαδικασία μπορεί να συνεχιστεί μέχρι να δημιουργηθεί πλήρως το δέντρο ομαδοποίησης (dendrogram), και ο τελικός αριθμός των ομάδων μπορεί να επιλεγεί από το dendrogram με βάση τα δεδομένα [69].

Πλεονεκτήματα και Μειονεκτήματα

Όπως σε όλους τους αλγορίθμους, έτσι και εδώ, έχουμε κάποια πλεονεκτήματα αλλά και κάποια μειονεκτήματα, κάποια από τα βασικά θα αναφερθούν παρακάτω.

Πλεονεκτήματα:

- **Χωρίς Προκαθορισμένο Αριθμό Ομάδων**: Η ιεραρχική ομαδοποίηση επιτρέπει στους χρήστες να βρουν τη δομή των δεδομένων χωρίς περιορισμούς που βασίζονται σε προκαθορισμένα κριτήρια, καθώς δεν απαιτεί προκαθορισμένο αριθμό ομάδων [67].
- **Ευέλικτη και Λεπτομερής Αναπαράσταση**: Η αναπαράσταση μέσω του dendrogram επιτρέπει μια πιο ευέλικτη κατηγοριοποίηση σε διάφορα επίπεδα ανάλυσης.
- **Ιδανική για Μικρά και Μεσαία Σύνολα Δεδομένων**: Τα σύνολα μικρών και μεσαίων δεδομένων όπου η υπολογιστική πολυπλοκότητα δεν είναι πολύ υψηλή βρίσκουν πολύ χρήσιμη την ιεραρχική ομαδοποίηση.

Μειονεκτήματα:

- **Υπολογιστικά Απαιτητική**: Η ιεραρχική ομαδοποίηση μπορεί να είναι υπολογιστικά απαιτητική για μεγάλα σύνολα δεδομένων, καθώς η υπολογιστική πολυπλοκότητα είναι συνήθως $O(n^2)$, όπου n είναι ο συνολικός αριθμός των σημείων δεδομένων [65].
- **Ανθεκτικότητα στον Θόρυβο**: Η Ιεραρχική Ομαδοποίηση δεν είναι ιδιαίτερα ανθεκτική στον θόρυβο και τα εκκρεμή δεδομένα, καθώς η παραμικρή αλλαγή στις αποστάσεις μπορεί να επηρεάσει σημαντικά την ιεραρχία των ομάδων [71].
- **Δυσκολία στην Επιλογή του Αριθμού Ομάδων**: Αν και το dendrogram παρέχει μια οπτική αναπαράσταση των συγχωνεύσεων, η επιλογή του βέλτιστου αριθμού ομάδων παραμένει υποκειμενική και εξαρτάται από την ερμηνεία των δεδομένων [70].

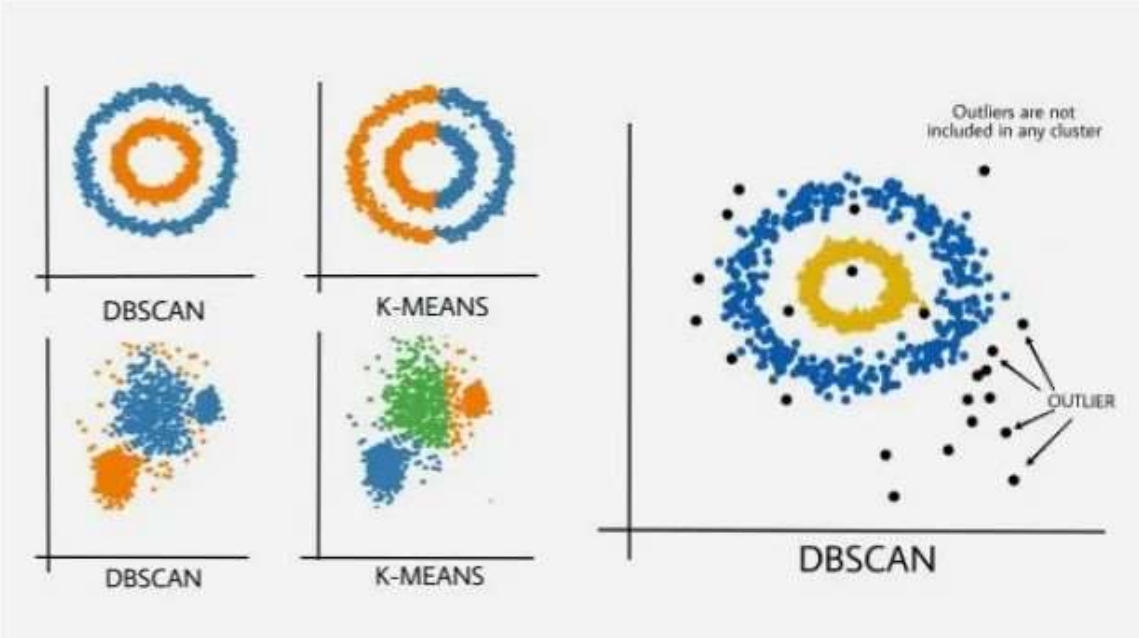
Ιδιαίτερα σε περιπτώσεις που ο αριθμός των ομάδων δεν μπορεί να είναι γνωστός εκ των προτέρων, η ιεραρχική ομαδοποίηση είναι μια εξαιρετική προσέγγιση για την ανάλυση της ιεραρχικής δομής των δεδομένων. Παρ' όλα αυτά, ειδικά σε μεγάλα και θορυβώδη σύνολα δεδομένων, οι ανάγκες επεξεργασίας και η ευαισθησία στο θόρυβο καθιστούν απαραίτητη την προσεκτική εφαρμογή της.

2.4.3 DBSCAN (Density-Based Spatial Clustering of Applications with Noise)

Ο αλγόριθμος **DBSCAN (Density-Based Spatial Clustering of Applications with Noise)** είναι μια τεχνική ομαδοποίησης δεδομένων που βασίζεται στην πυκνότητα και χρησιμοποιείται για την ανίχνευση περιοχών υψηλής πυκνότητας σε έναν χώρο δεδομένων. Η βασική αρχή του DBSCAN είναι ότι μια ομάδα δεδομένων (cluster) αντιπροσωπεύει μια περιοχή του χώρου όπου παρατηρείται υψηλή πυκνότητα σημείων, ενώ οι ομάδες διαχωρίζονται από περιοχές με χαμηλή πυκνότητα.

Ο DBSCAN έχει τη δυνατότητα να αναγνωρίσει και να χειριστεί περιοχές «θορύβου» ή «εκκρεμοτήτων», καθιστώντας τον χρήσιμο για δεδομένα που περιέχουν εξαιρέσεις ή ακανόνιστα σχήματα [72].

Η DBSCAN δεν εξαρτάται από τις καθορισμένες πληροφορίες σε αντίθεση με άλλες τεχνικές ομαδοποίησης, όπως η K-Means, που θέλουν τον προκαθορισμένο αριθμό ομάδων. Αντίθετα, σχηματίζει ομάδες χρησιμοποιώντας παραμέτρους πυκνότητας, επιτρέποντάς της επομένως να καθορίζει αυτόματα τον αριθμό των ομάδων και να χειρίζεται δεδομένα που δεν εμπίπτουν σε καμία ομάδα (θόρυβος).



Εικόνα 10: Απεικόνιση του DBSCAN. Η εικόνα παρουσιάζει την σύγκριση των αλγορίθμων DBSCAN και K-Means ως προς την διαφορετική ομαδοποίηση των τιμών. [73]

Βασικές Έννοιες

1. **Πυκνότητα (Density)**: Ο όρος «πυκνότητα» αναφέρεται στον αριθμό των σημείων δεδομένων που βρίσκονται εντός μιας καθορισμένης περιοχής γύρω από ένα σημείο (επιλεγμένο δείγμα). Η απόσταση (e) και ο αριθμός των σημείων εντός αυτής της περιοχής καθορίζουν την πυκνότητα. Οι υπολογισμοί της πυκνότητας βοηθούν στον εντοπισμό θέσεων με υψηλή συγκέντρωση σημείων, που θεωρούνται έτσι «ομάδες» (clusters) [72].
2. **Περιοχή Επίδρασης (εύρος) (Epsilon, ε)**: Η παράμετρος **Epsilon (ε)** ορίζει την ακτίνα ενός κυκλικού παραθύρου ή ενός υπερσφαιρικού όγκου γύρω από ένα σημείο. Η πυκνότητα γύρω από το σημείο υπολογίζεται με τη χρήση αυτού του φάσματος, το οποίο βοηθά επίσης στην εξακρίβωση του μεγέθους της «γειτονιάς» που εξετάζεται κατά την ομαδοποίηση.
3. **Ελάχιστος Αριθμός Δεδομένων (MinPts)**: Η παράμετρος **MinPts** καθορίζει τον ελάχιστο αριθμό γειτονικών σημείων (στο εσωτερικό της περιοχής Epsilon) που απαιτούνται για να θεωρηθεί ένα σημείο ως πυκνό σημείο (core point). Ο αριθμός των γειτονικών σημείων επηρεάζει την ευαισθησία του αλγόριθμου στη διάκριση μεταξύ ομάδων και θορύβου.

Διαδικασία

Η διαδικασία και τα βασικά βήματα του αλγορίθμου είναι τα εξής:

1. **Επιλογή Ενός Κεντρικού Δείγματος (Core Point)**: Η μέθοδος ξεκινά με την επιλογή ενός κεντρικού δείγματος του συνόλου δεδομένων, ενός σημείου δεδομένων.

Η τιμή Epsilon βοηθάει να προσδιορίσει κανείς τη σφαίρα επιρροής που περιβάλλει κάθε σημείο.

2. **Υπολογισμός Περιοχής Επίδρασης (Epsilon)**: Η περιοχή επίδρασης υπολογίζεται για το επιλεγμένο σημείο με βάση την τιμή του Epsilon, καθορίζοντας τα σημεία που βρίσκονται εντός αυτής της περιοχής.
3. **Εντοπισμός Περιοχής Πυκνότητας**: Στη συνέχεια, υπολογίζεται η πυκνότητα για όλα τα σημεία μέσα στην περιοχή επίδρασης, και αν ο αριθμός των σημείων που βρίσκονται σε αυτήν την περιοχή είναι μεγαλύτερος ή ίσος με το όριο MinPts, τότε το σημείο θεωρείται πυκνό (core point).
4. **Καθορισμός Κεντρικού Δείγματος ως Πυκνό (Core) ή Σύνορο (Border)**: Εάν το επιλεγμένο σημείο έχει τουλάχιστον MinPts γειτονικά σημεία εντός της περιοχής Epsilon, τότε χαρακτηρίζεται ως **πυκνό σημείο (core point)**. Αν έχει λιγότερα σημεία, τότε το σημείο αυτό θεωρείται **σύνορο (border)** και δεν μπορεί να επεκτείνει περαιτέρω την ομάδα.
5. **Σχηματισμός Ομάδων (Clusters)**: Ο αλγόριθμος συνεχίζει με την ένωση όλων των σημείων που είναι γειτονικά σε ένα πυκνό σημείο (core point), δημιουργώντας μια ομάδα. Εάν η ομάδα επεκτείνεται σε άλλες περιοχές πυκνότητας, η διαδικασία συνεχίζεται μέχρι να καλυφθούν όλα τα σημεία δεδομένων.
6. **Ετικέτα Noise (Θόρυβος)**: Τα σημεία που δεν ανήκουν σε καμία ομάδα και δεν πληρούν τα κριτήρια πυκνότητας ονομάζονται **θόρυβοι (noises)**. Αυτά τα σημεία είναι εκτός των περιοχών υψηλής πυκνότητας και δεν θεωρούνται μέλη καμίας ομάδας.

Πλεονεκτήματα και Μειονεκτήματα

Όπως και άλλοι αλγόριθμοι, έχουμε ορισμένα πλεονεκτήματα και μειονεκτήματα, μερικά από τα σημαντικά θα συζητηθούν εδώ.

Πλεονεκτήματα:

- **Αναγνώριση Ομάδων Με Ποικιλία Σχημάτων και Μεγεθών**: Ο DBSCAN είναι κατάλληλος για πιο περίπλοκες δομές δεδομένων, καθώς μπορεί να ανιχνεύσει ομάδες με μια σειρά από σχήματα, όχι μόνο στρογγυλά ή ελλειπτικά, όπως στον αλγόριθμο K-Means, γεγονός που καθιστά την αναγνώριση ομάδων πιο ποικίλη. [72].
- **Μη Ανάγκη για Προκαθορισμένο Αριθμό Ομάδων**: Σε αντίθεση με άλλους αλγόριθμους, ο DBSCAN δεν απαιτεί τον προκαθορισμένο αριθμό των ομάδων, κάνοντάς τον ιδιαίτερα ευέλικτο και ικανό να ανιχνεύει αυτόματα τον αριθμό των ομάδων.

- **Ανθεκτικότητα στον Θόρυβο και Εκκρεμότητες**: Ο αλγόριθμος είναι ανθεκτικός στον θόρυβο, καθώς μπορεί να αναγνωρίσει τα σημεία που δεν ανήκουν σε καμία ομάδα και να τα χαρακτηρίσει ως θόρυβο, αντί να τα ενσωματώσει σε μια ομάδα [74].

Μειονεκτήματα:

- **Ευαισθησία στις Παραμέτρους (Epsilon, MinPts)**: Η επιλογή των παραμέτρων Epsilon και MinPts έχει σημαντική επίδραση στην απόδοση του αλγόριθμου. Μικρές

αλλαγές στις παραμέτρους μπορούν να οδηγήσουν σε διαφορετικά αποτελέσματα, καθιστώντας τον ευαίσθητο στις ρυθμίσεις [72].

- **Υπολογιστικό Κόστος**: Για μεγάλα σύνολα δεδομένων, ο αλγόριθμος μπορεί να γίνει υπολογιστικά δαπανηρός, καθώς αναλύει τις αποστάσεις μεταξύ κάθε σημείου δεδομένων, απαιτώντας έτσι την υπολογιστική διαδικασία.
- **Δυσκολία στην Ανίχνευση Ομάδων με Διαφορετικές Πυκνότητες**: Ο DBSCAN δυσκολεύεται να ανιχνεύσει ομάδες που έχουν διαφορετικές πυκνότητες, καθώς οι παράμετροι Epsilon και MinPts ισχύουν ομοιόμορφα για όλο το σύνολο δεδομένων. Αυτό μπορεί να οδηγήσει σε εσφαλμένη ομαδοποίηση όταν οι ομάδες έχουν σημαντικές διαφορές στην πυκνότητά τους [74].

Η ανίχνευση ομάδων δεδομένων με βάση την πυκνότητα μπορεί να επιτευχθεί πραγματικά αποτελεσματικά με τον αλγόριθμο DBSCAN. Είναι ιδιαίτερα χρήσιμο όταν οι ομάδες έχουν ακανόνιστες μορφές ή όταν τα δεδομένα περιλαμβάνουν θόρυβο. Αν και έχει πλεονεκτήματα, ο σωστός ορισμός των παραμέτρων και ο χειρισμός δεδομένων με διάφορες πυκνότητες μπορεί να είναι δύσκολος.

2.4.4 PCA (Principal Component Analysis)

Η δημοφιλής τεχνική μείωσης των διαστάσεων που εφαρμόζεται ευρέως στη στατιστική ανάλυση και τη μηχανική μάθηση για την εξέταση και οπτικοποίηση τεράστιων και πολύπλοκων συνόλων δεδομένων είναι η Ανάλυση Κύριων Συνιστωσών (Principal Component Analysis - PCA). Ο πρωταρχικός στόχος της PCA είναι η μετατροπή των δεδομένων σε ένα νέο χώρο χαμηλότερων διαστάσεων, διατηρώντας παράλληλα τη μέγιστη γνώση σχετικά με τη διακύμανση των δεδομένων [75]. Βασικά, αναζητά τις γραμμικές συνιστώσες που εξηγούν τη μεγαλύτερη διακύμανση στα δεδομένα και τις χρησιμοποιεί για να μειώσει τη διάσταση, επιτρέποντας έτσι την ανάλυση σε ένα πιο συμπαγές και λογικό περιβάλλον.

Βασικές Έννοιες

1. **Κύριες Συνιστώσες (Principal Components - PCs)**: Οι νέοι άξονες που ορίζονται στο νέο χώρο αποτελούν τις κύριες συνιστώσες. Κάθε κύρια συνιστώσα ευθυγραμμίζεται με μια κατεύθυνση στο χώρο δεδομένων με τη μεγαλύτερη δυνατή διακύμανση. Η πρώτη κύρια συνιστώσα εξηγεί τη μεγαλύτερη διακύμανση, η δεύτερη την αμέσως μεγαλύτερη και ούτω καθεξής (Jolliffe, 2002). Η σημασία των κύριων συνιστωσών στη διευκρίνιση της διακύμανσης των δεδομένων καθοδηγεί τη διάταξή τους.
2. **Διακύμανση (Variance)**: Η διακύμανση μιας κύριας συνιστώσας δείχνει το ποσοστό της συνολικής διακύμανσης των δεδομένων που εξηγείται από την εν λόγω συνιστώσα. Συνήθως αυτές που εξηγούν τη μεγαλύτερη διακύμανση και συνεπώς είναι οι πιο κρίσιμες για τη μείωση της διακύμανσης είναι οι πρώτες κύριες συνιστώσες [76].

Διαδικασία

Ακολουθεί η διαδικασία και τα βασικά βήματα του αλγορίθμου:

1. **Κανονικοποίηση Δεδομένων**: Η κανονικοποίηση των δεδομένων είναι το αρχικό στάδιο στην εφαρμογή της PCA, αυτό είναι ζωτικής σημασίας όταν τα

χαρακτηριστικά των δεδομένων ποικίλλουν σε κλίμακα. Συνήθως, η κανονικοποίηση συνεπάγεται την κλιμάκωση των τιμών με την τυπική απόκλιση και τη μετακίνηση των μέσων τιμών κάθε χαρακτηριστικού στο μηδέν, ώστε να διασφαλιστεί ότι κάθε χαρακτηριστικό συμβάλλει εξίσου στον προσδιορισμό των κύριων συνιστωσών [75].

2. **Υπολογισμός Πίνακα Συνδιακύμανσης (Covariance Matrix)**: Ο πίνακας συνδιακύμανσης προσφέρει μια αναπαράσταση του τρόπου με τον οποίο τα χαρακτηριστικά των δεδομένων αλληλεπιδρούν μεταξύ τους και μετρά την αλληλεπίδρασή τους. Η συνδιακύμανση δύο χαρακτηριστικών δείχνει αν κινούνται προς την ίδια ή αντίθετη κατεύθυνση. Μεγαλύτερες συνδιακυμάνσεις υποδηλώνουν στενά συνδεδεμένες ιδιότητες.

3. **Υπολογισμός Κυρίων Συνιστωσών (Eigenvectors and Eigenvalues)**: Ο επόμενος υπολογισμός είναι τα **ιδιοδιανύσματα** (eigenvectors) και οι **ιδιοτιμές** (eigenvalues) του πίνακα συνδιακύμανσης. Τα ιδιοδιανύσματα καθορίζουν τις κατευθύνσεις του νέου χώρου (δηλαδή τις κύριες συνιστώσες), ενώ οι ιδιοτιμές υποδεικνύουν πόση διακύμανση εξηγείται από κάθε κατεύθυνση. Οι πιο σημαντικές κατευθύνσεις αντιστοιχούν στις μεγαλύτερες ιδιοτιμές, οι οποίες βοηθούν στην εξήγηση της μεγαλύτερης διακύμανσης των δεδομένων [75].

4. **Επιλογή Κύριων Συνιστωσών**: Οι κύριες συνιστώσες που επιλέγονται εξηγούν το μεγαλύτερο ποσοστό της συνολικής διακύμανσης των δεδομένων ανάλογα με τις ιδιοτιμές τους. Επιλέγονται οι συνιστώσες με τις υψηλότερες ιδιοτιμές- οι άλλες αγνοούνται, μειώνοντας έτσι τη διάσταση των δεδομένων.

5. **Δημιουργία Πίνακα Μετασχηματισμού (Transformation Matrix)**: Ο πίνακας μετασχηματισμού -που χρησιμοποιείται για τον πολλαπλασιασμό των αρχικών δεδομένων ώστε να προκύψει η νέα μειωμένη διάσταση, σχηματίζεται από τις επιλεγμένες κύριες συνιστώσες. Περαιτέρω έρευνα ή προεπεξεργασία για άλλες τεχνικές μηχανικής μάθησης μπορεί να επωφεληθεί από αυτό το αλλαγμένο περιβάλλον [76].

Πλεονεκτήματα και Μειονεκτήματα

Όπως και άλλοι αλγόριθμοι, έχουμε ορισμένα πλεονεκτήματα και μειονεκτήματα, μερικά από τα σημαντικά θα αναφερθούν παρακάτω.

Πλεονεκτήματα:

- **Μείωση Διάστασης**: Το PCA βελτιώνει την ερμηνευτική ικανότητα των δεδομένων μειώνοντας τη διάστασή τους, διατηρώντας έτσι τις σημαντικότερες πληροφορίες και αφαιρώντας το θόρυβο και τις ανεπιθύμητες αποκλίσεις.
- **Αντιμετώπιση Πολυκοινοτικότητας**: Σε σύνολα δεδομένων με υψηλή πολυκοινοτικότητα, το PCA βοηθά στην επίλυση των προβλημάτων που μπορεί να προκαλέσουν η συσχέτιση μεταξύ των χαρακτηριστικών.
- **Βελτίωση Άλλων Αλγορίθμων**: Πολλές άλλες τεχνικές μηχανικής μάθησης την χρησιμοποιούν ως εργαλείο προεπεξεργασίας, ενισχύοντας έτσι την απόδοση και επιταχύνοντας την εκτέλεση [75].

Μειονεκτήματα:

- **Δυσκολία στην Ερμηνεία**: Οι νέες διαστάσεις που προκύπτουν από το PCA είναι γραμμικοί συνδυασμοί των αρχικών χαρακτηριστικών, καθιστώντας δύσκολη την άμεση ερμηνεία τους [76].
- **Εξάρτηση από την Κανονικοποίηση**: Το PCA εξαρτάται ως επί το πλείστον από την κανονικοποίηση των δεδομένων, έτσι, η ακατάλληλη χρήση της ανάλυσης θα μπορούσε να οδηγήσει σε εσφαλμένα αποτελέσματα.
- **Μη Κατάλληλο για Μη Γραμμικές Σχέσεις**: Το PCA βασίζεται στη γραμμική υπόθεση των συσχετίσεων μεταξύ των χαρακτηριστικών, οπότε ενδέχεται να μην είναι χρήσιμη σε μη γραμμικές σχέσεις [75].

2.4.5 t-SNE (t-Distributed Stochastic Neighbor Embedding)

Συνήθως δισδιάστατο ή τρισδιάστατο, η **t-distributed stochastic neighbor embedding** (t-SNE) είναι μια τεχνική μείωσης των διαστάσεων της οποίας κύριος στόχος είναι να παρουσιάσει δεδομένα υψηλής διάστασης σε μικρότερους χώρους, διατηρώντας παράλληλα τους γεωμετρικούς δεσμούς μεταξύ των περιβαλλόντων δειγμάτων. Παρουσιάστηκε αρχικά από τους Laurens van der Maaten και Geoffrey Hinton και είναι ιδιαίτερα επιτυχής στο να τονίζει τις υποκείμενες δομές των δεδομένων προσφέροντας μια σαφή και προσιτή οπτική εικόνα [77].

Βασικές Έννοιες

1. **Πιθανότητα Κατανομής (Probability Distribution)**: Για κάθε ζεύγος δειγμάτων στον πολυδιάστατο χώρο, υπολογίζουμε κατανομές πιθανοτήτων για να δείξουμε την πιθανότητα το ένα δείγμα να επιλεγεί ως γείτονας του άλλου. Η απόσταση μεταξύ των δειγμάτων καθορίζει την πιθανότητα [77].

2. **Αντιστοίχιση Κατανομής (Distributional Matching)**: Στη συνέχεια, οι κατανομές στον χώρο χαμηλής διάστασης (2D ή 3D) συγκρίνονται με τις κατανομές του (υψηλής διάστασης) χώρου. Στόχος είναι η μείωση των αποκλίσεων μεταξύ των δύο κατανομών [77].

3. **Συνάρτηση Κόστους (Cost Function)**: Η συνάρτηση κόστους μετρά την απόκλιση μεταξύ των πιθανοτήτων γειτονίας στους δύο χώρους και ελαχιστοποιείται κατά τη διάρκεια της εκπαίδευσης του αλγορίθμου, ώστε οι σχέσεις γειτονίας να διατηρούνται όσο το δυνατόν καλύτερα.

Διαδικασία

Η διαδικασία και τα βασικά βήματα του αλγορίθμου είναι τα εξής:

1. **Υπολογισμός Πιθανοτήτων Γειτονίας**: Προσδιορίζουμε την πιθανότητα δύο δείγματα στον χώρο υψηλών διαστάσεων να είναι γείτονες για κάθε ζεύγος δειγμάτων. Υπολογίζεται με μια κατανομή Gauss, η πιθανότητα αυτή εξαρτάται από τις αποστάσεις των δειγμάτων.
2. **Συγκρίσεις Κατανομής στο Χαμηλό Χώρο**: Στο χώρο χαμηλής διάστασης, η πιθανότητα γειτονίας υπολογίζεται χρησιμοποιώντας μια κατανομή **t-distribution**.

Αυτό επιτρέπει την αποτελεσματική διατήρηση της δομής των δεδομένων ακόμη και σε περιπτώσεις μη γραμμικών δεδομένων.

3. **Ελαχιστοποίηση Συνάρτησης Κόστους**: Η συνάρτηση κόστους ελαχιστοποιείται για να βελτιωθεί η τοποθέτηση των σημείων στον νέο χώρο. Αυτό γίνεται μέσω αλγορίθμων βελτιστοποίησης, όπως ο **stochastic gradient descent** [77].

Πλεονεκτήματα και Μειονεκτήματα

Όπως κάθε αλγόριθμος, υπάρχουν ορισμένα οφέλη και μειονεκτήματα, μερικά από τα βασικά θα αναφερθούν παρακάτω.

Πλεονεκτήματα:

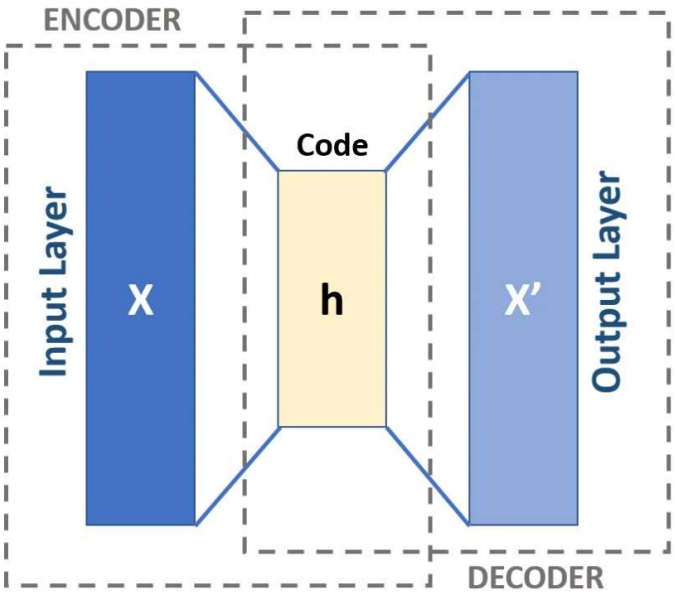
- **Διατήρηση Σχέσεων Γειτονίας**: Το t-SNE είναι εξαιρετικό στη διατήρηση των τοπικών δομών των δεδομένων και των γεωμετρικών σχέσεων.
- **Κατάλληλο για Οπτικοποίηση**: Ιδανικό για την οπτικοποίηση δεδομένων υψηλής διάστασης σε 2D ή 3D [77].
- **Ανθεκτικό στο Θόρυβο**: Είναι ανθεκτικό σε δεδομένα με θόρυβο, καθώς επικεντρώνεται στις σχέσεις μεταξύ κοντινών γειτόνων.

Μειονεκτήματα:

- **Απαιτεί Υπολογιστική Ικανότητα**: Η υπολογιστική ισχύς μπορεί να είναι υψηλή για μεγάλα σύνολα δεδομένων, καθώς το t-SNE δεν είναι βέλτιστο για ανάλυση μεγάλων κλιμάκων.
- **Ρύθμιση Υπερ-παραμέτρων**: Η τεχνική διαθέτει υπερπαραμέτρους, συμπεριλαμβανομένης της απόστασης γειτονιάς και του αριθμού των επαναλήψεων, οι οποίες χρειάζονται προσεκτική ρύθμιση για καλύτερα αποτελέσματα [77].

2.4.6 Αυτόματοι κωδικοποιητές (Autoencoders)

Οι αυτόματοι κωδικοποιητές (Autoencoders) είναι ένα είδος νευρωνικών δικτύων που χρησιμοποιούνται για την αυτόματη εξαγωγή χαρακτηριστικών ή για τη συμπίεση και αποκωδικοποίηση των δεδομένων, ανήκοντας στην κατηγορία των μη επιβλεπόμενων μεθόδων μηχανικής μάθησης. Οι αλγόριθμοι αυτοί δεν απαιτούν ετικέτες ή επισημάνσεις για την εκπαίδευσή τους, κάνοντάς τους ιδιαίτερα χρήσιμους σε περιβάλλοντα όπου οι ετικέτες είναι περιορισμένες ή ανύπαρκτες [37]. Ο αυτόματος κωδικοποιητής στοχεύει στην ανάπτυξη μιας χρήσιμης αναπαράστασης των δεδομένων εισόδου, μειώνοντας έτσι τη διάσταση των δεδομένων και εξάγοντας τα σημαντικότερα χαρακτηριστικά τους [78].



Εικόνα 11: Απεικόνιση του Αυτόματου Κωδικοποιητή. Η εικόνα παρουσιάζει έναν κωδικοποιητή-αποκωδικοποιητή με τις εισόδους και εξόδους. [79]

Βασικές Έννοιες

1. **Κωδικοποίηση (Encoding)**: Η διαδικασία της κωδικοποίησης στον αυτόματο κωδικοποιητή αναφέρεται στον μετασχηματισμό της εισόδου σε μια πιο συμπυκνωμένη και χαμηλής διάστασης αναπαράσταση, η οποία αποτελεί τη "κωδικοποίηση" των δεδομένων.

   Η κωδικοποίηση επιτρέπει στο δίκτυο να αντιληφθεί τις σημαντικές δομές και συνδέσεις δεδομένων χωρίς να απαιτείται πλήρης αναπαράσταση των δεδομένων [80]. Το υποσύστημα του δικτύου που ασχολείται με αυτή την απεικόνιση είναι ο κωδικοποιητής (encoder).
2. **Αποκωδικοποίηση (Decoding)**: Η ανακατασκευή των αρχικών δεδομένων από την κωδικοποιημένη μορφή είναι η διαδικασία της αποκωδικοποίησης. Το τμήμα του δικτύου που ορίζεται ως αποκωδικοποιητής, αποκωδικοποιεί για να ανακατασκευάσει την αρχική είσοδο με όσο το δυνατόν μεγαλύτερη ακρίβεια από την περιορισμένη και συμπιεσμένη κωδικοποίηση [82].
3. **Σφάλμα Ανακατασκευής (Reconstruction Error)**: Το σφάλμα ανακατασκευής ποσοτικοποιεί τη διαφορά μεταξύ της ανακατασκευής που προκύπτει από τον αποκωδικοποιητή και της αρχικής εισόδου. Συνήθως υπολογίζεται χρησιμοποιώντας μια συνάρτηση κόστους όπως τη μέση τετραγωνική απόκλιση (Mean Squared Error, MSE), το σφάλμα αυτό χρησιμεύει ως η κύρια απαίτηση εκπαίδευσης του μοντέλου. Ένα ελάχιστο αυτού του σφάλματος είναι ο στόχος της εκπαίδευσης μέσω της βέλτιστης ανακατασκευής δεδομένων [78].

Διαδικασία Εκπαίδευσης

Η εκπαίδευση ενός αυτόματου κωδικοποιητή σημαίνει ταυτόχρονη εκπαίδευση των δύο υποσυστημάτων του, του κωδικοποιητή και του αποκωδικοποιητή. Ενώ ο αποκωδικοποιητής προσπαθεί να ανακατασκευάσει τα δεδομένα από αυτή τη συμπυκνωμένη αναπαράσταση, ο κωδικοποιητής μαθαίνει κατά την εκπαίδευση να μεταφράζει την είσοδο σε μια τέτοια αναπαράσταση. Με στόχο τη μείωση αυτής της απόκλισης, η διαδικασία αυτή συνεχίζει τη βελτιστοποίηση μιας συνάρτησης κόστους που μετρά την ακρίβεια της ανακατασκευής [37].

Για να μειωθεί το σφάλμα ανακατασκευής σε κάθε εποχή εκπαίδευσης, η διαδικασία εκπαίδευσης συνίσταται στη χρήση μιας μεθόδου βελτιστοποίησης, όπως ο **stochastic gradient descent** (SGD), για την ενημέρωση των βαρών των νευρωνικών συνδέσεων [81].

Συμπίεση Χαρακτηριστικών (Feature Compression)

Η εκμάθηση μιας συμπιεσμένης αναπαράστασης των δεδομένων είναι ο θεμελιώδης στόχος του αυτόματου κωδικοποιητή- συμβάλλει στη διαχείριση της διατήρησης των πιο κρίσιμων πληροφοριών και στην απόρριψη των λιγότερο σημαντικών ή περιττών. Η εκπαίδευση του κωδικοποιητή παράγει αυτή τη διαδικασία, η οποία είναι γνωστή ως συμπίεση χαρακτηριστικών. Το δίκτυο ουσιαστικά εκπαιδεύεται ώστε να απεικονίζει τα δεδομένα σε μικρότερη διάσταση, μειώνοντας έτσι την απαιτούμενη χωρητικότητα αποθήκευσης και επεξεργασίας των πόρων [80]. Συχνά αναφέρεται ως κωδικοποίηση ή latent space, αυτή η μειωμένη διάσταση είναι η πιο κρίσιμη πτυχή για την αναπαράσταση των δεδομένων.

Πλεονεκτήματα και Μειονεκτήματα

Όπως σε όλους τους αλγορίθμους, έτσι και εδώ, έχουμε κάποια πλεονεκτήματα αλλά και κάποια μειονεκτήματα, κάποια από τα βασικά θα αναφερθούν παρακάτω.

Πλεονεκτήματα:

- **Συμπίεση και Ανάκτηση Χαρακτηριστικών**: Οι αυτόματοι κωδικοποιητές είναι αρκετά καλοί στο να εξάγουν σημαντικά στοιχεία από τα δεδομένα και να τα παρουσιάζουν σε συμπυκνωμένη μορφή, διατηρώντας έτσι τις πιο ζωτικές πληροφορίες. Η χρησιμότητά τους για χρήσεις όπως η συμπίεση δεδομένων και η αναγνώριση προτύπων απορρέει από αυτό [78].
- **Μη Επιβλεπόμενη Μάθηση**: Ιδανικές για περιπτώσεις όπου οι ετικέτες είναι λίγες ή ανύπαρκτες, δεν χρειάζονται ετικέτες για την εκπαίδευσή τους [37].
- **Ευελιξία**: Οι αυτόματοι κωδικοποιητές είναι ευέλικτοι σε διάφορες εφαρμογές, καθώς μπορούν να προσαρμοστούν σε διάφορα είδη δεδομένων, συμπεριλαμβανομένων εικόνων, κειμένου και ήχου.

Μειονεκτήματα:

- **Εξάρτηση από Υπερπαραμέτρους**: Η σωστή διαμόρφωση των υπερπαραμέτρων, όπως το μέγεθος της κωδικοποίησης, ο αριθμός των επιπέδων δικτύου και το είδος της συνάρτησης κόστους, καθορίζει την απόδοση των αυτόματων κωδικοποιητών κατά το σημαντικότερο μέρος [82].

- **Αποτυχία σε Υψηλής Διάστασης Δεδομένα**: Οι αυτόματοι κωδικοποιητές ενδέχεται να μην είναι εξαιρετικά αποτελεσματικοί στην αναπαραγωγή λεπτομερειών για δεδομένα υψηλής διάστασης, ιδίως όταν η ανακατασκευή απαιτεί την αποθήκευση ή την αναπαραγωγή πολύπλοκων δομών [80]. Παρόλο που μπορούν να συμπιέσουν τα δεδομένα.
- **Δυσκολία στην Ερμηνεία των Κωδικοποιήσεων**: Οι άνθρωποι συνήθως δυσκολεύονται να κατανοήσουν αμέσως την κωδικοποίηση που παράγεται από τον κωδικοποιητή, καθώς οι χαρακτηριστικές μεταβλητές του **latent space** δεν αντιστοιχούν σε εύκολα γνωστές έννοιες ή ιδιότητες της εισόδου [78].

Ιδανικοί για καταστάσεις μάθησης χωρίς επίβλεψη όταν δεν παρέχονται ετικέτες, οι αυτόματοι κωδικοποιητές είναι ισχυροί αλγόριθμοι για συμπίεση και αναπαράσταση δεδομένων. Αν και η αξία τους είναι μεγάλη, η αποτελεσματικότητά τους βασίζεται κυρίως στον κατάλληλο συντονισμό των υπερπαραμέτρων και στην ικανότητα του μοντέλου να αντιλαμβάνεται τα χαρακτηριστικά των δεδομένων σε περιβάλλοντα υψηλής διάστασης.

2.5 Αλγόριθμοι Μηχανικής Μάθησης στην Ενισχυτική Μάθηση

Στην Ενισχυτική Μάθηση (RL), οι αλγόριθμοι διδάσκονται να λαμβάνουν αποφάσεις μέσα σε ένα περιβάλλον για να μεγιστοποιήσουν κάποια συνολική χρησιμότητα ή ανταμοιβή. Μέσω της δοκιμής και του λάθους και της ανταμοιβής που λαμβάνει για τις δραστηριότητές του, ένας πράκτορας αποκτά κατανόηση των κατάλληλων ενεργειών για διάφορες περιστάσεις μέσω της επαφής του με το περιβάλλον. Όταν οι αποφάσεις πρέπει να λαμβάνονται διαδοχικά και τα αποτελέσματα των ενεργειών δεν είναι άμεσα ξεκάθαρα, αυτό το είδος μάθησης είναι ιδιαίτερα χρήσιμο. Οι πιο γνωστοί και εφαρμοσμένοι αλγόριθμοι παρατίθενται παρακάτω.

2.5.1 Q-Learning

Ένας από τους πιο συχνά χρησιμοποιούμενους αλγορίθμους Ενισχυτικής Μάθησης (Reinforcement Learning-RL), η μέθοδος Q-Learning είναι βασική για τη διδασκαλία ενός πράκτορα να επιλέγει τις καλύτερες ενέργειες μέσω της επαφής του με το περιβάλλον χωρίς προηγούμενη γνώση του περιβάλλοντος [83]. Η εκμάθηση βασίζεται σε μια συνάρτηση αξίας δράσης (Q(s,a)), η οποία προσεγγίζει την υψηλότερη αναμενόμενη ανταμοιβή που μπορεί να λάβει ο αναγνωριστής ξεκινώντας από την κατάσταση s, ενεργώντας σύμφωνα με αυτήν και ακολουθώντας τη βέλτιστη στρατηγική.

Θεμελιώδη Στοιχεία και Διαδικασία

Στη συνέχεια περιγράφονται ορισμένα από τα θεμελιώδη στοιχεία και η προσέγγιση του αλγορίθμου που ακολουθείται.

1. **Ορισμός του Χώρου Καταστάσεων και Ενεργειών**

   Ενώ ο χώρος δράσης (A) περιλαμβάνει όλες τις δραστηριότητες που μπορεί να εκτελέσει ο πράκτορας σε κάθε κατάσταση, ο χώρος καταστάσεων (S) περιλαμβάνει το σύνολο των πιθανών καταστάσεων του περιβάλλοντος. Σε ένα πρόβλημα

πλοήγησης, για παράδειγμα, ο χώρος καταστάσεων S μπορεί να περιλαμβάνει όλες τις πιθανές θέσεις στο χώρο, ενώ ο χώρος δράσης A μπορεί να δείχνει κινήσεις όπως «πάνω», «κάτω», «αριστερά», «δεξιά» [9].

2. **Αρχικοποίηση του Q-Table**
   Κάθε γραμμή στον δισδιάστατο πίνακα Q (Q(s,a)) σχετίζεται με μια κατάσταση, ενώ κάθε στήλη αναφέρεται σε μια ενέργεια. Οι τιμές του Q(s,a) μπορούν αρχικά να οριστούν είτε τυχαία είτε ως ισοδύναμες με το μηδέν. Καθώς οι αλληλεπιδράσεις με το περιβάλλον αυξάνονται, ο πίνακας αυτός αλλάζει κατά τη διάρκεια της μάθησης και συγκλίνει στις ιδανικές τιμές [83].

3. **Αλγόριθμος Επαναληπτικής Εκπαίδευσης**
   Η εκπαίδευση του πράκτορα βασίζεται σε επαναλαμβανόμενες αλληλεπιδράσεις με το περιβάλλον και περιλαμβάνει τα εξής βήματα:
   - **Επιλογή Ενέργειας**: Ο πράκτορας επιλέγει μια ενέργεια a σε μια κατάσταση s, είτε τυχαία (εξερεύνηση) είτε βάσει των υπαρχόντων Q(s,a) (εκμετάλλευση). Η στρατηγική ϵ είναι ιδιαίτερα δημοφιλής, καθώς επιτρέπει έναν ισορροπημένο συνδυασμό εξερεύνησης και εκμετάλλευσης [9].
   - **Εκτέλεση Ενέργειας και Παρατήρηση Ανταμοιβής**: Με την εκτέλεση της ενέργειας, ο πράκτορας μεταβαίνει σε μια νέα κατάσταση s′ και λαμβάνει μια ανταμοιβή r, που εξαρτάται από την ποιότητα της ενέργειας στο συγκεκριμένο περιβάλλον.
   - **Ενημέρωση Q-Value**: Ο πίνακας Q(s,a) ενημερώνεται με τη χρήση της εξίσωσης:

     $$Q(s, \alpha) \leftarrow Q(s, \alpha) + \alpha * (r + \gamma * \max_{\alpha'} Q(s', \alpha') - Q(s, \alpha))$$

     όπου:

     - α: Ρυθμός μάθησης, καθορίζει πόσο γρήγορα προσαρμόζονται οι τιμές στις νέες παρατηρήσεις [83].
     - r: Άμεση ανταμοιβή από την ενέργεια a.
     - γ: Παράγοντας εκπτωτικής αξίας, ρυθμίζει τη σημασία των μελλοντικών ανταμοιβών [9].
     - $\max_{a'} Q(s',a')$: Η καλύτερη δυνατή εκτίμηση της μελλοντικής ανταμοιβής για τη νέα κατάσταση s′.
   - **Επανάληψη Διαδικασίας**: Η διαδικασία συνεχίζεται για μεγάλο αριθμό επεισοδίων, με κάθε επεισόδιο να διαρκεί μέχρι να φτάσει ο πράκτορας σε κατάσταση τερματισμού ή να εκπνεύσει ο μέγιστος αριθμός βημάτων [83].

Σύγκλιση και Βέλτιστη Πολιτική

Ο πίνακας Q(s,a) συγκλίνει στις βέλτιστες τιμές υπό τις εξής προϋποθέσεις:

1. Ο πράκτορας εξερευνά επαρκώς όλες τις καταστάσεις και ενέργειες (S και A).

2. Ο ρυθμός μάθησης α μειώνεται σταδιακά, ώστε να διασφαλιστεί ότι οι τιμές δεν διαταράσσονται από νέες παρατηρήσεις.

Η βέλτιστη πολιτική π^* προκύπτει από την επιλογή της ενέργειας με τη μέγιστη τιμή $Q(s,a)$ για κάθε κατάσταση s:

$$\pi^*(s) = \text{argma}\chi_\alpha Q(s, a)$$

Αυτή η πολιτική οδηγεί τον πράκτορα στη μεγιστοποίηση της συνολικής ανταμοιβής του, ακόμα και σε περιβάλλοντα με στοχαστική συμπεριφορά [9].

Πλεονεκτήματα και Μειονεκτήματα

Υπάρχουν όπως σε όλους τους αλγορίθμους κάποια πλεονεκτήματα αλλά και μειονεκτήματα, τα πιο βασικά από τα οποία αναφέρονται παρακάτω.

Πλεονεκτήματα

1. **Ανεξαρτησία από το Περιβάλλον**: Ο αλγόριθμος Q-Learning μπορεί να χρησιμοποιηθεί σε περιβάλλοντα όπου οι ανταμοιβές ή οι μεταβάσεις καταστάσεων είναι άγνωστες εκ των προτέρων [83].
2. **Ευελιξία**:Από τη ρομποτική και τον έλεγχο συστημάτων έως τα παιχνίδια και τις προσομοιώσεις, βρίσκει χρησιμότητα σε μια σειρά από πεδία [84].
3. **Θεωρητική Εγγύηση Σύγκλισης**: Υπό τις κατάλληλες συνθήκες, ο αλγόριθμος συγκλίνει στη βέλτιστη λύση [83].

Μειονεκτήματα

1. **Curse of Dimensionality**: Καθώς οι καταστάσεις ή οι ενέργειες αυξάνονται, ο πίνακας Q(s,a) γίνεται αρκετά μεγάλος.
2. **Αργή Σύγκλιση**: Η σύγκλιση μπορεί να είναι κάπως αργή σε ρυθμίσεις με περίπλοκα κίνητρα ή μεγάλους χώρους καταστάσεων.
3. **Διαχείριση Στοχαστικών Περιβάλλοντων**: Η αξιόπιστη συλλογή δεδομένων σε ιδιαίτερα στοχαστικά πλαίσια απαιτεί εκτεταμένη έρευνα.

2.5.2. SARSA (State-Action-Reward-State-Action)

Ο αλγόριθμος **SARSA (State-Action-Reward-State-Action)** είναι ένας αλγόριθμος ενισχυτικής μάθησης (Reinforcement Learning, RL) που εφαρμόζεται για την εκμάθηση βέλτιστων στρατηγικών δράσης σε περιβάλλοντα με αβεβαιότητα. Σε αντίθεση με τον Q-Learning, ο αλγόριθμος SARSA είναι **on-policy**, δηλαδή η ενημέρωση των τιμών Q βασίζεται στην πραγματική πολιτική που ακολουθεί ο πράκτορας κατά τη διάρκεια της εκπαίδευσης [9].

Θεωρητικό υπόβαθρο και Στόχοι

Ο στόχος του SARSA είναι να εκτιμήσει τις **τιμές κατάστασης-ενέργειας (Q-values)** για κάθε ζεύγος κατάστασης s και ενέργειας a, ώστε ο πράκτορας να μπορεί να αποφασίζει

βέλτιστες ενέργειες που μεγιστοποιούν τη μακροπρόθεσμη αθροιστική ανταμοιβή. Η διαδικασία αυτή διατυπώνεται μέσω του **προβλήματος ελέγχου του Markov Decision Process (MDP)**, όπου η δυναμική του περιβάλλοντος περιγράφεται από μεταβάσεις καταστάσεων και ανταμοιβές [86].

Κύρια Βήματα του SARSA

Κάποια από τα βασικά βήματα υλοποίησης του αλγορίθμου περιγράφονται παρακάτω.

1. **Ορισμός του Χώρου Καταστάσεων και Ενεργειών**

   Οριοθετείται το σύνολο των πιθανών καταστάσεων S και των δράσεων A που είναι διαθέσιμες στον πράκτορα. Οι καταστάσεις υποδηλώνουν τις υπάρχουσες συνθήκες του συστήματος, ενώ οι δράσεις οριοθετούν τις δυνητικές επιλογές που είναι διαθέσιμες στον πράκτορα. Το περιβάλλον τηρεί την πιθανότητα μετάβασης $P(s'|s,a)$, η οποία περιγράφει την πιθανότητα μετάβασης στην κατάσταση s' κατά την εκτέλεση της ενέργειας an στην κατάσταση s [9].

2. **Αρχικοποίηση του Q-Table**

   Δημιουργείται ένας πίνακας Q(s,a), με κάθε κελί να αντιπροσωπεύει την αναμενόμενη τιμή των μελλοντικών ανταμοιβών για το ζεύγος (s,a). Οι αρχικές τιμές του πίνακα Q συχνά μηδενίζονται ή ανατίθενται τυχαία.

3. **Επαναληπτική Διαδικασία Εκπαίδευσης**

   Η εκπαίδευση πραγματοποιείται μέσω επαναλαμβανόμενων επεισοδίων (episodes), όπου κάθε επεισόδιο περιλαμβάνει τις εξής ενέργειες:
   - **Επιλογή ενέργειας στην τρέχουσα κατάσταση:** Ο πράκτορας επιλέγει μια ενέργεια a στην τρέχουσα κατάσταση s με βάση μια στρατηγική, όπως η ϵ, που εξισορροπεί την εξερεύνηση (exploration) με την εκμετάλλευση (exploitation) [83].
   - **Εκτέλεση ενέργειας και παρατήρηση ανταμοιβής:** Ο πράκτορας εκτελεί την ενέργεια a και παρατηρεί την άμεση ανταμοιβή r και τη νέα κατάσταση s'.
   - **Επιλογή επόμενης ενέργειας:** Στη νέα κατάσταση s', επιλέγεται μια επόμενη ενέργεια a' βάσει της ίδιας στρατηγικής.
   - **Ενημέρωση των τιμών Q:** Οι τιμές Q(s,a) ενημερώνονται με την εξίσωση SARSA: $Q(s,a) \leftarrow Q(s,a)+\alpha\cdot[r+\gamma\cdot Q(s',a')-Q(s,a)]$

     Όπου:

     - α: είναι ο Ρυθμός μάθησης (learning rate), που καθορίζει την ταχύτητα προσαρμογής των Q-values.
     - γ: είναι ο Παράγοντας εκπτωτικής αξίας (discount factor), που ορίζει τη βαρύτητα των μελλοντικών ανταμοιβών.
     - Q(s,a): είναι η Τρέχουσα εκτίμηση της τιμής για το ζεύγος (s,a).
     - r: Η ανταμοιβή που λαμβάνεται μετά την εκτέλεση της ενέργειας a.

4. **Επανάληψη της Διαδικασίας**

   Οι προαναφερθείσες διαδικασίες επαναλαμβάνονται έως ότου ο πίνακας Q συγκλίνει, δηλαδή οι τιμές του παύουν να μεταβάλλονται αισθητά [87].

Ιδιαιτερότητες του SARSA

- **On-policy Learning:** Ο αλγόριθμος SARSA βασίζεται στην πολιτική που τηρείται καθ' όλη τη διάρκεια της εκπαίδευσης. Ο SARSA λαμβάνει υπόψη του την ενέργεια που πραγματικά επιλέγεται, σε αντίθεση με την Q-Learning, η οποία αλλάζει τις τιμές Q με βάση την εκτιμώμενη καλύτερη ενέργεια.
- **Ευαισθησία στην επιλογή πολιτικής:** Η απόδοση του αλγορίθμου SARSA επηρεάζεται από την εκτελούμενη πολιτική, τη μέθοδο εξερεύνησης και τις περιβαλλοντικές αλληλεπιδράσεις.

Εφαρμογές

Ο αλγόριθμος SARSA χρησιμοποιείται εκτενώς σε σενάρια όπου η ασφάλεια ή η σταθερότητα είναι υψίστης σημασίας, όπως:

- **Αυτόνομη πλοήγηση ρομπότ:** Εκμάθηση ασφαλών διαδρομών σε περιβάλλοντα με εμπόδια.
- **Ενεργειακά συστήματα:** Βελτίωση της αξιοποίησης των πόρων.
- **Προσομοιώσεις και παιχνίδια:** Ρεαλιστική εκπαίδευση πρακτόρων σε δυναμικά περιβάλλοντα [9].

2.5.3 Deep Q-Networks (DQN)

Συνδυάζοντας τη μέθοδο Q-Learning, έναν βασικό αλγόριθμο ενισχυτικής μάθησης, με την ικανότητα των βαθιών νευρωνικών δικτύων να αντιμετωπίζουν θέματα υψηλής πολυπλοκότητας, ο αλγόριθμος Deep Q-Networks (DQN) Παρουσιάστηκε από τον Mnih και τους συνεργάτες του και αποτέλεσε επανάσταση στον τομέα της ενισχυτικής μάθησης, καθώς έδειξε πόσο καλά μπορούσε να μάθει περίπλοκες στρατηγικές με βάση μόνο τα ακατέργαστα δεδομένα [88].

Θεωρητικό Υπόβαθρο και Σκοπός

Στο Q-Learning, η συνάρτηση Q(s,a) εκτιμά τη μέγιστη αναμενόμενη συνολική ανταμοιβή όταν ο πράκτορας βρίσκεται σε κατάσταση s, εκτελεί την ενέργεια a και ακολουθεί την πολιτική π. Η εξίσωση Bellman ορίζει την αναδρομική σχέση:

$$Q(s,a) = r + \gamma \max_{a'} Q(s',a'),$$

όπου:

- r: η άμεση ανταμοιβή.
- γ: ο παράγοντας έκπτωσης (discount factor).

- s': η επόμενη κατάσταση.
- a': η επόμενη ενέργεια.

Το πρόβλημα με το κλασικό Q-Learning είναι ότι απαιτεί αποθήκευση του πίνακα Q, που είναι πρακτικά αδύνατο για μεγάλους ή συνεχείς χώρους καταστάσεων. Ο αλγόριθμος DQN επιλύει αυτό το ζήτημα αντικαθιστώντας τον πίνακα Q με ένα νευρωνικό δίκτυο που προσεγγίζει τη συνάρτηση Q.

Κύρια Στοιχεία του DQN

- **Δομή του Νευρωνικού Δικτύου**

Το Deep Q-Network είναι ένα πολυεπίπεδο νευρωνικό δίκτυο, συνήθως τύπου CNN (Convolutional Neural Network) για είσοδο εικόνας, ή MLP (Multi-Layer Perceptron) για άλλου τύπου δεδομένα. Σαν είσοδο έχει την τρέχουσα κατάσταση s_t του περιβάλλοντος και σαν έξοδο έναν πίνακα Q-values $Q(s_t,a;\theta^-)$, που περιέχει τις εκτιμήσεις Q για κάθε πιθανή ενέργεια a.

- **Replay Buffer**

Ο αλγόριθμος DQN χρησιμοποιεί έναν μηχανισμό αποθήκευσης εμπειριών, γνωστό ως Replay Buffer. Αποθηκεύει ζεύγη εμπειριών (s_t,a_t,r_t,s_{t+1}) που συλλέγονται κατά τη διάρκεια της εξερεύνησης του περιβάλλοντος. Ο σκοπός του είναι να διασπά τη συσχέτιση μεταξύ διαδοχικών εμπειριών, διευκολύνοντας την εκπαίδευση και να επιτρέπει την επαναχρησιμοποίηση δεδομένων, μειώνοντας την ανάγκη για συνεχείς αλληλεπιδράσεις με το περιβάλλον.

- **Δίκτυο Στόχου (Target Network)**

Για να αποφευχθεί η αστάθεια στη διαδικασία εκπαίδευσης, χρησιμοποιείται ένα σταθερό δίκτυο στόχου $Q(s,a;\theta^-)$ που έχει ίδιες δομές με το κύριο δίκτυο αλλά ενημερώνεται λιγότερο συχνά. Οι τιμές Q του στόχου υπολογίζονται ως: $y = r_t + \gamma \max_{a'} Q(s_{t+1},a';\theta^-)$.

- **Εξερεύνηση vs Εκμετάλλευση (Exploration vs Exploitation)**

Η πολιτική που ακολουθεί ο πράκτορας είναι η ϵ, με την οποία πιθανότητα ϵ, επιλέγεται μια τυχαία ενέργεια (εξερεύνηση), ενώ με την πιθανότητα $1-\epsilon$, επιλέγεται η ενέργεια με το μέγιστο Q-value (εκμετάλλευση). Η τιμή ϵ μειώνεται σταδιακά με την πάροδο του χρόνου (annealing), επιτρέποντας στον πράκτορα να μεταβαίνει από την εξερεύνηση στην εκμετάλλευση.

Εκπαιδευτική Διαδικασία

Η διαδικασία εκπαίδευσης ενός DQN περιλαμβάνει τα εξής βασικά στάδια:

1. **Αρχικοποίηση**: Οι παράμετροι του DQN θ, αρχικοποιούνται τυχαία και δημιουργείται ένας άδειος Replay Buffer.

2. **Αλληλεπίδραση με το Περιβάλλον**: Ο πράκτορας παρατηρεί την κατάσταση s_t και επιλέγει και εκτελεί μια ενέργεια a_t βάσει της ϵ-greedy policy. Επίσης, παρατηρεί την ανταμοιβή r_t και τη νέα κατάσταση s_{t+1}. Τέλος, αποθηκεύει την εμπειρία $(st,at,rt,st+1)$ στο Replay Buffer.

3. **Ενημέρωση Παραμέτρων**: Δειγματοληπτείται ένα τυχαίο υποσύνολο εμπειριών από το Replay Buffer. Στη συνέχεια υπολογίζεται ο στόχος εκπαίδευσης για κάθε εμπειρία με τον τύπο:

$$y = r_t + \gamma \max_{a'} Q(s_{t+1}, a'; \theta^-)$$

   Έπειτα, υπολογίζεται το σφάλμα (loss) μεταξύ του εκτιμώμενου Q-value και του στόχου y:

$$L(\theta) = E[(y - Q(s_t, a_t; \theta))^2]$$

   Οι παράμετροι $\boldsymbol{\theta}$ ενημερώνονται με χρήση gradient descent.
4. **Ενημέρωση Δικτύου Στόχου**: Περιοδικά, αντιγράφονται οι παράμετροι του κύριου δικτύου στο δίκτυο στόχου: $\theta^- \leftarrow \theta$
5. **Επανάληψη**: Η διαδικασία συνεχίζεται για πολλούς κύκλους μέχρι να επιτευχθεί ικανοποιητική απόδοση.

Πλεονεκτήματα και Μειονεκτήματα

Υπάρχουν όπως σε όλους τους αλγορίθμους κάποια πλεονεκτήματα αλλά και μειονεκτήματα, τα πιο βασικά από τα οποία αναφέρονται παρακάτω.

Πλεονεκτήματα:

- Υπάρχει η ικανότητα διαχείρισης σύνθετων περιβάλλοντων.
- Έχουμε μάθηση από εικόνες και ακατέργαστα δεδομένα.
- Υπάρχει βελτιωμένη σταθερότητα μέσω Replay Buffer και δικτύου στόχου.

Μειονεκτήματα:

- Έχουμε υψηλή υπολογιστική απαίτηση.
- Υπάρχει ευαισθησία στις παραμέτρους εκπαίδευσης (όπως το ϵ και το learning rate).

Εφαρμογές

Κάποιες από τις πολλές εφαρμογές του αλγορίθμου είναι:

- Στη Ρομποτική: Βελτιστοποίηση κινήσεων.
- Στα Παιχνίδια: Αυτόματοι παίκτες (π.χ. Atari).
- Στα Προβλήματα ελέγχου: Αυτόνομα οχήματα, διαχείριση πόρων.

2.5.4 Policy Gradients

Ο αλγόριθμος *Policy Gradients* ανήκει στην κατηγορία των αλγορίθμων ενισχυτικής μάθησης (*reinforcement learning*), ο οποίος προσπαθεί να βρει την βέλτιστη πολιτική με την εκτίμηση των παραμέτρων της πολιτικής άμεσα, αντί να επικεντρωθεί στην εκτίμηση των Q-values όπως συμβαίνει στους αλγορίθμους αξίας (*value-based methods*). Ο αλγόριθμος αυτός είναι ιδιαίτερα αποτελεσματικός όταν ο χώρος των ενεργειών είναι συνεχής ή όταν η πολιτική δεν μπορεί να παραμετροποιηθεί με εύκολο ή απλό τρόπο [9]. Στην καρδιά του αλγορίθμου βρίσκεται η βελτιστοποίηση μιας συνάρτησης στόχου που αφορά τις παραμέτρους της πολιτικής, χρησιμοποιώντας την έννοια του *gradient ascent* [96].

Η διαδικασία εκπαίδευσης στον αλγόριθμο Policy Gradients περιλαμβάνει τα εξής βήματα:

1. **Ορισμός του Χώρου Καταστάσεων και Ενεργειών (State and Action Space):** Το πρώτο βήμα στην εφαρμογή του αλγορίθμου είναι ο καθορισμός του χώρου των καταστάσεων, ο οποίος αντιπροσωπεύει το σύνολο των πιθανών καταστάσεων του περιβάλλοντος. Επίσης, ορίζεται ο χώρος των ενεργειών, που αποτελεί το σύνολο των ενδεχόμενων ενεργειών που μπορεί να εκτελέσει ο πράκτορας σε κάθε δεδομένη κατάσταση. Αυτός ο ορισμός είναι κρίσιμος για τον καθορισμό της πολιτικής και της στρατηγικής που θα ακολουθήσει ο πράκτορας [9].
2. **Δημιουργία του Δίκτυο Πολιτικής (Policy Network):** Ο αλγόριθμος απαιτεί την κατασκευή ενός νευρωνικού δικτύου, το οποίο λαμβάνει ως είσοδο την κατάσταση του περιβάλλοντος και υπολογίζει την κατανομή πιθανοτήτων για τις διάφορες ενέργειες. Η έξοδος του δικτύου είναι μια κατανομή πιθανοτήτων πάνω στο σύνολο των ενεργειών, και ο πράκτορας επιλέγει μια ενέργεια με βάση αυτές τις πιθανότητες. Το δίκτυο πολιτικής μπορεί να παραμετροποιηθεί με βάρη, τα οποία ενημερώνονται κατά τη διάρκεια της εκπαίδευσης [92].
3. **Καθορισμός της Συνάρτησης Πλεονεκτήματος (Advantage Function):** Η συνάρτηση πλεονεκτήματος χρησιμοποιείται για να μετρήσει την ποιοτική αξία μιας ενέργειας σε σχέση με τη μέση αξία όλων των ενδεχόμενων ενεργειών για μια δεδομένη κατάσταση. Η συνάρτηση πλεονεκτήματος εκτιμά πόσο καλύτερη είναι η απόδοση μιας ενέργειας σε σχέση με το αναμενόμενο μέσο όρο. Συχνά, η συνάρτηση πλεονεκτήματος υπολογίζεται ως η διαφορά μεταξύ της ανταμοιβής που έλαβε ο πράκτορας και της αναμενόμενης ανταμοιβής του από την πολιτική [93].
4. **Υπολογισμός της Συνάρτησης Καταλληλότητας (Fitness Function):** Στη συνέχεια, υπολογίζεται η συνάρτηση καταλληλότητας, η οποία αξιολογεί πόσο καλή είναι η πολιτική με βάση την αναμενόμενη απόδοση. Η συνάρτηση καταλληλότητας συνήθως περιλαμβάνει το γινόμενο των πιθανοτήτων ενεργειών από το δίκτυο πολιτικής και των αντίστοιχων πλεονεκτημάτων. Αυτή η συνάρτηση παρέχει την αναμενόμενη απόδοση για τον πράκτορα και καθοδηγεί την ενημέρωση των παραμέτρων του δικτύου πολιτικής [88].
5. **Υπολογισμός της κλίσης (Gradient):** Ο αλγόριθμος υπολογίζει τη κλίση της συνάρτησης καταλληλότητας ως προς τις παραμέτρους του δικτύου πολιτικής, χρησιμοποιώντας τη μέθοδο του *back propagation*. Αυτό επιτρέπει την ενημέρωση των παραμέτρων του δικτύου με στόχο τη μεγιστοποίηση της αναμενόμενης απόδοσης. Η τεχνική του *gradient ascent* χρησιμοποιείται για να κατευθυνθεί η μάθηση προς τις παραμέτρους που βελτιστοποιούν την πολιτική [96].
6. **Ενημέρωση των Παραμέτρων:** Οι παράμετροι του δικτύου πολιτικής ενημερώνονται σύμφωνα με τον υπολογισμό της κλίσης και τον ρυθμό μάθησης (learning rate). Η διαδικασία αυτή επιτρέπει στην πολιτική να βελτιώνεται διαρκώς

μέσω της αλληλεπίδρασης με το περιβάλλον, καθώς οι παράμετροι του δικτύου τροποποιούνται για να μεγιστοποιήσουν την αναμενόμενη απόδοση.

7. **Επανάληψη της Διαδικασίας:** Η διαδικασία αυτή επαναλαμβάνεται για ένα συγκεκριμένο αριθμό επαναλήψεων ή μέχρι να επιτευχθεί ικανοποιητική απόδοση. Καθώς ο πράκτορας συνεχίζει να αλληλεπιδρά με το περιβάλλον, η πολιτική εξελίσσεται και προσαρμόζεται στην καλύτερη δυνατή στρατηγική, με στόχο την επίτευξη του βέλτιστου αποτελέσματος.

Όταν έχουμε να κάνουμε με περιβάλλοντα συνεχούς δράσης, η προσέγγιση του αλγορίθμου Policy Gradients είναι αρκετά χρήσιμη, καθώς μπορεί να παραμετροποιήσει και να εκπαιδεύσει την πολιτική σε τέτοιες συνθήκες χωρίς να απαιτείται εκτίμηση τιμών. Είναι ιδιαίτερα χρήσιμη σε πεδία όπου οι συμβατικές προσεγγίσεις, όπως οι προσεγγίσεις που βασίζονται σε τιμές, είναι λιγότερο κατάλληλες ή αποτελεσματικές [97].

2.5.5 Actor-Critic

Στην ενισχυτική μάθηση, η τεχνική Actor-Critic είναι μια σημαντική κατηγορία αλγορίθμου που συνδυάζει δύο βασικά στοιχεία: προσεγγίσεις βασισμένες στην αξία (Critic) και πολιτική (Actor). Η επίτευξη αποτελεσματικής και σταθερής εκπαίδευσης σε πολύ σύνθετα και μεταβλητά περιβάλλοντα είναι ο πρωταρχικός στόχος αυτών των προσεγγίσεων. Η μέθοδος αυτή χρησιμοποιεί τόσο τις εκτιμήσεις των τιμών Q που δίνει ο Critic όσο και τη γνώση από τις ανταμοιβές για να ενισχύσει την πολιτική του πράκτορα (Agent) μέσω της αλληλεπίδρασής του με το περιβάλλον [9].

Στρατηγική Εκπαίδευσης Actor-Critic

Η εκπαίδευση των τεχνικών Actor-Critic μπορεί να αναλυθεί στα ακόλουθα βήματα:

1. **Ορισμός Χώρου Καταστάσεων και Ενεργειών**: Κατά τη χρήση της προσέγγισης, το πρώτο βήμα είναι ο ορισμός του χώρου καταστάσεων, ο οποίος είναι η συλλογή των πιθανών περιβαλλοντικών σεναρίων. Επιπλέον, ορίζεται ο χώρος δράσης -η συλλογή κάθε ενέργειας που θα μπορούσε να κάνει ο πράκτορας σε κάθε περίσταση. Σύμφωνα με τους Sutton και Barto [9], ο ορισμός αυτός είναι απαραίτητος για την πολιτική και τη στρατηγική που θα εφαρμόσει ο πράκτορας.
2. **Δημιουργία του Actor Network**: Το πρώτο στοιχείο του συστήματος είναι το "Actor", το οποίο είναι υπεύθυνο για την παραγωγή της πολιτικής του πράκτορα. Το Actor είναι ένα νευρωνικό δίκτυο που παίρνει ως είσοδο την κατάσταση του περιβάλλοντος και επιστρέφει πιθανότητες για τις ενέργειες που μπορούν να ληφθούν. Το δίκτυο αυτό καθοδηγεί την επιλογή της επόμενης ενέργειας, βασισμένη στις πιθανότητες που υπολογίζει για κάθε δράση [88].
3. **Δημιουργία του Critic Network**: Το δεύτερο στοιχείο είναι ο "Critic", ο οποίος έχει ως στόχο να εκτιμήσει την ποιότητα των επιλογών που κάνει ο Actor. Ο Critic, μέσω ενός άλλου νευρωνικού δικτύου, δέχεται την κατάσταση και την ενέργεια του Actor και υπολογίζει τις Q-values, που αντιπροσωπεύουν τη μακροπρόθεσμη ανταμοιβή από την ενέργεια αυτή. Η Q-value παρέχει ένα μέτρο της αξίας μιας ενέργειας σε μια δεδομένη κατάσταση [9].
4. **Καθορισμός Συνάρτησης Καταλληλότητας (Loss Function)**: Για την εκπαίδευση των δύο δικτύων, χρησιμοποιείται μια συνάρτηση καταλληλότητας (loss function) που ενσωματώνει την ανταμοιβή που παίρνει ο Actor και την εκτίμηση του Critic για

την ποιότητα της ενέργειας. Συνήθως χρησιμοποιείται η Advantage function, η οποία υπολογίζει τη διαφορά μεταξύ της ανταμοιβής και της εκτιμώμενης Q-value [97]. Αυτή η συνάρτηση προσδιορίζει πόσο καλύτερη ή χειρότερη ήταν η επιλογή του Actor σε σχέση με την εκτίμηση του Critic.

5. **Υπολογισμός Κλίσης (Gradient Calculation)**: Ο υπολογισμός της κλίσης της συνάρτησης καταλληλότητας με σκοπό την ανατροφοδότηση στα δίκτυα Actor και Critic πραγματοποιείται μέσω του αλγορίθμου back propagation. Οι παράμετροι των νευρωνικών δικτύων προσαρμόζονται χρησιμοποιώντας τον υπολογισμό των παραγώγων της συνάρτησης καταλληλότητας σε σχέση με τις παραμέτρους του κάθε δικτύου [95].
6. **Ενημέρωση των Παραμέτρων**: Οι παράμετροι του Actor και του Critic ενημερώνονται βάση των υπολογισμένων κλίσεων. Ο ρυθμός μάθησης καθορίζει το μέγεθος της προσαρμογής, και το μοντέλο επαναλαμβάνει τη διαδικασία μέχρι την επίτευξη ικανοποιητικής απόδοσης.
7. **Επανάληψη της Διαδικασίας**: Τα παραπάνω βήματα επαναλαμβάνονται για πολλές επαναλήψεις, βελτιώνοντας σταδιακά την πολιτική του Actor και την εκτίμηση του Critic μέσω της αλληλεπίδρασης με το περιβάλλον. Κάθε επανάληψη προσφέρει νέες πληροφορίες για την εκτίμηση των Q-values και την ενημέρωση της πολιτικής.

Η μέθοδος Actor-Critic συνδυάζει τα πλεονεκτήματα της πολιτικής και της εκτίμησης αξιών, προσφέροντας έναν ισχυρό μηχανισμό μάθησης για πολύπλοκα και αβέβαια περιβάλλοντα. Συγκεκριμένα, ο διαχωρισμός της εκμάθησης εκτίμησης τιμών και της εκμάθησης πολιτικής επιταχύνει τη σύγκλιση του μοντέλου και μειώνει τον απαιτούμενο όγκο δεδομένων. Ο Πράκτορας μπορεί να μάθει πιο αποτελεσματικά όταν ο Κριτικός μπορεί να προσφέρει ανατροφοδότηση σχετικά με το διαμέτρημα των επιλεγμένων πράξεων και η εκτίμηση των τιμών Q βοηθάει στην καλύτερη κατανόηση των μακροπρόθεσμων επιπτώσεων κάθε πράξης [94].

2.5.6 Proximal Policy Optimization (PPO)

Το Proximal Policy Optimization (PPO) αποτελεί έναν αλγόριθμο ενισχυτικής μάθησης, που έχει σχεδιαστεί για τη βελτιστοποίηση πολιτικών ελέγχου σε περιβάλλοντα ενισχυτικής μάθησης, χωρίς την ανάγκη χρήσης Q-values. Ο PPO είναι καινοτόμος ως προς την ελαχιστοποίηση των σημαντικών αλλαγών πολιτικής κατά τη διάρκεια της εκπαίδευσης, έτσι ώστε να εγγυάται σταθερή προσαρμογή της πολιτικής και να αποφεύγεται η ασταθής συμπεριφορά και η πιθανή αποτυχία αύξησης της απόδοσης [97][98]. Η διασφάλιση της σταθερότητας κατά τη διάρκεια της εκπαίδευσης είναι η βασική δυσκολία στις μεθόδους ενισχυτικής μάθησης, ο PPO το επιλύει αυτό επιβάλλοντας έναν περιορισμό αλλαγής πολιτικής μεταξύ των επαναλήψεων.

Η διαδικασία εκπαίδευσης με τη χρήση του PPO περιλαμβάνει τα εξής βήματα:

Ορισμός του Χώρου Καταστάσεων και Ενεργειών

Ο σαφής καθορισμός της κατάστασης και του χώρου δράσης του περιβάλλοντος έρχεται πρώτα. Ενώ ο χώρος δράσης αποτελείται από όλες τις πιθανές ενέργειες που μπορεί να εκτελέσει ο πράκτορας σε κάθε κατάσταση, ο χώρος καταστάσεων δείχνει όλες τις πιθανές καταστάσεις στις οποίες μπορεί να βρίσκεται το σύστημα.

Δημιουργία του Actor Network (Πολιτικό Δίκτυο)

Το actor network είναι ένα νευρωνικό δίκτυο που δέχεται την τρέχουσα κατάσταση του περιβάλλοντος ως είσοδο και παράγει τις πιθανότητες για κάθε ενέργεια που μπορεί να εκτελέσει ο πράκτορας. Ο στόχος του actor είναι να καθοδηγεί τις επιλογές του πράκτορα, εκτελώντας τις ενέργειες που μεγιστοποιούν τη συνολική ανταμοιβή (reward).

Δημιουργία του Critic Network (Εκτιμητής Advantage)

Ο critic είναι επίσης ένα νευρωνικό δίκτυο που εκτιμά την Advantage Function, η οποία υπολογίζει τη διαφορά μεταξύ της απόδοσης μιας ενέργειας και της μέσης απόδοσης της πολιτικής. Η Advantage Function υπολογίζεται χρησιμοποιώντας τη διαφορά της τιμής Q για την ενέργεια που εκτελέστηκε και την τιμή V για την κατάσταση στην οποία έγινε η ενέργεια [9]. Αυτή η εκτίμηση βοηθά στην κατεύθυνση της εκπαίδευσης του actor, καθώς προσδιορίζει ποιες ενέργειες είναι καλύτερες από τις άλλες.

Ορισμός της Loss Function

Η loss function για το PPO συνδυάζει δύο κύριες συνιστώσες: τον περιορισμό της αλλαγής της πολιτικής (προκειμένου να αποτραπούν μεγάλες, αποσταθεροποιητικές αλλαγές) και τη μέτρηση της ποιότητας της ενέργειας μέσω της Advantage Function. Ο περιορισμός αυτός καθορίζεται με τη βοήθεια του clipping mechanism, το οποίο περιορίζει την αλλαγή της πολιτικής εντός ενός εύρους, ώστε να αποφεύγεται η υπερβολική διαφοροποίηση [98]. Η κύρια ιδέα είναι να επιτρέπεται η μικρή βελτίωση της πολιτικής, χωρίς να γίνονται δραστικές αλλαγές που ενδέχεται να αποσταθεροποιήσουν το σύστημα.

Υπολογισμός της κλίσης (Gradient)

Ο υπολογισμός του gradient της loss function ως προς τις παραμέτρους των δικτύων actor και critic γίνεται χρησιμοποιώντας την τεχνική του back propagation, η οποία επιτρέπει την οπισθοδρομική μετάδοση του σφάλματος για την ενημέρωση των παραμέτρων των δικτύων.

Ενημέρωση των Παραμέτρων

Η υπολογισμένη κλίση και ο ρυθμός μάθησης βοηθούν στην αλλαγή των παραμέτρων του δικτύου, ώστε η πολιτική να αποδίδει καλύτερα στην επόμενη επανάληψη.

Επανάληψη της Διαδικασίας

Η διαδικασία επαναλαμβάνεται για έναν καθορισμένο αριθμό φορών ή έως ότου επιτευχθεί επαρκής απόδοση. Κάθε επανάληψη ενημερώνει την πολιτική και τις εκτιμήσεις των τιμών, βελτιώνοντας έτσι την όλη προσέγγιση του πράκτορα.

Η σταθερότητα που παρέχει η μέθοδος PPO στην εκπαίδευση στρατηγικών σε απαιτητικά περιβάλλοντα έχει συμβάλει στην εξήγηση της ευελιξίας και της ευρωστίας της. Η PPO χρησιμοποιεί λιγότερους υπολογιστικούς πόρους και είναι απλούστερη στη ρύθμιση σε σχέση με άλλους αλγορίθμους, επομένως, ακόμη και αν αποδίδει ισοδύναμα ή καλύτερα από άλλους, η υλοποίησή της είναι σημαντικά πιο απλή [98].

2.5.7 Soft Actor-Critic (SAC)

Οι προηγμένοι αλγόριθμοι ενισχυτικής μάθησης που ταξινομούνται ως Soft Actor-Critic (SAC) ανήκουν στην οικογένεια των Deep Reinforcement Learning (DRL). Σχεδιασμένος για να συνδυάζει τα οφέλη των τεχνικών πολιτικής (actor) και κριτικής (critic), ο Soft Actor-Critic δημιουργήθηκε κυρίως για την επίλυση προβλημάτων σε περιβάλλοντα συνεχούς δράσης και προορίζεται να παρέχει πιο αποτελεσματική και σταθερή εκπαίδευση σε αβέβαια και στοχαστικά περιβάλλοντα [99].

Ορισμός του Περιβάλλοντος και του Προβλήματος

Με τον ορισμό του περιβάλλοντος και του προβλήματος που χρήζει ενίσχυσης ξεκινά η διαδικασία εκπαίδευσης του SAC. Το περιβάλλον περιλαμβάνει έναν συνεχώς μεταβαλλόμενο χώρο καταστάσεων στον οποίο ο πράκτορας πρέπει να αποφασίσει όσον αφορά τις διάφορες ενέργειες. Λαμβάνοντας υπόψη την περιβαλλοντική αβεβαιότητα και το απρόβλεπτο, ο στόχος είναι η επίτευξη ενός συνολικού στόχου που συνήθως συνδέεται με τη βελτιστοποίηση μιας ανταμοιβής με την πάροδο του χρόνου.

Δημιουργία του Actor Network (Πολιτικό Δίκτυο)

Το λεγόμενο Actor Network, ένα νευρωνικό δίκτυο υπεύθυνο για την εκμάθηση της πολιτικής του πράκτορα, εφαρμόζεται στη μέθοδο SAC. Αυτή η πολιτική ελέγχει τη συμπεριφορά σε κάθε κατάσταση, το δίκτυο δημιουργεί την κατανομή πιθανοτήτων των ενεργειών (π.χ. μέσοι όροι και αποκλίσεις για συνεχείς ενέργειες). Με βάση την ιδέα της προσαρμογής της πολιτικής, το δίκτυο των πρακτόρων στοχεύει στη δημιουργία των ενεργειών που θα μεγιστοποιήσουν την προβλεπόμενη ανταμοιβή [100].

Δημιουργία του Critic Network (Κριτικό Δίκτυο)

Αντίστοιχα με το Actor, το **Critic Network** έχει στόχο να εκτιμήσει την ποιότητα των ενεργειών που προτείνει ο Actor. Το Critic Network υπολογίζει την Q-value για κάθε ενέργεια που εκτελείται σε κάθε δεδομένη κατάσταση. Η συνάρτηση Q υπολογίζεται λαμβάνοντας υπόψη την τρέχουσα πολιτική και αναφέρεται στην εκτίμηση της μέγιστης αναμενόμενης ανταμοιβής που θα προκύψει αν ο πράκτορας ακολουθήσει αυτή την πολιτική στο μέλλον [99]. Ο αλγόριθμος SAC υιοθετεί την κλασική προσέγγιση της αξίας δράσης, η οποία βοηθά στην καθοδήγηση της εκμάθησης του Actor.

Ορισμός της Συνάρτησης Απώλειας (Loss Function)

Η συνάρτηση απώλειας του SAC συνδυάζει την απώλεια του Actor και του Critic, προσδιορίζοντας έτσι την επιθυμητή κατεύθυνση εκπαίδευσης και τις αναγκαίες βελτιώσεις των παραμέτρων. Η βασική καινοτομία του SAC έγκειται στο γεγονός ότι η συνάρτηση απώλειας περιλαμβάνει έναν όρο περιορισμού (entropy regularization term), ο οποίος ενθαρρύνει την πολιτική να είναι στοχαστική (δηλαδή να διατηρεί την αβεβαιότητα στις ενέργειες), αποτρέποντας έτσι τη μονοτονία της πολιτικής [99].

Η ενσωμάτωση αυτής της παραμέτρου αυξάνει την εξερευνητική συμπεριφορά του πράκτορα, κάτι που είναι κρίσιμο για την αποτελεσματικότητα σε πολύπλοκα και δυναμικά περιβάλλοντα.

Υπολογισμός του Κλίσης (Gradient)

Μετά τη συνάρτηση απώλειας, η κλίση υπολογίζεται με τη μέθοδο του back propagation για να διαπιστωθούν οι απαιτούμενες τροποποιήσεις στις παραμέτρους του δικτύου Actor και Critic. Ο υπολογισμός αυτός γίνεται για να μεγιστοποιηθεί η ανταμοιβή που δίνει η πολιτική και ταυτόχρονα να ελαχιστοποιηθεί η συνολική απώλεια.

Ενημέρωση των Παραμέτρων

Αφού υπολογιστούν οι κλίσεις, οι παράμετροι του Actor και του Critic ενημερώνονται με βάση την κατεύθυνση που δείχνουν οι κλίσεις και τον ρυθμό μάθησης, προκειμένου να βελτιώσουν τη στρατηγική εκτέλεσης και εκτίμησης των ενεργειών αντίστοιχα. Η διαδικασία αυτή γίνεται μέσω της χρήσης αλγορίθμων βελτιστοποίησης όπως ο Stochastic Gradient Descent (SGD) [99].

Επανάληψη της Διαδικασίας

Πολλές φορές, ο πράκτορας (agent) συνεχίζει να αναπτύσσει την πολιτική του μαθαίνοντας από τις αλληλεπιδράσεις με το περιβάλλον, επομένως επαναλαμβάνεται η παραπάνω διαδικασία. Η διαδικασία αυτή συνεχίζεται είτε για ένα καθορισμένο αριθμό επαναλήψεων είτε μέχρι η απόδοση του αλγορίθμου να γίνει λογική.

Αντιμετώπιση των Προκλήσεων στην Ενισχυτική Μάθηση

Ο SAC έχει αναπτυχθεί για να αντιμετωπίσει βασικές προκλήσεις στην ενισχυτική μάθηση, όπως η ανάγκη για εκτενή εξερεύνηση του περιβάλλοντος και η ισχυρή ισορροπία μεταξύ εξερεύνησης και εκμετάλλευσης (exploration vs. exploitation). Η χρήση του όρου περιορισμού της εντροπίας (entropy regularization) επιτρέπει στον πράκτορα να διατηρήσει μια στοχαστική πολιτική, ενθαρρύνοντας τη συνεχιζόμενη εξερεύνηση ενώ παράλληλα εκμεταλλεύεται τις ήδη αποκτηθείσες γνώσεις [99]. Επιπλέον, ο SAC επιτρέπει τη διαχείριση της παραλλαγής στις ανταμοιβές μέσω της βελτίωσης της αξιολόγησης των ενεργειών του πράκτορα, μειώνοντας έτσι την ευαισθησία του αλγορίθμου στις αυξομειώσεις των ανταμοιβών.

Τελικά, ειδικά σε δυναμικές και συνεχείς καταστάσεις, ο αλγόριθμος Soft Actor-Critic είναι από τις πιο σύγχρονες και αποτελεσματικές μεθόδους ενισχυτικής μάθησης που υπάρχουν. Η επιτυχία του πηγάζει από την ικανότητά του να προσφέρει συνεπείς εκτιμήσεις μέσω των δικτύων Actor και Critic και να περιλαμβάνει μηχανισμούς εξερεύνησης μέσω της εντροπίας. Αντίθετα, η βελτίωση των παραμέτρων του SAC για συγκεκριμένα ζητήματα απαιτεί σχολαστική παραμετροποίηση και συνεχή αξιολόγηση των επιδόσεων υπό διάφορες συνθήκες.

3.Μέτρα Αξιολόγησης

Η ανάλυση των προβλημάτων ταξινόμησης είναι θεμελιώδης για την αξιολόγηση και βελτίωση της απόδοσης των μοντέλων μηχανικής μάθησης, ειδικότερα σε εφαρμογές όπως η διάγνωση ιατρικών παθήσεων, όπως ο διαβήτης. Στο πλαίσιο αυτό, ο πίνακας σύγχυσης (confusion matrix) αποτελεί ένα από τα πιο διαδεδομένα και χρήσιμα εργαλεία για την ποσοτική αξιολόγηση της αποτελεσματικότητας των ταξινομητών, καθώς συγκρίνει τις πραγματικές ετικέτες (κατηγορίες) με τις προβλεπόμενες ετικέτες από το μοντέλο [101]. Η ανάλυση των επιμέρους στοιχείων του πίνακα σύγχυσης και των δεικτών απόδοσης που προκύπτουν από αυτόν είναι απαραίτητη για την κατανόηση της συμπεριφοράς του μοντέλου και τη βελτίωση της ακρίβειας των προβλέψεων του.

3.1 Πίνακας Σύγχυσης (Confusion Matrix)

Για κάθε κατηγορία, ο πίνακας σύγχυσης προσφέρει μια σύνοψη των ακριβών και λανθασμένων προβλέψεων για το μοντέλο. Σε ένα δυαδικό πρόβλημα ταξινόμησης, όπως μια διάγνωση διαβήτη, ο πίνακας περιλαμβάνει τέσσερις πρωταρχικές κατηγορίες:

- **Σωστά Αρνητικά (True Negative, TN):** Ο αριθμός των περιπτώσεων που είναι πραγματικά αρνητικές και το μοντέλο τις έχει ταξινομήσει σωστά ως αρνητικές. Στο παράδειγμα διάγνωσης, αυτό αφορά τους υγιείς ανθρώπους που το μοντέλο αναγνωρίζει σωστά ως υγιείς [102].
- **Σωστά Θετικά (True Positive, TP):** Ο αριθμός των περιπτώσεων που είναι πραγματικά θετικές και το μοντέλο τις έχει ταξινομήσει σωστά ως θετικές. Στο παράδειγμα διάγνωσης, αυτό αφορά τα άτομα με διαβήτη που το μοντέλο αναγνωρίζει σωστά ως διαβητικά.
- **Ψευδώς Θετικά (False Positive, FP):** Ο αριθμός των περιπτώσεων που είναι πραγματικά αρνητικές αλλά το μοντέλο τις έχει ταξινομήσει λανθασμένα ως θετικές. Αυτό συμβαίνει όταν το μοντέλο εντοπίζει λάθος ένα υγιές άτομο ως άτομο με διαβήτη [102].
- **Ψευδώς Αρνητικά (False Negative, FN):** Ο αριθμός των περιπτώσεων που είναι πραγματικά θετικές αλλά το μοντέλο τις έχει ταξινομήσει λανθασμένα ως αρνητικές. Σε αυτή την περίπτωση, το μοντέλο αποτυγχάνει να εντοπίσει ένα άτομο με διαβήτη [101].

  Ο παρακάτω πίνακας δείχνει τον πίνακα σύγχυσης για δυαδική ταξινόμηση (Confusion matrix for binary classification)

Confusion Matrix		**Predicted**	
		Negative	**Positive**
Actual	**Negative**	TN	FP
	Positive	FN	TP

3.2 Δείκτες Απόδοσης

Από τα στοιχεία του πίνακα σύγχυσης, μπορούμε να υπολογίσουμε διάφορους δείκτες απόδοσης που μας επιτρέπουν να αξιολογήσουμε την ποιότητα των προβλέψεων του μοντέλου. Κάποιες από τις πιο γνωστές και ευρέως χρησιμοποιούμενες, αναφέρονται παρακάτω.

3.2.1. Πραγματικά θετικό ποσοστό (True Positive Rate - TPR) / Ανάκληση (Recall)

Ο δείκτης αυτός δείχνει το ποσοστό των πραγματικών θετικών περιπτώσεων που εντοπίζονται σωστά από το μοντέλο και υπολογίζεται με τον τύπο:

$$\text{TPR} = \frac{TP}{TP+FN}$$

Ο TPR (ή ευαισθησία) είναι κρίσιμος σε ιατρικές εφαρμογές, όπου το να εντοπιστούν όσο το δυνατόν περισσότερες θετικές περιπτώσεις είναι ζωτικής σημασίας για την έγκαιρη διάγνωση [102].

3.2.2. Ποσοστό Σωστά Αρνητικά (True Negative Rate - TNR) / Ειδικότητα (Specificity)

Ο δείκτης αυτός αποτυπώνει το ποσοστό των πραγματικών αρνητικών περιπτώσεων που εντοπίζονται σωστά από το μοντέλο και υπολογίζεται ως εξής:

$$\text{TNR} = \frac{TN}{TN+FP}$$

Η ειδικότητα είναι επίσης σημαντική σε εφαρμογές ιατρικής διάγνωσης, όπου το μοντέλο πρέπει να αποφεύγει τη λανθασμένη διάγνωση υγιών ατόμων ως ασθενών [101].

3.2.3. Ακρίβεια (Precision) / Θετική Προβλεπτική Αξία (Positive Predictive Value)

Η ακρίβεια υπολογίζεται ως το ποσοστό των θετικών προβλέψεων που είναι σωστές και εκφράζεται με τον τύπο:

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

Η ακρίβεια είναι σημαντική όταν οι ψευδώς θετικές προβλέψεις πρέπει να μειωθούν, όπως σε περιπτώσεις όπου οι λάθος θετικές προβλέψεις έχουν σημαντικές συνέπειες [101].

3.2.4. Ορθότητα (Accuracy)

Η ορθότητα υπολογίζεται ως το ποσοστό των σωστών προβλέψεων (θετικών και αρνητικών) σε σχέση με το σύνολο των προβλέψεων:

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

Η ορθότητα παρέχει μια συνολική εκτίμηση του μοντέλου, αλλά μπορεί να παραπλανήσει σε περιπτώσεις ανισόρροπων δεδομένων, όπου οι θετικές και αρνητικές περιπτώσεις είναι ασύμμετρα κατανεμημένες [102].

3.2.5. Βαθμολογία F1 (F1 Score)

Η βαθμολογία F1 είναι ο αρμονικός μέσος όρος της ακρίβειας και της ανάκλησης και χρησιμοποιείται όταν απαιτείται μια ισχυρή ισορροπία μεταξύ των δύο αυτών δεικτών:

$$\mathrm{F1} = 2 \times \frac{Precision \times Recall}{Precision + Recall}$$

Η βαθμολογία F1 είναι ιδιαίτερα χρήσιμη όταν έχουμε ανισόρροπες κατηγορίες, καθώς αποφεύγει να ευνοεί υπερβολικά τον έναν δείκτη σε βάρος του άλλου [101].

3.2.6. Εμβαδόν κάτω από την Καμπύλη (AUC - Area Under Curve)

Το AUC αναφέρεται στο εμβαδόν κάτω από την καμπύλη ROC (Receiver Operating Characteristic), η οποία απεικονίζει την ισορροπία μεταξύ ευαισθησίας και ψευδώς θετικού ρυθμού (FPR). Το AUC δίνει μια γενική εικόνα της ικανότητας του μοντέλου να διακρίνει μεταξύ των κατηγοριών:

$$\mathrm{AUC} = \int \mathrm{ROC}$$

Όταν το AUC είναι κοντά στο 1, το μοντέλο έχει εξαιρετική ικανότητα διάκρισης, ενώ όταν είναι κοντά στο 0.5, η διάκριση είναι τυχαία [102].

3.3 Συμπεράσματα

Η εξέταση του πίνακα σύγχυσης και των μετρικών απόδοσης είναι ζωτικής σημασίας για την κατανόηση της αποτελεσματικότητας ενός μοντέλου μηχανικής μάθησης, όπως στη διάγνωση του διαβήτη. Ο συνδυασμός αυτών των δεικτών διευκολύνει την αξιολόγηση της αποτελεσματικότητας του μοντέλου και βοηθά στη βελτίωσή του μέσω τροποποιήσεων και βελτιστοποιήσεων [102].

4.Βάσεις δεδομένων για την διάγνωση του Σακχαρώδης Διαβήτη

4.1 Εισαγωγή

Η βιβλιογραφία σχετικά με τη διάγνωση του διαβήτη με τη χρήση τεχνικών μηχανικής μάθησης δείχνει ότι χρησιμοποιούνται συχνά σύνολα δεδομένων που περιλαμβάνουν διάφορους βιοχημικούς δείκτες και δημογραφικά δεδομένα, οι περισσότερες μελέτες συγκλίνουν στη βάση δεδομένων των PIMA Indians, η οποία χρησιμοποιείται εκτενώς για εκπαιδευτικούς και ερευνητικούς σκοπούς. Επιπλέον, αναγνωρίζοντας τις προκλήσεις που συνδέονται με την ανάπτυξη μιας βάσης δεδομένων που απαιτεί μια συγκεκριμένη διαδικασία για τη διασφάλιση των προσωπικών δεδομένων και άλλων κριτηρίων, η βάση δεδομένων PIMA Indians επιλέχθηκε για την προσομοίωση στην παρούσα διπλωματική λόγω της επικράτησής της στην πρόβλεψη του διαβήτη. Η βάση δεδομένων του UCI Machine Learning Repository είναι επίσης προσβάσιμη στη διεύθυνση: **https://www.kaggle.com/uciml/pima-indians-diabetes-database**, όπου παρέχεται παρακάτω μια ολοκληρωμένη περιγραφή του περιεχομένου και των εφαρμογών της βάσης δεδομένων [11].

4.2 Βάση Δεδομένων PIMA Indians (PIMA Indians Diabetes Database)

Το σύνολο δεδομένων PIMA Indians είναι ένα από τα πιο γνωστά και συχνά χρησιμοποιούμενα σύνολα δεδομένων για την ανάλυση και πρόβλεψη του διαβήτη τύπου 2. Προέρχεται από μια μελέτη για τον διαβήτη σε γυναίκες των PIMA Indians, μια ομάδα με υψηλή συχνότητα εμφάνισης διαβήτη. Ας δούμε αναλυτικά τα χαρακτηριστικά του:

4.2.1 Περιγραφή του Συνόλου Δεδομένων

Το σύνολο δεδομένων περιέχει 768 εγγραφές και 8 χαρακτηριστικά (επιπλέον την ετικέτα στόχου - Outcome). Κάθε εγγραφή αναφέρεται σε μια γυναίκα ηλικίας άνω των 21 ετών [111].

4.2.2 Χαρακτηριστικά

1. **Pregnancies**: Αριθμός κυήσεων.

2. **Glucose**: Συγκέντρωση γλυκόζης στο αίμα 2 ώρες μετά από δοκιμασία ανοχής γλυκόζης (mg/dL).

3. **Blood Pressure**: Διαστολική αρτηριακή πίεση (mmHg).

4. **Skin Thickness**: Πάχος δερματικής πτυχής του τρικέφαλου (mm).

5. **Insulin**: Επίπεδα ινσουλίνης στον ορό 2 ώρες μετά από δοκιμασία ανοχής γλυκόζης (μIU/mL).

6. **BMI**: Δείκτης μάζας σώματος (βάρος σε kg / (ύψος σε m)^2).

7. **Diabetes Pedigree Function**: Λειτουργία κληρονομικότητας διαβήτη (μέτρο της πιθανότητας εμφάνισης διαβήτη με βάση το οικογενειακό ιστορικό).

8. **Age**: Ηλικία (σε χρόνια).

9. **Outcome**: Τιμή στόχου (0 ή 1), όπου 1 σημαίνει ότι ο ασθενής έχει διαγνωστεί με διαβήτη.

4.2.3 Παράδειγμα Δεδομένων

Pregnancies	BloodPressure	Glucose	SkinThickness	Insulin	BMI	DiabetesPedigreeFunction	Age	Outcome
6	148	72	35	0	33.6	0.627	50	1
1	85	66	29	0	26.6	0.351	31	0
8	183	64	0	0	23.3	0.672	32	1

4.2.4 Περιγραφή Χαρακτηριστικών

Χαρακτηριστικά	**Περιγραφή**	**Κατηγορία**	**Τύπος**
Pregnancies	Ο αριθμός των φορών που μια γυναίκα έχει μείνει έγκυος	Αριθμητικός-Numerical	Ακέραιος Αριθμός-Integer
Glucose	Η συγκέντρωση γλυκόζης στο αίμα 2 ώρες μετά από δοκιμασία ανοχής γλυκόζης (mg/dL)	Αριθμητικός-Numerical	Ακέραιος Αριθμός-Integer
BloodPressure	Η διαστολική αρτηριακή πίεση (mmHg)	Αριθμητικός-Numerical	Ακέραιος Αριθμός-Integer
SkinThickness	Το πάχος της δερματικής πτυχής του τρικέφαλου (mm)	Αριθμητικός-Numerical	Ακέραιος Αριθμός-Integer
Insulin	Τα επίπεδα ινσουλίνης στον ορό 2 ώρες μετά από δοκιμασία ανοχής γλυκόζης (μIU/mL)	Αριθμητικός-Numerical	Ακέραιος Αριθμός-Integer
BMI	Ο δείκτης μάζας σώματος υπολογίζεται ως βάρος σε κιλά διαιρούμενο με το τετράγωνο του ύψους σε μέτρα (kg/m²)	Αριθμητικός-Numerical	float

DiabetesPedigreeFunction	Ένας δείκτης που αναπαριστά την πιθανότητα εμφάνισης διαβήτη με βάση το οικογενειακό ιστορικό	Αριθμητικός-Numerical	float
Age	Η ηλικία της γυναίκας σε έτη	Αριθμητικός-Numerical	Ακέραιος Αριθμός-Integer
Outcome	Δυαδικός δείκτης (0 ή 1) που υποδηλώνει αν η γυναίκα έχει διαγνωστεί με διαβήτη (1) ή όχι (0)	Categorical	Ακέραιος Αριθμός-Integer

4.2.5 Σημασία των χαρακτηριστικών του συνόλου δεδομένων PIMA Indians

Τα χαρακτηριστικά του συνόλου δεδομένων, το καθένα με μοναδικό τρόπο, επηρεάζουν τις προβλέψεις και τα αποτελέσματα. Η σημασία τους περιγράφεται παρακάτω.

Pregnancies (Αριθμός Κυήσεων): Οι εγκυμοσύνες μπορούν να επηρεάσουν την ευαισθησία στην ινσουλίνη και να αυξήσουν τον κίνδυνο διαβήτη κύησης, που ίσως οδηγήσει σε διαβήτη τύπου 2 στη συνέχεια.

Glucose (Συγκέντρωση Γλυκόζης): Η συγκέντρωση γλυκόζης είναι απαραίτητη για την αξιολόγηση της ικανότητας του οργανισμού να ρυθμίζει τα επίπεδα σακχάρου στο αίμα. Οι αυξημένες μετρήσεις σηματοδοτούν προδιαβήτη ή διαβήτη.

Blood Pressure (Αρτηριακή Πίεση): Η υπέρταση αποτελεί παράγοντα κινδύνου για καρδιαγγειακά νοσήματα και σχετίζεται με τον διαβήτη. Η διαστολική πίεση ποσοτικοποιεί την αρτηριακή πίεση κατά τη φάση ηρεμίας της καρδιάς μεταξύ των παλμών.

Skin Thickness (Πάχος Δερματικής Πτυχής): Η μέτρηση του πάχους των δερματικών πτυχών χρησιμεύει ως δείκτης του υποδόριου λίπους και της συνολικής μάζας σωματικού λίπους. Η μέτρηση αυτή μπορεί να σηματοδοτήσει την ποσότητα του σωματικού λίπους.

Insulin (Ινσουλίνη): Η ινσουλίνη είναι η ορμόνη που ρυθμίζει τα επίπεδα γλυκόζης στο αίμα. Τα μη φυσιολογικά επίπεδα ινσουλίνης μπορεί να υποδηλώνουν αντίσταση στην ινσουλίνη ή δυσλειτουργία των β-κυττάρων του παγκρέατος, τα οποία είναι και τα δύο ενδεικτικά του διαβήτη τύπου 2.

BMI (Δείκτης Μάζας Σώματος): Ο Δείκτης Μάζας Σώματος (BMI) είναι ένα ευρέως χρησιμοποιούμενο μέτρο για την αξιολόγηση της παχυσαρκίας, ενός σημαντικού παράγοντα κινδύνου για την εμφάνιση διαβήτη.

Diabetes Pedigree Function (Λειτουργία Κληρονομικότητας Διαβήτη): Αυτός ο δείκτης αξιολογεί την πιθανότητα εμφάνισης διαβήτη λαμβάνοντας υπόψη τη γενετική προδιάθεση και το οικογενειακό ιστορικό του ατόμου. Αυτός ο δείκτης είναι ζωτικής σημασίας για την κατανόηση της γενετικής προδιάθεσης.

Age (Ηλικία): Ο κίνδυνος ανάπτυξης διαβήτη αυξάνεται με την ηλικία. Οι γυναίκες προχωρημένης ηλικίας παρουσιάζουν μεγαλύτερη τάση για ανάπτυξη διαβήτη τύπου 2.

Outcome (Αποτέλεσμα): Αυτή η μεταβλητή χρησιμεύει ως στόχος για την εκπαίδευση και την αξιολόγηση των μοντέλων πρόβλεψης.

4.2.6 Χρήσεις στην Ανάλυση

Αυτή η βάση δεδομένων έχει πολλαπλές εφαρμογές σε διάφορους τομείς, βοηθώντας σημαντικά στη δημιουργία συμπερασμάτων που ενισχύουν τη μελέτη και την κατανόηση του διαβήτη. Ακολουθούν ορισμένες από αυτές τις εφαρμογές.

- Μοντελοποίηση Πρόβλεψης (Predictive Modeling): Χρησιμοποιείται για τη δημιουργία μοντέλων μηχανικής μάθησης που προβλέπουν την πιθανότητα εμφάνισης διαβήτη. Μερικά από τα εργαλεία που χρησιμοποιούνται περιλαμβάνουν τη λογιστική παλινδρόμηση, τα δέντρα αποφάσεων, τα νευρωνικά δίκτυα και τα μοντέλα τυχαίου δάσους.

- Ανάλυση Δεδομένων (Data Analysis): Η εξέταση δεδομένων για τον εντοπισμό μοτίβων και σχέσεων μεταξύ των χαρακτηριστικών. Χρησιμοποιείται για στατιστικές μεθόδους για τη διάκριση σημαντικών στοιχείων κινδύνου.

- Εκπαίδευση και Εκμάθηση (Education and Learning): Αυτή η βάση δεδομένων χρησιμοποιείται σε μαθήματα μηχανικής μάθησης και ανάλυσης δεδομένων για την εκπαίδευση των φοιτητών. Προσφέρει ένα απτό παράδειγμα ενός συνόλου δεδομένων για πρακτική εκπαίδευση.

- Δημόσια Υγεία (Public Health): Χρησιμοποιείται για την εξέταση της επιδημιολογίας του διαβήτη μεταξύ των PIMA Indians, βοηθώντας στην κατανόηση των γενετικών και περιβαλλοντικών παραγόντων που επηρεάζουν τον διαβήτη.

Αυτό το σύνολο δεδομένων χρησιμεύει ως μια ισχυρή βάση για τη διερεύνηση του διαβήτη και τη διαμόρφωση προγνωστικών μοντέλων που διευκολύνουν την έγκαιρη ανίχνευση και την πρόληψη της νόσου.

5. Παρουσίαση προηγούμενων μελετών με εστίαση στη διάγνωση του διαβήτη με μοντέλα μηχανικής μάθησης

Αρκετές μελέτες στον τομέα της διάγνωσης του διαβήτη με τη χρήση αλγορίθμων μηχανικής μάθησης έχουν διεξαχθεί τα προηγούμενα χρόνια κυρίως και καθώς ο τομέας της μηχανικής μάθησης εξελίσσεται. Στο πλαίσιο των αλγορίθμων και των παραμέτρων που εφαρμόζονται, το παρόν κεφάλαιο θα παρουσιάσει ορισμένες από τις μελέτες σχετικά με τη διάγνωση του διαβήτη μαζί με την αξιολόγηση της απόδοσής τους. Σε γενικές γραμμές, οι περισσότερες από τις μελέτες που παρουσιάστηκαν έκαναν χρήση της βάσης δεδομένων PIMA Indians, η οποία περιγράφηκε προηγουμένως.

5.1 Πρώτη Μελέτη: "Comparative Approaches for Classification of Diabetes Mellitus Data: Machine Learning Paradigm" by Md. Maniruzzaman

Εστιάζοντας στη βάση δεδομένων PIMA Indians, η εργασία του Md. Maniruzzaman και των συνεργατών του, αποσκοπεί στην αξιολόγηση διαφόρων προσεγγίσεων μηχανικής μάθησης για την ταξινόμηση δεδομένων διαβήτη [132]. Η έρευνα συγκρίνει την απόδοση της ταξινόμησης Gaussian Process (GPC) με διαφορετικούς τύπους πυρήνα (γραμμικός, πολυωνυμικός και ραδιοβαθμίδας) και τρεις παραδοσιακούς ταξινομητές: τη Γραμμική Διακριτική Ανάλυση (LDA), την Τετραγωνική Διακριτική Ανάλυση (QDA) και το μοντέλο Naive Bayes (NB). Ειδικότερα για τη διάγνωση του διαβήτη, η μελέτη προάγει τη γνώση της αποτελεσματικότητας των διαφόρων προσεγγίσεων για την οργάνωση των ιατρικών δεδομένων.

5.1.1 Κύρια Ευρήματα

Σε αυτή τη μελέτη, έγινε χρήση K=5 και K=10 cross-validation protocol, που φάνηκε πως για K=10, είχαμε τα καλύτερα αποτελέσματα. Παρατίθενται ξεχωριστά για κάθε μοντέλο μηχανικής μάθησης, τα κυριότερα συμπεράσματα και αποτελέσματα μετά την εφαρμογή ορισμένων αλγορίθμων στη βάση δεδομένων PIMA Indians και για το cross-validation protocol με K=10.

1. **Ταξινόμηση Gaussian Process (GPC-Gaussian Process Classification)**

   Ειδικότερα λαμβάνοντας υπόψη τον πυρήνα ραδιοβαθμίδας (RBF), η μελέτη υπογραμμίζει το μεγάλο πλεονέκτημα της GPC. Με ευαισθησία 91,79%, ειδικότητα 63,33%, θετική προγνωστική αξία (PPV) 84,91% και αρνητική προγνωστική αξία (NPV) 62,50%, η GPC πέτυχε μεγάλη ακρίβεια 81,97%. Τα ευρήματα αυτά δείχνουν πόσο αποτελεσματικά μπορεί η GPC να διαχειριστεί περίπλοκα, μη γραμμικά

δεδομένα, μια απαραίτητη ικανότητα για την επεξεργασία ιατρικών δεδομένων με ποικίλες ιδιότητες [108].

2. **Γραμμική Διακριτική Ανάλυση (LDA-Linear Discriminant Analysis)**

   Η LDA πέτυχε ακρίβεια 77.86%, με ευαισθησία 87.23%, ειδικότητα 50.00%, PPV 79.31% και NPV 57.89%. Η LDA, παρ' όλη την απλότητά της, αποδείχθηκε λιγότερο ικανή για την ταξινόμηση δεδομένων που αποκλίνουν από γραμμικά πρότυπα ή κανονικές κατανομές [105]. Η LDA εξακολουθεί να είναι μια δημοφιλής τεχνική λόγω της απλότητάς της, ωστόσο η παρούσα μελέτη αποκαλύπτει τις αδυναμίες της στο χειρισμό πολύπλοκων ιατρικών δεδομένων [132].

3. **Τετραγωνική Διακριτική Ανάλυση (QDA-Quadratic Discriminant Analysis)**

   Η QDA πέτυχε ακρίβεια 76.56%, με ευαισθησία 87.23%, ειδικότητα 46.67%, PPV 78.95% και NPV 55.00%. Αν και πιο ισχυρή από την LDA, η QDA δεν είχε την ικανότητα απόδοσης έναντι της GPC. Αν και εξακολουθεί να είναι περιορισμένη, η ικανότητά της να αναπαράγει μη γραμμικές αλληλεπιδράσεις δείχνει πρόοδο σε σχέση με την LDA [106].

4. **Naive Bayes (NB)**

   Η μέθοδος Naive Bayes (NB) πέτυχε ακρίβεια 77,57%, ευαισθησία 85,11%, ειδικότητα 53,33%, PPV 80,00% και NPV 54,55%. Οι απαιτήσεις ανεξαρτησίας του NB περιορίζουν την απόδοσή του σε πιο περίπλοκα δεδομένα, παρόλο που είναι γρήγορος και οικονομικός σε βασικά σενάρια [104]. Η μελέτη υπογραμμίζει την ανάγκη πιο εξελιγμένων μεθόδων για δεδομένα που εμφανίζουν αλληλεξάρτηση χαρακτηριστικών.

5.1.2 Αναλυτική Επισκόπηση

Δίνοντας έμφαση στα πλεονεκτήματα του μοντέλου GPC έναντι των συμβατικών τεχνικών, η εργασία δείχνει την αξία της μηχανικής μάθησης για την ταξινόμηση ιατρικών δεδομένων. Το GPC με τον πυρήνα RBF είναι πολύ χρήσιμο για την κατηγοριοποίηση ιατρικών παθήσεων όπως ο διαβήτης [107], καθώς μπορεί να προσαρμόσει καλύτερα τα μη γραμμικά δεδομένα, βελτιώνοντας έτσι την απόδοσή του.

Η εκτενής σύγκριση με άλλες τεχνικές, όπως η LDA, η QDA και η Naive Bayes, υπογραμμίζει τα όρια και τους περιορισμούς αυτών των συμβατικών προσεγγίσεων όταν χρησιμοποιούνται σε περίπλοκα ιατρικά δεδομένα, τονίζοντας επομένως την ανάγκη για πιο εξελιγμένες τεχνικές όπως η GPC.

Η επιλογή της κατάλληλης K μπορεί να συμβάλει στη βελτίωση της ακρίβειας ταξινόμησης, η οποία είναι ζωτικής σημασίας για την ιατρική κοινότητα και τη διάγνωση του διαβήτη. Η χρήση τεχνικών διασταυρούμενης αξιολόγησης K5 και K10 βοηθά στην αύξηση της αξιοπιστίας των αποτελεσμάτων, μέσω των οποίων μπορεί να επιτευχθεί αυτό [103].

Με προφανείς συνέπειες για την πρόληψη και τη θεραπεία της πάθησης, η εφαρμογή αυτής της έρευνας μπορεί να οδηγήσει σε πιο αξιόπιστες και ακριβείς μεθόδους για την έγκαιρη αναγνώριση του διαβήτη [132].

5.2 Δεύτερη Μελέτη: "Prediction of Type 2 Diabetes using Machine Learning Classification Methods" by N. Tigga

Έξι διαφορετικές τεχνικές μηχανικής μάθησης για την πρόβλεψη του κινδύνου διαβήτη τύπου 2 διερευνώνται σε αυτή την εργασία από τον N. Tigga και τους συνεργάτες του [109]. Σκοπός της εργασίας είναι να αξιολογηθεί η δυνατότητα γενίκευσης αυτών των αλγορίθμων σε νέα δεδομένα και η διακριτική ικανότητα (discriminability), καθορίζοντας έτσι την απόδοσή τους. Οι αλγόριθμοι που εφαρμόζονται είναι οι εξής:

- Τυχαία Δάση (Random Forest, RF)
- Λογιστική Παλινδρόμηση (Logistic Regression, LR)
- k-Πλησιέστεροι Γείτονες (k-Nearest Neighbors, k-NN)
- Μηχανές Διανυσμάτων Υποστήριξης (Support Vector Machines, SVM)
- Naive Bayes (NB)
- Δέντρα Αποφάσεων (Decision Trees, DT)

Χρησιμοποιήθηκαν δύο βάσεις δεδομένων. Η πρώτη, είχε 952 εγγραφές, εκ των οποίων 580 αρσενικά και 372 θηλυκά, όλα τα άτομα άνω των 18 ετών. Τα στοιχεία και τα χαρακτηριστικά που υπήρχαν ήταν:

Ηλικία: 18 και άνω

Φύλο: Αρσενικά = 580/ Θηλυκά = 372

Οικογενειακό ιστορικό με διαβήτη: Ναι/Όχι

Διάγνωση με υψηλή τιμή πίεσης: Ναι/ Όχι

Περπάτημα/Τρέξιμο/Φυσική δραστηριότητα: καθόλου/κάτω από μισή ώρα/πάνω από μισή ώρα/πάνω από 1 ώρα.

BMI: Αριθμός

Κάπνισμα: Ναι/ Όχι

Κατανάλωση Αλκοόλ: Ναι/ Όχι

Ώρες ύπνου: Αριθμός

Ώρες υγιούς ύπνου: Αριθμός

Τακτική λήψη φαρμάκων: Ναι/ Όχι

Κατανάλωση ανθυγιεινού φαγητού: Ναι/ Όχι

Το άλλο σύνολο δεδομένων που χρησιμοποιήθηκε είναι το **PIMA Indians Diabetes Dataset**, το οποίο περιλαμβάνει 768 εγγραφές γυναικών, με 9 χαρακτηριστικά όπως το επίπεδο γλυκόζης, η πίεση αίματος, ο δείκτης μάζας σώματος (BMI), η ηλικία και άλλα.

5.2.1 Μέθοδοι

Συλλογή και Προεπεξεργασία Δεδομένων:
Αποτελούμενο από 768 δείγματα, το σύνολο δεδομένων PIMA Indians Diabetes αντιπροσωπεύει γυναίκες ηλικίας τουλάχιστον 21 ετών. Τα χαρακτηριστικά αυτής της βάσης δεδομένων αναφέρθηκαν στο προηγούμενο κεφάλαιο και το σύνολο δεδομένων που περιλαμβάνει τα αναφερόμενα χαρακτηριστικά και 952 εγγραφές άνω των 18 ετών που χρησιμοποιήθηκαν στην παρούσα έρευνα.

Για την αποφυγή εσφαλμένων και ακραίων αποτελεσμάτων, η προεπεξεργασία δεδομένων συνίσταται στον καθαρισμό των δεδομένων και στην αντικατάσταση των δεδομένων για τις ελλείπουσες τιμές.

Εκπαίδευση και Δοκιμή:
Τα δεδομένα χωρίστηκαν σε δύο σύνολα:

- **Εκπαίδευση**: Οι αλγόριθμοι εκπαιδεύτηκαν χρησιμοποιώντας το 75% των δεδομένων.
- **Δοκιμή**: Τα μοντέλα δοκιμάστηκαν και αξιολογήθηκαν οι επιδόσεις τους χρησιμοποιώντας το 25% των δεδομένων.

Οι επιδόσεις των αλγορίθμων αξιολογήθηκαν με τη χρήση **K-fold** cross-validation με τιμή K=10. Η προσέγγιση αυτή εγγυάται ότι τα μοντέλα αξιολογούνται σε διάφορα υποσύνολα δεδομένων, αυξάνοντας έτσι τη γενικευσιμότητα των αποτελεσμάτων.

5.2.2 Αποτελέσματα
Τα αποτελέσματα που είχαμε στην συγκεκριμένη μελέτη για κάθε μοντέλο μηχανικής μάθησης που χρησιμοποιήθηκε περιγράφονται παρακάτω.

Τυχαία Δάση (RF)
Η βάση δεδομένων της μελέτης πέτυχε 94.10% ακρίβεια σε σχέση με το PIMA που πέτυχε 75%. Το RF παρουσίασε την υψηλότερη ακρίβεια, επιτυγχάνοντας εξαιρετική απόδοση στην ταξινόμηση. Το AUC (Περιοχή Υπό την Καμπύλη ROC) πέτυχε 1.0 και στις 2 βάσεις δεδομένων, την υψηλότερη διακριτική ικανότητα, υποδεικνύοντας τέλεια ικανότητα διάκρισης μεταξύ των θετικών και αρνητικών περιπτώσεων. Η μέτρηση Κάππα στην πρώτη βάση ήταν 0.992 και στο PIMA 0.488. Στην πρώτη βάση υπήρξε υψηλή συμφωνία με τα πραγματικά αποτελέσματα, επιβεβαιώνοντας την αξιοπιστία των προβλέψεων. Το RF μπορεί να διαχειριστεί δεδομένα με πολλές μεταβλητές, μειώνοντας έτσι τον κίνδυνο υπερπροσαρμογής.

Λογιστική Παλινδρόμηση (LR)

Στη Λογιστική Παλινδρόμηση, η βάση δεδομένων της μελέτης πέτυχε 85.70% ακρίβεια σε σχέση με το PIMA που πέτυχε 74.40%.

Το LR παρουσίασε καλή απόδοση, χωρίς να ξεπερνά βεβαίως το Random Forest. Το AUC (Περιοχή Υπό την Καμπύλη ROC) πέτυχε 0.908 και 0.765 στις 2 βάσεις δεδομένων αντίστοιχα, με υψηλή διαχωριστικότητα, επιτρέποντας καλές προβλέψεις. Η μέτρηση Κάππα

στην πρώτη βάση ήταν 0.727 και στο PIMA 0.470. Η LR είναι πιο απλή και εύκολη στην ερμηνεία, αλλά δεν έχει την ίδια απόδοση σε πιο σύνθετα προβλήματα όπως το RF [109].

k-Πλησιέστεροι Γείτονες (k-NN)

Στον αλγόριθμο με Κ-Πλησιέστερους Γείτονες, η βάση δεδομένων της μελέτης πέτυχε 77.30% ακρίβεια σε σχέση με το PIMA που πέτυχε 70.80%. Ο αλγόριθμος πέτυχε αρκετά χαμηλότερη απόδοση από τα υπόλοιπα μοντέλα. Το AUC (Περιοχή Υπό την Καμπύλη ROC) πέτυχε 0.916 και 0.815 στις 2 βάσεις δεδομένων αντίστοιχα. Η μέτρηση Κάππα στην πρώτη βάση ήταν 0.516 και στο PIMA 0.419.

Μηχανές Διανυσμάτων Υποστήριξης (SVM)

Στον αλγόριθμο Support Vector Machine (SVM), η βάση δεδομένων της μελέτης πέτυχε 86.50% ακρίβεια σε σχέση με το PIMA που πέτυχε 74.40%. Ο αλγόριθμος πέτυχε καλύτερη απόδοση από τα υπόλοιπα μοντέλα, αλλά πιο χαμηλή από το RF . Το AUC (Περιοχή Υπό την Καμπύλη ROC) πέτυχε 0.893 και 0.771 στις 2 βάσεις δεδομένων αντίστοιχα. Η μέτρηση Κάππα στην πρώτη βάση ήταν 0.713 και στο PIMA 0.466. Το SVM αποδίδει καλά σε δεδομένα υψηλής διάστασης και είναι ικανό να διαχωρίσει τις κατηγορίες με καλή ακρίβεια [109].

Naive Bayes(NB)

Στον αλγόριθμο Naive Bayes (NB), η βάση δεδομένων της μελέτης πέτυχε 80.60% ακρίβεια σε σχέση με το PIMA που πέτυχε 68.90%. Ο αλγόριθμος δεν πέτυχε από τις καλύτερες αποδόσεις σε σύγκριση με τα υπόλοιπα μοντέλα. Το AUC (Περιοχή Υπό την Καμπύλη ROC) πέτυχε 0.857 και 0.760 στις 2 βάσεις δεδομένων αντίστοιχα. Η μέτρηση Κάππα στην πρώτη βάση ήταν 0.638 και στο PIMA 0.447. Ο αλγόριθμος είναι εύκολος στην εκπαίδευση, αλλά η απόδοσή του επηρεάζεται από την υπόθεση ανεξαρτησίας των χαρακτηριστικών.

Δέντρα Αποφάσεων (DT)

Στον αλγόριθμο των Δέντρων Αποφάσεων, η βάση δεδομένων της μελέτης πέτυχε 84.00% ακρίβεια σε σχέση με το PIMA που πέτυχε 69.70%. Ο αλγόριθμος πέτυχε καλή απόδοση, όχι τόσο όσο άλλα μοντέλα βέβαια. Το AUC (Περιοχή Υπό την Καμπύλη ROC) πέτυχε 0.916 και 0.842 στις 2 βάσεις δεδομένων αντίστοιχα που δείχνει καλή διαχωριστικότητα. Η μέτρηση Κάππα στην πρώτη βάση ήταν 0.646 και στο PIMA 0.422. Ο αλγόριθμος είναι εύκολος στην ερμηνεία, αλλά απαιτεί προσοχή για την αποφυγή υπερ-προσαρμογής.

5.2.3 Συμπεράσματα

Η μελέτη καταδεικνύει την αποδοτικότητα των αλγορίθμων μηχανικής μάθησης για την πρόβλεψη του κινδύνου εμφάνισης διαβήτη τύπου 2. Με τη μεγάλη ακρίβεια, τη διαχωρισιμότητα και τη δυνατότητα γενίκευσης στις δύο βάσεις που διερευνήθηκαν, η μέθοδος Random Forest (RF) ξεχώρισε. Αν και δεν ξεπέρασαν το RF, τα δέντρα απόφασης (Decision Trees, DT) και η λογιστική παλινδρόμηση (Logistic Regression, LR) είχαν αρκετά καλές επιδόσεις. Αντίθετα, λόγω των περιορισμών τους με την υπόθεση ανεξαρτησίας και τη διαχείριση του θορύβου, οι αλγόριθμοι Naïve Bayes (NB) και k-NN είχαν αισθητά χαμηλότερη επίδοση [109].

Η μελέτη υπογραμμίζει την ανάγκη μεγαλύτερης έρευνας και βελτίωσης των αλγορίθμων πρόβλεψης του διαβήτη για τη μεγιστοποίηση των μοντέλων και τη χρήση πιο σύνθετων στρατηγικών ανάλυσης και διαχείρισης δεδομένων.

5.2.4 Συγκριτική Ανασκόπηση των Μοντέλων

- **Λογιστική Παλινδρόμηση (LR)**: Αποτελεσματική για τα βασικά μοντέλα, η λογιστική παλινδρόμηση (LR) περιορίζεται σε πιο περίπλοκα δεδομένα.
- **Μηχανές Διανυσμάτων Υποστήριξης (SVM)**: Ικανές να χειριστούν δεδομένα υψηλής διάστασης με καλή διαχωριστικότητα.
- **Naive Bayes (NB)**: Απλές στη χρήση, αλλά περιορισμένες λόγω της παραδοχής της ανεξαρτησίας των χαρακτηριστικών.
- **Δέντρα Αποφάσεων (DT)**: Ευκολία στην ερμηνεία, αλλά απαιτούν σωστή προσαρμογή για την αποφυγή υπερπροσαρμογής.
- **Τυχαία Δάση (RF)**: Εξαιρετική γενίκευση και υψηλή ακρίβεια.
- **k-Πλησιέστεροι Γείτονες (k-NN)**: Εξαρτάται από την επιλογή του k και την καλή διαχείριση του θορύβου.

5.3 Τρίτη Μελέτη: "A Comparison of Machine Learning Algorithms for Diabetes Prediction" by Jobeda Jamal Khanam and Simon Y. Foo

Η μελέτη αυτή, που εκπονήθηκε από τους Jobeda Jamal Khanam και Simon Y. Foo και τους συνεργάτες τους [110], έχει ως στόχο την αξιολόγηση και σύγκριση διαφόρων αλγορίθμων μηχανικής μάθησης για την πρόβλεψη του διαβήτη, χρησιμοποιώντας το σύνολο δεδομένων PIMA Indians Diabetes (PID) και την εφαρμογή μηχανικής μάθησης WEKA, που είναι open-source. Οι αλγόριθμοι που εξετάστηκαν περιλαμβάνουν:

- **Πολυωνυμική Παλινδρόμηση (Logistic Regression, LR)**
- **Υποστήριξη Διανυσμάτων (Support Vector Machine, SVM)**
- **Αλγόριθμος Naive Bayes (NB)**
- **Δέντρα Απόφασης (Decision Trees, DT)**
- **Τυχαίο Δάσος (Random Forest, RF)**
- **Κοντινότεροι Γείτονες (K-Nearest Neighbors, KNN)**
- **Ενίσχυση Προσαρμοστικότητας (Adaptive Boosting, AB)**
- **Νευρωνικά Δίκτυα (Neural Networks, NN)**

5.3.1 Μέθοδοι

1. Συλλογή Δεδομένων και Προεπεξεργασία

Το σύνολο δεδομένων PID αντλήθηκε από το **UCI Machine Learning Repository**, το οποίο περιλαμβάνει 768 εγγραφές γυναικών με 9 χαρακτηριστικά όπως έχει αναφερθεί και σε προηγούμενες μελέτες.

Για την προεπεξεργασία των δεδομένων, χρησιμοποιήθηκαν οι εξής τεχνικές:

- **Αντικατάσταση Ελλειπόντων Τιμών:** Τα δεδομένα που λείπουν αντικαταστάθηκαν με τη μέση τιμή.
- **Ανίχνευση και Αφαίρεση Εξωφύλλων:** Εξώφυλλα εντοπίστηκαν και αφαιρέθηκαν χρησιμοποιώντας το εργαλείο WEKA.
- **Κανονικοποίηση:** Τα χαρακτηριστικά κανονικοποιήθηκαν για να διασφαλιστεί η ομοιογένεια στη κλίμακα από το εύρος 0 έως 1, η οποία αύξησε την ταχύτητα υπολογισμού του αλγορίθμου.

2. Επιλογή Χαρακτηριστικών

Η **Μέθοδος Συσχέτισης Pearson** χρησιμοποιήθηκε για την αναγνώριση των πιο σημαντικών χαρακτηριστικών που σχετίζονται με την πρόβλεψη του διαβήτη. Η ανάλυση αυτή δείχνει τα χαρακτηριστικά που σχετίζονται με την πιθανότητα εμφάνισης διαβήτη. Τα χαρακτηριστικά με συντελεστή συσχέτισης μεγαλύτερο από 0.2 επιλέχθηκαν για το μοντέλο [110].

Τα χαρακτηριστικά που βρέθηκαν να έχουν ισχυρή συσχέτιση με την πιθανότητα εμφάνισης διαβήτη περιλαμβάνουν τη Γλυκόζη, το BMI, την ινσουλίνη, την Κύηση (Preg) και την ηλικία.

Αυτά τα χαρακτηριστικά θεωρήθηκαν σημαντικά και επιλέχθηκαν για τη δημιουργία των αλγορίθμων μηχανικής μάθησης.

3. Εκπαίδευση και Δοκιμή

Η **διασταυρούμενη επικύρωση (K-Fold Cross-Validation)** χρησιμοποιήθηκε για την αξιολόγηση της γενικότητας του μοντέλου, με 7-folds. Τα δεδομένα χωρίστηκαν σε 7 ίσα μέρη, και κάθε μέρος χρησιμοποιήθηκε ως σύνολο δοκιμής μία φορά, ενώ τα υπόλοιπα 6 μέρη χρησιμοποιήθηκαν για εκπαίδευση. Αυτή η διαδικασία επέτρεψε την εκτίμηση της απόδοσης του μοντέλου και τη μείωση του κινδύνου υπερ-προσαρμογής (overfitting). Η **διαίρεση των δεδομένων** για εκπαίδευση και δοκιμή έγινε με ποσοστό 85% για εκπαίδευση και 15% για δοκιμή.

5.3.2 Αναλυτική Συγκριτική Ανασκόπηση των Μοντέλων

Παρακάτω, αναλύονται τα αποτελέσματα που προέκυψαν κατά τη διάρκεια της μελέτης στα μοντέλα που μελετήθηκαν.

1. Logistic Regression (Λογιστική Παλινδρόμηση):

Η λογιστική παλινδρόμηση εφαρμόζει γραμμική συνάρτηση για δυαδική ταξινόμηση, προβλέποντας την πιθανότητα εμφάνισης του διαβήτη. Η ακρίβεια που πέτυχε στη συγκεκριμένη μελέτη είναι 76.82% για το K-fold cross validation και 78.85% στη μέθοδο διαχωρισμού Train/Test. Είναι γνωστή για την απλότητα, στην εύκολη ερμηνεία της και στην γρήγορη εκπαίδευση. Από την άλλη, έχει περιορισμένη απόδοση σε μη γραμμικά δεδομένα.

2. Support Vector Machine (SVM):

Ο Support Vector Machine δημιουργεί το βέλτιστο υπερεπίπεδο που διαχωρίζει τις κατηγορίες. Στην μελέτη αυτή, η ακρίβεια που πέτυχε στη συγκεκριμένη μελέτη είναι 76.82% για το K-fold cross validation και 77.71% στη μέθοδο διαχωρισμού Train/Test. Είναι γνωστή για την ικανότητα διαχείρισης μη γραμμικών δεδομένων με πυρήνες και είναι αποτελεσματικός σε υψηλής διάστασης χώρους. Βέβαια, έχει υψηλές απαιτήσεις υπολογιστικής ισχύος και δυσκολία στην επιλογή πυρήνα και παραμέτρων [110].

3. Naive Bayes:

Ο Naive Bayes εφαρμόζει το θεώρημα του Μπέιζ με την υπόθεση ανεξαρτησίας των χαρακτηριστικών. Πέτυχε ακρίβεια 75.53% για το K-fold cross validation και 78.28% στη μέθοδο διαχωρισμού Train/Test. Είναι γνωστή για την γρήγορη εκπαίδευση και την καλή απόδοση σε μεγάλα σύνολα δεδομένων. Από την άλλη, υποθέτει ανεξαρτησία των χαρακτηριστικών και είναι ευαίσθητος σε χαρακτηριστικά με χαμηλή πιθανότητα.

4. Decision Trees (Δέντρα Απόφασης):

Τα δέντρα απόφασης, δημιουργούν ένα μοντέλο διάσπασης των δεδομένων σε βάθος. Η ακρίβεια που πέτυχε στη συγκεκριμένη μελέτη είναι 74.24% για το K-fold cross validation και 73.14% στη μέθοδο διαχωρισμού Train/Test.. Είναι εύκολη στην ερμηνεία, χωρίς να χρειάζεται κανονικοποίηση. Στα αρνητικά του αλγορίθμου είναι ότι είναι επιρρεπή στην υπερπροσαρμογή και στην ευαισθησία στις μικρές αλλαγές στα δεδομένα.

5. Random Forest (Τυχαία Δάση):

Ο αλγόριθμος Random Forest χρησιμοποιεί συλλογή από δέντρα απόφασης για τη βελτίωση της ακρίβειας και της σταθερότητας.Η ακρίβεια που πέτυχε είναι 74.96% για το K-fold cross validation και 77.14% στη μέθοδο διαχωρισμού Train/Test. Είναι γνωστή για την αντοχή στην υπερπροσαρμογή και έχει καλή απόδοση σε μεγάλα σύνολα δεδομένων. Βέβαια, έχει υψηλότερο κόστος υπολογισμού και είναι πιο δύσκολη στην ερμηνεία από άλλους αλγορίθμους.

6. k-Nearest Neighbors (k-NN):

Στον αλγόριθμο με Κ-κοντινότεροι γείτονες, η ακρίβεια που πέτυχε είναι 75.10% για το K-fold cross validation και 79.42% στη μέθοδο διαχωρισμού Train/Test.

7. **Adaptive Boosting (AB)*:***

Στη μελέτη, ο αλγόριθμος Adaptive Boosting πέτυχε ακρίβεια 73.96% για το K-fold cross validation και 79.42% στη μέθοδο διαχωρισμού Train/Test.

8. Neural Networks (Νευρωνικά Δίκτυα):

Τα νευρωνικά δίκτυα είναι ένα μοντέλο μηχανικής μάθησης, που αποτελείται από μη γραμμικές σχέσεις μέσω σύνθετων στρωμάτων. Στη συγκεκριμένη μελέτη, δημιουργήθηκε νευρωνικό δίκτυο με το πρώτο και πέμπτο στρώμα να είναι στρώματα εισόδου και εξόδου, που είχαν το ίδιο σχήμα εισόδου, τους ίδιους νευρώνες και τη συνάρτηση ενεργοποίησης όπως το NN με ένα κρυφό στρώμα. Το δεύτερο, τρίτο και τέταρτο στρώμα, ήταν τα κρυφά στρώματα και είχαν 16,10,5 νευρώνες αντίστοιχα [110]. Στη μελέτη, υπολογίστηκαν οι ακρίβειες για 1,2,3 κρυφά στρώματα. Διαπιστώθηκε ότι το μοντέλο NN με δύο κρυφά στρώματα και 400 epochs και με ρυθμό μάθησης 0.01 πέτυχε 88.6% ακρίβεια. Το νευρωνικό δίκτυο έχει ικανότητα να μαθαίνει πολύπλοκες σχέσεις και πρότυπα. Από την άλλη, έχει υψηλές απαιτήσεις υπολογιστικών πόρων και κρύβει κινδύνους υπερπροσαρμογής.

5.3.4 Συμπεράσματα

Η μελέτη καταλήγει στο συμπέρασμα ότι τα **Νευρωνικά Δίκτυα** προσφέρουν την υψηλότερη απόδοση για την πρόβλεψη του διαβήτη με ακρίβεια 88.6%. Ωστόσο, και άλλοι αλγόριθμοι που μελετήθηκαν προσφέρουν επίσης ισχυρές επιδόσεις και ενδέχεται να είναι προτιμότεροι σε περιβάλλοντα με περιορισμένους υπολογιστικούς πόρους ή μικρότερα δεδομένα [110].

5.4 Τέταρτη Μελέτη: "An Ensemble Approach for Classification and Prediction of Diabetes Mellitus Using Soft Voting Classifier" by Saloni Kumari, Deepika Kumar and Mamta Mittal

Η μελέτη αυτή, που πραγματοποιήθηκε από τους Kumari, Kumar και Mittal, εισάγει μια καινοτόμο προσέγγιση για την πρόβλεψη και ταξινόμηση του διαβήτη, χρησιμοποιώντας έναν **soft voting classifier**. Με στόχο την ενίσχυση της απόδοσης της ταξινόμησης του διαβήτη σε σχέση με τη χρήση μεμονωμένων αλγορίθμων, η έρευνα εστιάζει στην εφαρμογή προσεγγίσεων συνδυασμού αλγορίθμων μηχανικής μάθησης [132]. Ο ταξινομητής soft voting βασίζεται στη θεωρία ότι η συνδυαστική τεχνική μπορεί να μειώσει την πιθανότητα σφαλμάτων υπερπροσαρμογής και να βελτιώσει τη γενίκευση του μοντέλου, επιτυγχάνοντας έτσι καλύτερα αποτελέσματα στη διαφοροποίηση θετικών και αρνητικών περιπτώσεων διαβήτη.

Η εργασία αναλύει την απόδοση διαφόρων μοντέλων κατηγοριοποίησης τόσο μεμονωμένα όσο και στο πλαίσιο του soft-voting, παρουσιάζοντας έτσι μια εξαντλητική συγκριτική αξιολόγηση μεταξύ τους. Η εργασία αυτή αποσκοπεί στη βελτίωση της διαγνωστικής ικανότητας των μοντέλων για την πρόβλεψη της νόσου του διαβήτη με τη χρήση συνδυαστικών προσεγγίσεων.

5.4.1 Μοντέλα που Χρησιμοποίησαν οι Ερευνητές

Στην εργασία εφαρμόζονται οκτώ διαφορετικές τεχνικές μηχανικής μάθησης. Για την αξιολόγηση αυτών των μοντέλων, ελήφθησαν τα αποτελέσματά τους ξεχωριστά καθώς και σε σχέση με τον soft-voting classifier.

1. Logistic Regression (Λογιστική Παλινδρόμηση)

Ο αλγόριθμος υπολογίζει την πιθανότητα το δείγμα να ανήκει στη θετική κλάση (διαβήτης), όπως σημειώνεται στην ανάλυση του αλγορίθμου. Το δείγμα λέγεται ότι είναι θετικό όταν η πιθανότητα είναι μεγαλύτερη από ένα καθορισμένο επίπεδο, συνήθως 0,5. Η μέθοδος πέτυχε στην παρούσα εργασία ακρίβεια 74.89%, Precision 64.47%, Recall 61.25% και F1-score 62.82%. Το AUC ήταν 80.10%. Αν και η soft voting βοηθά στην αύξηση της ακρίβειας, η λογιστική παλινδρόμηση αποδεικνύεται μάλλον επιτυχής [132].

2. Naive Bayes (NB)

Ο Naive Bayes βασίζεται στο Θεώρημα του Bayes και υποθέτει ότι τα χαρακτηριστικά είναι ανεξάρτητα μεταξύ τους. Είναι αποδοτικό σε καταστάσεις με μεγάλο αριθμό χαρακτηριστικών και χαμηλή επεξεργαστική ισχύ. Στη μελέτη αυτή, ο αλγόριθμος πέτυχε ακρίβεια 74.12%, Precision 61.90%, Recall 65%, F1-score 63.41% και AUC 79.01%. Παρόλο που αυτή η προσέγγιση παράγει καλά αποτελέσματα, η αδυναμία της να λάβει υπόψη τις εξαρτήσεις μεταξύ των χαρακτηριστικών μειώνει την ακρίβεια σε σύγκριση με άλλες προσεγγίσεις.

3. Random Forest (Τυχαία Δάση)

Ο αλγόριθμος Random Forest αποτελείται από πολλά δέντρα απόφασης. Κάθε δέντρο απόφασης δίνει μια πρόβλεψη και η τελική απόφαση προκύπτει από τη μέση τιμή ή την πλειοψηφία των προγνωστικών αποφάσεων όλων των δέντρων. Στη μελέτη αυτή, ο

αλγόριθμος πέτυχε ακρίβεια 77.48%, Precision 71.21%, Recall 58.75% και F1-score 64.38%. Το AUC ήταν 78.10%.

Η καλύτερη απόδοση από τη λογιστική παλινδρόμηση και το Naive Bayes που παρέχει το Random Forest βοηθά στη μείωση του κινδύνου υπερπροσαρμογής και στην ενίσχυση της γενίκευσης [132].

4. Support Vector Machine (SVM)

Ο αλγόριθμος SVM προσπαθεί να βρει το υπερεπίπεδο (hyperplane) που χωρίζει τις δύο κατηγορίες δεδομένων με τη μεγαλύτερη δυνατή απόσταση (margin). Στη μελέτη αυτή, ο αλγόριθμος SVM πέτυχε ακρίβεια 74.02%, Precision 67.24%, Recall 48.75% και F1-score 56.52%.

5. AdaBoost

Συνδυάζοντας πολυάριθμους αδύναμους ταξινομητές, η προσέγγιση AdaBoost (adaptive Boosting) παράγει έναν ισχυρό ταξινομητή. Κάθε επόμενος ταξινομητής μαθαίνει χρησιμοποιώντας τα λάθη των προηγούμενων. Στη μελέτη αυτή, ο αλγόριθμος πέτυχε ακρίβεια 75.32%, Precision 68.25%, Recall 53.75%, F1-score 60.13% και τιμή AUC 74.98%.

6. Gradient Boosting

Η μέθοδος **Gradient Boosting** δημιουργεί έναν ισχυρό ταξινομητή μέσω της συνεχούς προσθήκης νέων δέντρων απόφασης, όπου κάθε νέο δέντρο διορθώνει τα σφάλματα των προηγούμενων. Στη μελέτη αυτή, ο αλγόριθμος πέτυχε ακρίβεια 75.32%, Precision 70.90%, Recall 48.75% και F1-score 57.77%. Η τιμή AUC είναι 71.89%.

7. XGBoost

Πρόκειται για μια βελτιωμένη έκδοση του **Gradient Boosting**, με έμφαση στη βελτιστοποίηση της απόδοσης και της ταχύτητας. Ο συγκεκριμένος αλγόριθμος εκμεταλλεύεται τις τεχνικές παραλληλίας και βελτιστοποίησης για να επιταχύνει την εκπαίδευση και να βελτιώσει την απόδοση. Στη μελέτη αυτή, ο αλγόριθμος πέτυχε ακρίβεια 75.75%, Precision 64.28%, Recall 67.50%, F1-score 65.85% και τιμή AUC 69.01%.

8. CatBoost

Ο αλγόριθμος **CatBoost** είναι ειδικά σχεδιασμένος για να χειρίζεται κατηγορικές μεταβλητές χωρίς να απαιτεί κωδικοποίηση one-hot. Χρησιμοποιεί προηγμένες τεχνικές για την κατηγοριοποίηση κατηγορικών χαρακτηριστικών, μειώνοντας την ανάγκη για προεπεξεργασία. Στη μελέτη αυτή, ο αλγόριθμος πέτυχε ακρίβεια 75.32%, Precision 64.19%, Recall 65%, F1-score 64.59% και τιμή AUC 74.56%.

5.4.2 Προτεινόμενη Μεθοδολογία: Soft Voting Classifier

Η μεθοδολογία **Soft Voting Classifier** αποτελεί μια συνδυαστική τεχνική, η οποία συνδυάζει τις προβλέψεις από διάφορα μοντέλα μέσω του μέσου όρου των πιθανοτήτων. Το **Soft Voting** υπολογίζει τη μέση πιθανότητα από τις προβλέψεις των μοντέλων για να καθορίσει την τελική κατηγορία. Για κάθε δείγμα, οι πιθανότητες των μοντέλων **Logistic Regression**,

Naive Bayes, και **Random Forest** συγκεντρώνονται και συνδυάζονται μέσω του μέσου όρου για να παραχθεί η τελική πρόβλεψη [132].

5.4.3 Αποτελέσματα της Προτεινόμενης Μεθοδολογίας

Η εφαρμογή του **Soft Voting Classifier** στα δεδομένα του **PIMA Indian Diabetes Dataset** δίνει τα εξής αποτελέσματα:

- **Accuracy**: 79.08% - Υψηλότερη απόδοση σε σχέση με τα μεμονωμένα μοντέλα, δείχνοντας βελτίωση στην ακρίβεια.
- **Precision**: 73.13% - Υψηλότερη από την precision των επιμέρους μοντέλων, προσφέροντας καλύτερη ακρίβεια στις θετικές προβλέψεις.
- **Recall**: 70% - Υψηλότερη από την recall των μεμονωμένων μοντέλων, εντοπίζοντας περισσότερες πραγματικές θετικές περιπτώσεις.
- **F1-score**: 71.56% - Σημαντική βελτίωση στην ισορροπία μεταξύ precision και recall, προσφέροντας υψηλή συνολική απόδοση.

5.4.4 Συμπεράσματα

Η καλύτερη γενίκευση και απόδοση που παρέχει η συνδυασμένη τεχνική soft voting σε σχέση με τα μεμονωμένα μοντέλα βελτιώνει τη διαγνωστική ικανότητα του συστήματος. Τα εξαιρετικά αποτελέσματα σε δεδομένα που σχετίζονται με τον διαβήτη που παράγει αυτή η μέθοδος υποδηλώνουν την πρακτική της χρήση. Η διερεύνηση διαφόρων συνδυασμών μοντέλων και η χρήση αυτής της προσέγγισης σε νέες βάσεις δεδομένων μπορεί να βοηθήσει στην ανάδειξη ευκαιριών για ανάπτυξη και έρευνα [132].

5.5 Πέμπτη Μελέτη: "Diabetes Prediction Using Ensembling of Different Machine Learning Classifiers" by Kamrul Hasan

Χρησιμοποιώντας το σύνολο δεδομένων PIMA Indian Diabetes (PID), ο Kamrul Hasan και οι συνεργάτες του εξετάζουν την εφαρμογή και τη σύγκριση πολλών αλγορίθμων μηχανικής μάθησης για την πρόβλεψη του διαβήτη [113]. Η εργασία αυτή ασχολείται με την προετοιμασία των δεδομένων, την εκπαίδευση των μοντέλων, την τεχνική ενσωμάτωσης και την απόδοση σύγκρισης πολλών μοντέλων. Παρέχει μια πληρέστερη συζήτηση για τη μεθοδολογία, τα μοντέλα που χρησιμοποιήθηκαν, τα αποτελέσματα της έρευνας και τα ευρήματα της μελέτης βάσει δεδομένων.

5.5.1 Μεθοδολογία

Η μέθοδος που εφαρμόστηκε κατά τη διάρκεια της έρευνας περιγράφεται επακριβώς βήμα προς βήμα παρακάτω.

- **Προεπεξεργασία Δεδομένων**
    - **Αφαίρεση Ακραίων Τιμών**: Η εξάλειψη των ακραίων τιμών συμβάλλει στην αποφυγή λανθασμένων ή κάποιων σπάνιων γεγονότων που θα μπορούσαν να αλλοιώσουν τα δεδομένα. Μεταξύ των τεχνικών που εφαρμόστηκαν για τον εντοπισμό των ακραίων τιμών ήταν τα διαγράμματα διασποράς και το Z-score.
    - **Αντικατάσταση Ελλιπών Δεδομένων**: Η μέθοδος της πλήρωσης ελλιπών δεδομένων με την μέση τιμή (mean imputation) χρησιμοποιήθηκε για να

διατηρηθεί η στατιστική ακεραιότητα των δεδομένων και να αποφευχθούν προβλήματα κατά την εκπαίδευση του μοντέλου.

- **Κανονικοποίηση Δεδομένων**: Κάθε χαρακτηριστικό είχε μέσο όρο 0 και μοναδιαία διακύμανση που εξασφαλίστηκε με την τυποποίηση Z-score, εξαλείφοντας έτσι τα προβλήματα από πολλά επίπεδα χαρακτηριστικών.
- **Επιλογή Χαρακτηριστικών**: Η επιλογή των χαρακτηριστικών πραγματοποιήθηκε με τη χρήση των τεχνικών Principal Component Analysis (PCA) και Independent Component Analysis (ICA), οι οποίες βοηθούν στη μείωση της διάστασης και στην απομόνωση των πιο σημαντικών χαρακτηριστικών για την πρόβλεψη του διαβήτη [113].

- **Διασταυρωμένη Επαλήθευση (K-fold Cross-Validation)**

  Η K-fold cross-validation με K=5, αξιολογήθηκε η απόδοση του μοντέλου, εξασφαλίζοντας έτσι ακριβείς εκτιμήσεις και μειώνοντας τον κίνδυνο υπερπροσαρμογής. Υπήρχαν πέντε υποομάδες των δεδομένων και κάθε μοντέλο αξιολογήθηκε τέσσερις φορές σε διαφορετικά υποσύνολα.

- **Εκπαίδευση και Δοκιμή Μοντέλων**

  Διαφορετικά μοντέλα μηχανικής μάθησης χρησιμοποιήθηκαν για την πρόβλεψη του διαβήτη όπως οι k-Nearest Neighbors (k-NN), Decision Trees (DT), Random Forest (RF), AdaBoost (AB), Naive Bayes (NB), XGBoost (XB), Multilayer Perceptron (MLP) (Νευρωνικό δίκτυο που μπορεί να μάθει πολύπλοκες μη γραμμικές σχέσεις μέσω εκπαίδευσης σε δεδομένα).

5.5.2 Ενσωμάτωση Μοντέλων (Ensemble Method)

Για τη βελτίωση της ακρίβειας πρόβλεψης, εφαρμόστηκε μια σταθμισμένη μέθοδος συνδυαστικής ταξινόμησης (Weighted Soft Voting Ensemble). Οι αλγόριθμοι συνδυάστηκαν παίρνοντας σαν σημείο αναφοράς την καμπυλη AUC. Ο καλύτερος συνδυασμός μετά από επεξεργασία και μελέτη, ήταν ο AdaBoost και XGBoost (AB + XB), με τιμή AUC = 0.950. Και αυτό γιατί, ως γνωστόν, ο AdaBoost αποδίδει καλά καθώς αυξάνει το βάρος των πιο δύσκολων δειγμάτων, ενώ ο XGBoost, είναι αποδοτικός λόγω της παραλληλοποίησης.

5.5.3 Αποτελέσματα και Συγκριτική Ανάλυση

Τα αποτελέσματα της μελέτης, δηλαδή το προτεινόμενο μοντέλο AB + XB, συγκρίθηκαν με προηγούμενες έρευνες, και φάνηκε ότι το μοντέλο AdaBoost και XGBoost, πέτυχε αύξηση 2% στην AUC (στο 95%), και έτσι υπερβαίνει τις υπάρχουσες τεχνικές [113].

5.5.4 Συμπεράσματα

Η σταθμισμένη συνδυαστική προσέγγιση ταξινόμησης αποδείχθηκε πολύ επιτυχής για την πρόβλεψη του διαβήτη με τη χρήση πολλών μοντέλων (AdaBoost και XGBoost), αυξάνοντας έτσι την ακρίβεια και την αξιοπιστία των προβλέψεων.

Οι τεχνικές προετοιμασίας δεδομένων και η διαδικασία K-fold Cross-Validation βοήθησαν τα μοντέλα να αποδώσουν καλύτερα και να επιδείξουν αξιοπιστία. Η μέθοδος αυτή τονίζει την ικανότητά της να υποστηρίζει κλινικές αποφάσεις με ακριβή και συνεπή δεδομένα, έχοντας έτσι διάφορες εφαρμογές στον τομέα της ιατρικής διάγνωσης. Σχετικά με τα

επερχόμενα έργα, θεωρήθηκε ότι ακόμη καλύτερα αποτελέσματα θα μπορούσαν να προκύψουν από Convolutional Neural Network (CNN).

5.6 Έκτη Μελέτη: “Intelligible Support Vector Machines for Diagnosis of Diabetes Mellitus” by Nahla H. Barakat

Η μελέτη της Nahla H. Barakat και των συνεργατών της, με τίτλο "Intelligible Support Vector Machines for Diagnosis of Diabetes Mellitus", αποσκοπεί στην εφαρμογή της τεχνικής Μηχανές διανυσμάτων υποστήριξης (Support Vector Machines - SVM) για την πρόβλεψη και διάγνωση του διαβήτη τύπου 2. Το ιδιαίτερο χαρακτηριστικό αυτής της μελέτης είναι η προσπάθεια συνδυασμού της υψηλής ακρίβειας των SVM με την κατανοησιμότητα των αποτελεσμάτων, ώστε τα μοντέλα μηχανικής μάθησης να γίνουν πιο διαφανή και κατανοητά για τους χρήστες, ειδικά για ιατρούς και άλλους επαγγελματίες της υγείας [115].

5.6.1 Δεδομένα

Το σύνολο δεδομένων που χρησιμοποιήθηκε στη μελέτη προέρχεται από 4682 περιπτώσεις διαβήτη στο Ομάν. Μετά από χρήση κάποιων παγκόσμιων ιατρικών κριτηρίων και αφαίρεση κάποιων δεδομένων, προέκυψαν τελικά 3014 άτομα ηλικίας άνω των 20 ετών, από τα οποία το 9% είχε διαγνωστεί με διαβήτη τύπου 2. Τα χαρακτηριστικά που αναλύθηκαν περιλαμβάνουν:

- **Ηλικία**
- **Φύλο**
- **Οικογενειακό ιστορικό διαβήτη**
- **Δείκτης Μάζας Σώματος (BMI)**
- **Περίμετρος Μέσης**
- **Αρτηριακή πίεση**
- **Δοκιμή ανοχής γλυκόζης 2 ωρών (OGTT, βασική μέθοδος διάγνωσης διαβήτη)**
- **Επίπεδα χοληστερόλης**
- **Επίπεδα σακχάρου στο αίμα**

Αυτά τα χαρακτηριστικά επιλέχθηκαν καθώς θεωρούνται σημαντικοί παράγοντες κινδύνου για την εμφάνιση του διαβήτη, και τα δεδομένα ήταν αρκετά αντιπροσωπευτικά του πληθυσμού του Ομάν, καλύπτοντας ένα ευρύ φάσμα ηλικιών, φύλων και άλλων δημογραφικών παραμέτρων.

5.6.2 Μεθοδολογία:

Η έρευνα περιλαμβάνει τέσσερα κύρια στάδια, τα οποία ακολουθούνται για την ανάλυση και επεξεργασία των δεδομένων:

1. **Προεπεξεργασία Δεδομένων:** Το πρόβλημα είναι ότι, υπήρχε μεγάλη ανισορροπία μεταξύ διαβητικών (9%) και μη διαβητικών (91%). Για να αντιμετωπιστεί αυτό, εφαρμόστηκαν τεχνικές υποδειγματοληψίας (subsampling) με K-means clustering, έτσι ώστε να επιλεγεί ένα αντιπροσωπευτικό υποσύνολο των μη διαβητικών.

2. **Support Vector Machines (SVM):** Το SVM όπως έχουμε αναφέρει και σε προηγούμενο κεφάλαιο είναι μια ισχυρή τεχνική για την ταξινόμηση δεδομένων σε δύο κατηγορίες. Στην περίπτωση του διαβήτη, το SVM χρησιμοποιείται για να διακρίνει τις περιπτώσεις ασθενών που πάσχουν από διαβήτη από αυτές που δεν έχουν την πάθηση. Οι παράμετροι του μοντέλου ρυθμίζονται μέσω διασταυρωμένης επαλήθευσης, ώστε να διασφαλιστεί η γενίκευση του μοντέλου σε άγνωστα δεδομένα [30]. Στη περίπτωση αυτής της μελέτης, επιλέχθηκε να χρησιμοποιηθεί ο Radial Basis Function (RBF) kernel, ο οποίος μετασχηματίζει τα δεδομένα σε έναν υψηλότερης διάστασης χώρο. Οι παράμετροι που επιλέχθηκαν για τον αλγόριθμο SVM είναι για το γ (gamma) το 0.0005, που η μεταβλητή αυτή ορίζει την καμπυλότητα του διαχωριστικού επιπέδου και για τη μεταβλητή C η τιμή 5, η οποία ορίζει την ισορροπία μεταξύ απόδοσης και απλούστερης εξήγησης.

3. **Διαδικασία Εκπαίδευσης:** Η διαδικασία της εκπαίδευσης περιέχει πολλά μοντέλα SVMs με διαφορετικά κόστη ταξινόμησης. Υπάρχει αξιολόγηση της ακρίβειας και των δεικτών ευαισθησίας. Η εξαγωγή των κανόνων γίνεται από το καλύτερο μοντέλο, έτσι ώστε να γίνει πιο πολύ ερμηνεύσιμο.

5.6.3 Περιγραφή των Μοντέλων και Μεθόδων Εξαγωγής Κανόνων:

Οι συγγραφείς χρησιμοποίησαν δύο μεθόδους εξαγωγής κανόνων από τους SVMs, οι οποίες αναλύονται παρακάτω.

1. **SQRex-SVM (Sequential Covering Rule Extraction):**

   Η μέθοδος εξαγωγής κανόνων από το SVM με διαδοχική κάλυψη επιτρέπει την απόκτηση κανόνων οι οποίοι είναι εύκολα κατανοητοί και εξηγούν τη λειτουργία του μοντέλου.
   Οι κανόνες που εξάγονται είναι ιατρικά χρήσιμοι και κατανοητοί, ενώ διατηρείται υψηλή ακρίβεια στο μοντέλο. Από την άλλη η μέθοδος αυτή απαιτεί αυξημένο υπολογιστικό κόστος και είναι πιο περίπλοκη από το απλό SVM.

   Ακολουθούν οι κανόνες που εξάγονται από το SQRex-SVM με ίσο κόστος λανθασμένης ταξινόμησης:

   - Εάν το FBS > 106.2 mg/dL, τότε είναι διαβητικός
   - Εάν περιφέρεια μέσης ≥ 91 cm και BPDIAS ≥ 90 mmHg, τότε πάλι διαβητικός
   - Ο προεπιλεγμένος κανόνας είναι να μην είναι διαβητικός

2. **Eclectic Rule Extraction (συνδυασμός SVM με δέντρα απόφασης C5):**

   Η μέθοδος αυτή, συνδυάζει διάφορες τεχνικές εξαγωγής κανόνων για να παρέχει ένα πλουσιότερο σύνολο κανόνων, που είναι εύκολα ερμηνεύσιμο από τους χρήστες. Στα πλεονεκτήματα είναι ότι υπάρχει αύξηση της διαφάνειας του μοντέλου, που επιτρέπει στους χρήστες να κατανοήσουν τα κριτήρια που χρησιμοποιούνται για τις προβλέψεις. Από την άλλη, η ακρίβεια του μοντέλου μπορεί να μειωθεί ελαφρώς λόγω της προσθήκης αυτών των τεχνικών [115].

Ακολουθούν οι κανόνες που εξάγονται από την Eclectic Rule:

- Εάν το FBS > 124.2 mg/dL, τότε είναι διαβητικός
- Εάν το FBS > 90 mg/dL και FBS ≤ 124.2 mg/dL και περιφέρεια μέσης ≥ 84 cm και BPDIAS ≥ 90 mmHg, τότε πάλι διαβητικός

5.6.4 Συνολική Αξιολόγηση:

Η μελέτη καταδεικνύει ότι τα μοντέλα SVM είναι εξαιρετικά ακριβή (στη μελέτη επετεύχθει 94% ακρίβεια) και κατάλληλα για εφαρμογές όπου η ακρίβεια είναι ο κύριος στόχος. Τα μοντέλα **SQRex-SVM** και **Eclectic Rule Extraction** προσφέρουν καλύτερη κατανόηση και είναι πιο απλοί [115].

5.7 Έβδομη Μελέτη: "Predicting Diabetes Mellitus With Machine Learning Techniques" *by Quan Zou*

Στην έβδομη μελέτη, θα αναλύσουμε την επιστημονική έρευνα του Quan Zou και των συνεργατών του που επικεντρώνεται στην εφαρμογή τεχνικών μηχανικής μάθησης για την πρόβλεψη του σακχαρώδους διαβήτη. Το κύριο ενδιαφέρον της μελέτης είναι η ανάπτυξη και αξιολόγηση μοντέλων που χρησιμοποιούν δεδομένα από ιατρικές εξετάσεις, προκειμένου να προσδιορίσουν τον κίνδυνο για ανάπτυξη διαβήτη. Ο σκοπός της μελέτης ήταν η αύξηση της ακρίβειας και της αποδοτικότητας των μοντέλων πρόβλεψης, με ιδιαίτερη έμφαση στην ερμήνευση και τη γενίκευση των αποτελεσμάτων [123].

5.7.1 Μεθοδολογία και Δεδομένα

Η μελέτη βασίστηκε σε δύο σύνολα δεδομένων, στα ιατρικά δεδομένα από νοσοκομείο στη Luzhou της Κίνας, που η συγκεκριμένη βάση περιέχει 164.431 εγγραφές υγιών ατόμων και 151.598 εγγραφές διαβητικών. Κάθε εγγραφή, περιλαμβάνει 14 χαρακτηριστικά, τα οποία είναι η ηλικία, ο ρυθμός παλμών, οι αναπνοές ανα λεπτό, η αρτηριακή πίεση (αριστερή και δεξιά), το ύψος, το βάρος, ο δείκτης μάζας σώματος BMI, η γλυκόζη, ο περιφέρεια της μέσης, το LDL και το HDL. Η άλλη βάση δεδομένων είναι το PIMA Indians dataset που έχει αναλυθεί και σε προηγούμενες μελέτες και κεφάλαια.

5.7.2 Μοντέλα Μηχανικής Μάθησης

Χρησιμοποιήθηκαν τρεις αλγόριθμοι μηχανικής μάθησης για την ταξινόμηση των δεδομένων, τα οποία αναφέρονται παρακάτω.

1. **Δέντρα Αποφάσεων (Decision Trees)**

   Ο αλγόριθμος **J48** είναι μια υλοποίηση του αλγορίθμου C4.5 στην εφαρμογή WEKA και κατασκευάζει δέντρα αποφάσεων με τη μέθοδο διαίρεσης και κατάκτησης. Ο συγκεκριμένος αλγόριθμος επιλέγει χαρακτηριστικά με βάση το information gain και δημιουργεί διακλαδώσεις.

2. **Τυχαία Δάση (Random Forests)**

   Η μέθοδος **Random Forest** όπως έχει αναφερθεί συνδυάζει πολλά δέντρα αποφάσεων για να προβλέψει την κατηγορία, μειώνοντας την πιθανότητα υπερ-προσαρμογής.

3. **Νευρωνικά Δίκτυα (Neural Networks)**

   Σε αυτή τη μελέτη, χρησιμοποιήθηκε νευρωνικό δίκτυο δύο επιπέδων με σιγμοειδείς (**sigmoid)** κρυμμένους νευρώνες και νευρώνες εξόδου **softmax**.

5.7.3 Επεξεργασία δεδομένων

Για την ενίσχυση της απόδοσης των μοντέλων, χρησιμοποιήθηκαν δύο τεχνικές επιλογής χαρακτηριστικών, η ανάλυση κύριων συνιστωσών (PCA) και η ελάχιστη πλεονάζουσα μέγιστη σχετικότητα (mRMR). Το PCA χρησιμοποιεί διανύσματα ιδιοτιμών για τη μείωση των διαστάσεων των δεδομένων και βοηθά στη μείωση του θορύβου. Από την άλλη, το mRMR, επιλέγει τα πιο σημαντικά χαρακτηριστικά που σχετίζονται με την έξοδο, μειώνοντας την πλεονάζουσα πληροφορία [123].

5.7.4 Αποτελέσματα

1. **Χρήση όλων των χαρακτηριστικών**

   - **Τυχαία Δάση (Random Forest)**: Το Random Forest είχε την καλύτερη απόδοση με ακρίβεια 80.84% στα δεδομένα του Luzhou. Αυτή η απόδοση αποδίδεται στην ικανότητά του να συνδυάζει πολλαπλά δέντρα και να μειώνει την υπερπροσαρμογή. Η ακρίβεια για το σύνολο δεδομένων ***PIMA Indians*** ήταν ελαφρώς χαμηλότερη, για την ακρίβεια 76.04%, γεγονός που μπορεί να οφείλεται στις διαφορές στα χαρακτηριστικά και την ποιότητα των δεδομένων (Breiman, 2001).
   - **Νευρωνικά Δίκτυα**: Παρουσίασαν καλύτερη απόδοση στο σύνολο δεδομένων *PIMA Indians* με ακρίβεια 76.67% και στα δεδομένα του Luzhou 78.41% .
   - **Δέντρα Αποφάσεων**: Η απόδοση ήταν χαμηλότερη σε σύγκριση με τα άλλα μοντέλα, 78.53% στα δεδομένα του Luzhou και 72.75% στα δεδομένα του PIMA Indians.

2. **Πρόβλεψη μόνο με τη γλυκόζη**

   Στην περίπτωση, που είχαμε μόνο ένα χαρακτηριστικό, συγκεκριμένα τη γλυκόζη, τα αποτελέσματα ήταν όπως παρακάτω. Στα δεδομένα του Luzhou, το Random Forest εξασφάλισε 72.97% ακρίβεια, το δέντρο αποφάσεων J48 ακρίβεια 76.10% και το νευρωνικό δίκτυο 75.72%. Στη βάση δεδομένων των PIMA Indians τα αντίστοιχα αποτελέσματα ήταν 67.28% ακρίβεια στο Random Forest, 68.95% στο δέντρο αποφάσεων και στο νευρωνικό δίκτυο ακρίβεια 71.98%.

3. **Επίδραση της επιλογής χαρακτηριστικών**

Για να μελετηθεί η επίδραση του κάθε χαρακτηριστικού, συγκρίθηκαν τα αποτελέσματα για όλα τα χαρακτηριστικά, για το PCA και για το mRMR. Τα αποτελέσματα για την ακρίβεια στο καθένα, παρουσιάζονται στον παρακάτω πίνακα:

Μέθοδος	Σύνολο Δεδομένων	Ακρίβεια ACC Random Forest	Ακρίβεια ACC J48	Ακρίβεια ACC Neural Network
Όλα τα χαρακτηριστικά	Luzhou	80.84%	78.53%	78.41%
	PIMA Indians	76.04%	72.75%	76.67%
PCA	Luzhou	73.95%	73.88%	74.14%
	PIMA Indians	71.44%	71.67%	74.75%
mRMR	Luzhou	75.08%	76.13%	75.70%
	PIMA Indians	77.21%	75.34%	73.90%

Όπως φαίνεται, η mRMR ήταν πιο αποτελεσματική από την PCA και στις δύο βάσεις δεδομένων.

5.7.5 Συμπεράσματα

Η μελέτη καταδεικνύει ότι οι τεχνικές μηχανικής μάθησης, και ειδικότερα το Random Forest, είναι εξαιρετικά αποτελεσματική για την πρόβλεψη του σακχαρώδους διαβήτη, που έφτασε σε ακρίβεια 80.84%. Η χρήση όλων των χαρακτηριστικών ήταν καλύτερη από την επιλογή μερικών μέσω του PCA. Η mRMR βοήθησε στη μείωση της διάστασης των δεδομένων χωρίς μεγάλη απώλεια της ακρίβειας. Η επιλογή του κατάλληλου μοντέλου εξαρτάται από τις απαιτήσεις της εφαρμογής, την ανάγκη για εξήγηση των αποφάσεων και την υπολογιστική υποδομή [123].

5.8 Όγδοη Μελέτη: "Machine Learning Methods to Predict Diabetes Complications" by Dagliati

Η μελέτη της Dagliati και των συνεργατών της εξετάζει την εφαρμογή μεθόδων μηχανικής μάθησης στην πρόβλεψη επιπλοκών του σακχαρώδους διαβήτη τύπου 2 (T2DM), χρησιμοποιώντας δεδομένα από ηλεκτρονικούς φακέλους υγείας (EHR) σχεδόν 1000 ασθενών αξιοποιώντας ένα data mining pipeline για την κατασκευή και αξιολόγηση προβλεπτικών μοντέλων [133].

5.8.1 Δεδομένα και Προεπεξεργασία

Τα δεδομένα που χρησιμοποιήθηκαν προήλθαν από 943 ασθενείς από το **IRCCS Istituto Clinico Scientifico Maugeri (ICSM) στην Ιταλία**, καλύπτοντας διάστημα **10 ετών.** Περιλάμβαναν κλινικά χαρακτηριστικά, όπως ηλικία, φύλο, δείκτης μάζας σώματος (BMI), γλυκοζυλιωμένη αιμοσφαιρίνη (HbA1c), αρτηριακή υπέρταση, κάπνισμα καθώς και άλλες εργαστηριακές και κλινικές μετρήσεις. Εξαιρέθηκαν ασθενείς που είχαν ήδη διαγνωσθεί με επιπλοκές κατά την πρώτη επίσκεψη.

Τα στάδια του Data Mining Pipeline περιελάμβανε τέσσερα στάδια,

1. **Center Profiling**

   Δηλαδή ανάλυση δεδομένων για την κατανόηση των χαρακτηριστικών των ασθενών και των μοτίβων περίθαλψης.

2. **Predictive Model Targeting**

   Επιλογή στρατηγικών μοντελοποίησης βάσει της βιβλιογραφίας και των δεδομένων του ICSM.

3. **Predictive Model Construction**

Επεξεργασία ελλιπών δεδομένων και αντιμετώπιση των ανισορροπιών των κατηγοριών.

4. **Predictive Model Validation**

   Αξιολόγηση απόδοσης των μοντέλων με μετρικές όπως sensitivity, specificity, accuracy και καμπύλες ROC (AUC).

Στις τεχνικές επεξεργασίας δεδομένων, χρησιμοποιήθηκαν η διαχείριση ελλιπών δεδομένων (υπήρχαν ελλείψεις σε κάποια λιπιδικά δεδομένα, όπως συνολική χοληστερόλη και τριγλυκερίδια). και η μέθοδος **missForest** (αλγόριθμος Random Forest), η οποία βοήθησε στην αναπλήρωση των ελλιπών τιμών στα δεδομένα. Επίσης, στην επεξεργασία χρησιμοποιήθηκε τεχνική αντιμετώπισης ανισορροπίας κατηγοριών, δηλαδή εφαρμόστηκε επαναδειγματοληψία (oversampling) της μειοψηφούσας τάξης για να βελτιωθεί η ταξινόμηση [133].

5.8.2 Μοντέλα Μηχανικής Μάθησης

Η μελέτη αξιολόγησε την απόδοση διαφόρων μοντέλων μηχανικής μάθησης:

1. **Logistic Regression (LR)**
2. **Naïve Bayes (NB)**
3. **Support Vector Machines (SVMs)**
4. **Random Forest (RF)**

Για κάθε αλγόριθμο, αξιολογήθηκαν τρεις χρονικοί ορίζοντες πρόβλεψης, τα 3, 5 και 7 έτη αντίστοιχα από την πρώτη επίσκεψη σε νοσοκομείο. Η λογιστική παλινδρόμηση αποδείχθηκε η πιο αξιόπιστη και απέδωσε καλύτερα.

5.8.3 Ανάλυση σημαντικών παραγόντων κινδύνου

Οι σημαντικότεροι παράγοντες κινδύνου, μετά από την μελέτη των δεδομένων με τη χρήση των συγκεκριμένων αλγορίθμων είναι η HbA1c (γλυκοζυλιωμένη αιμοσφαιρίνη), η αύξησή της οδηγεί σε επιπλοκές, η διάρκεια του διαβήτη, ο δείκτης μάζας σώματος BMI και η υπέρταση.

5.8.5 Συμπεράσματα

Η μελέτη αποδεικνύει ότι οι μέθοδοι μηχανικής μάθησης είναι αποτελεσματικές στην πρόβλεψη των επιπλοκών του σακχαρώδους διαβήτη τύπου 2. Τα μοντέλα θα μπορούσαν να ενσωματωθούν σε συστήματα υποστήριξης αποφάσεων για πρώιμη παρέμβαση σε ασθενείς υψηλού κινδύνου [133].

5.9 Ένατη Μελέτη: "Analysis of diabetes mellitus for early prediction using optimal feature selection" by N. Sneha & Tarun Gangil

Η μελέτη αυτή, εκπονημένη από τους N. Sneha και Tarun Gangil, εστιάζει στην πρόβλεψη του διαβήτη μέσω μηχανικής μάθησης και της επιλογής των σημαντικότερων χαρακτηριστικών από τα δεδομένα [128]. Η έρευνα τονίζει τη σημασία της πρώιμης διάγνωσης και ανίχνευσης του διαβήτη, δεδομένων των σοβαρών επιπλοκών που προκαλεί, όπως καρδιαγγειακές παθήσεις, νεφρική ανεπάρκεια, απώλεια όρασης. Επομένως, η ανάγκη για αξιόπιστες μεθόδους πρόβλεψης του διαβήτη είναι πιο επιτακτική από ποτέ.

5.9.1 Δεδομένα και Προεπεξεργασία

Τα δεδομένα που χρησιμοποιήθηκαν προέρχονται από το σύνολο δεδομένων UCI machine repository με 2500 εγγραφές και 15 χαρακτηριστικά, Η προεπεξεργασία των δεδομένων περιλάμβανε τον καθαρισμό των δεδομένων και την κανονικοποίηση τους για τη βελτίωση των αποτελεσμάτων των μοντέλων. Για τη δοκιμή των δεδομένων με τις τεχνικές ταξινόμησης, εξετάσαμε 768 στοιχεία δεδομένων.

5.9.2 Μοντέλα Μηχανικής Μάθησης

Εφαρμόστηκαν 5 γνωστοί αλγόριθμοι μηχανικής μάθησης που είναι οι Support Vector Machine, Random Forest, Naive Bayes, Decision Tree και K-Nearest Neighbors. Τα αποτελέσματα που προέκυψαν για την ακρίβεια είναι 77.73% για το Support Vector Machine, 75.39% για το Random Forest, 73.48% για το Naive Bayes, 73.18% για το Decision Tree και 63.04% για το K-Nearest Neighbors [128].

5.9.3 Μέθοδοι Επιλογής Χαρακτηριστικών

Η επιλογή των κατάλληλων χαρακτηριστικών για την πρόβλεψη είναι κρίσιμη για την απόδοση των μοντέλων. Οι ερευνητές απέκλεισαν 4 χαρακτηριστικά που δεν επηρέασαν σημαντικά την ταξινόμηση και κράτησαν τα εξής χαρακτηριστικά (ηλικία, φύλο, πίεση αίματος, BMI, επίπεδα γλυκόζης, επίπεδα ινσουλίνης, δείκτης Diabetes Pedigree Function, ορός Sodium & Potassium και πάχος δέρματος)

5.9.4 Μοντέλα Μηχανικής Μάθησης

Μετά την βελτιστοποίηση και την αφαίρεση κάποιων χαρακτηριστικών, η ακρίβεια στο Naive Bayes αυξήθηκε από 73.48% σε 82.30% και του SVM σε 77%.

5.9.4 Συμπεράσματα

Η επιλογή των σωστών χαρακτηριστικών βελτιώνει σημαντικά την ακρίβεια της πρόβλεψης. Ο αλγόριθμος Naive Bayes πέτυχε την μεγαλύτερη αλλαγή και αύξηση μετά την βελτιστοποίηση των χαρακτηριστικών [128].

5.10 Δέκατη Μελέτη: "Machine-Learning-Based Disease Diagnosis: A Comprehensive Review" by Manjurul Ahsan

Μια εμπεριστατωμένη αξιολόγηση της εφαρμογής της μηχανικής μάθησης (ML) και της βαθιάς μάθησης (DL) στην ανίχνευση ασθενειών δίνεται από την εργασία του Md Manjurul Ahsan και των συνεργατών του [130]. Εκτός από τις διάφορες ασθένειες που διερευνώνται, γίνεται αναφορά και στον σακχαρώδη διαβήτη και στις μελέτες που πραγματοποιήθηκαν, αξιοποιώντας έτσι και άλλες πηγές. Η έγκαιρη διάγνωση του διαβήτη υποβοηθείται από μοντέλα μηχανικής μάθησης, βοηθούν επίσης στη βελτιστοποίηση των διαγνωστικών οργάνων και στην πρόβλεψη των συνεπειών, συμπεριλαμβανομένης της διαβητικής αμφιβληστροειδοπάθειας.

5.10.1 Ανάλυση Αλγορίθμων και Δεδομένων

Στο έγγραφο αναφέρονται αρκετές μελέτες που χρησιμοποίησαν διαφορετικούς αλγορίθμους μηχανικής μάθησης (όπως το Decision Tree, Random Forest, Naive Bayes Artificial Neural Network, Support Vector Machine κτλ.) και σε διαφορετικές βάσεις δεδομένων (όπως PIMA Indians, DIABIM-MUNE και άλλες ιδιόκτητες βάσεις δεδομένων). Κάποια από τα οποία έβγαλαν τα παρακάτω ευρήματα:

- Ο Random Forest, συχνά υπερτερεί των άλλων αλγορίθμων στις περισσότερες εφαρμογές μηχανικής μάθησης.
- Οι Deep Learning προσφέρουν υψηλή ακρίβεια, αλλά απαιτούν μεγάλες βάσεις δεδομένων και περισσότερο υπολογιστικό χρόνο.
- Η βάση δεδομένων των PIMA Indians χρησιμοποιείται εκτενώς.
- Η μελέτη **Alhassan** ανέπτυξε ένα **νέο μεγάλο dataset King Abdullah International Research Center for Diabetes - KAIMRCD,** με 14.000 δείγματα**)**, το οποίο είναι ένα από τα μεγαλύτερα για διαβήτη. Κατά τη διάρκεια αυτού του πειράματος, ο Alhassan και συνεργάτες του, παρουσίασαν μια αρχιτεκτονική CDSS βασισμένη σε LSTM και βαθιά νευρωνικά δίκτυα που πέτυχε 97% ακρίβεια [130].

5.10.2 Σημαντικά Θέματα και Προκλήσεις

Στη μελέτη, παρουσιάστηκαν κάποιες προκλήσεις αλλά και κάποια ζητήματα που υπάρχουν ακόμα στη διάγνωση του διαβήτη μέσα της μηχανικής μάθησης. Κάποια από τα οποία είναι:

- Ανισορροπία δεδομένων: Σε σύγκριση με τους ασθενείς με διαβήτη, οι περισσότερες βάσεις δεδομένων διαθέτουν σημαντικά περισσότερα άτομα χωρίς διαβήτη.
- Ερμηνευσιμότητα Μοντέλων: Οι αλγόριθμοι πρέπει να είναι κατανοητοί στους γιατρούς και τους ασθενείς, καθώς τις περισσότερες φορές λειτουργούν ως «μαύρο κουτί» και είναι κάπως δύσκολο να εξηγήσει κανείς γιατί ένα άτομο διαγνώστηκε με διαβήτη.

- Διαχείριση Μεγάλων Δεδομένων: Ιδιαίτερα στην εποχή όπου λαμβάνουμε γονιδιωματικές πληροφορίες από διάφορους αισθητήρες κάθε στιγμή (κινητά τηλέφωνα, wearables), η οργάνωση και η διασφάλιση των προσωπικών δεδομένων αποτελεί μεγάλη δυσκολία.

5.10.3 Συμπεράσματα και Μελλοντικές Τάσεις

Η έρευνα καταλήγει στο συμπέρασμα ότι η βαθιά μάθηση και η μηχανική μάθηση έχουν μεγάλη δύναμη να αλλάξουν τον τομέα της ιατρικής διάγνωσης γενικά. Στη διάγνωση του διαβήτη, η μηχανική μάθηση αποδίδει μάλλον καλά. Η μείωση των προκαταλήψεων στα μοντέλα μηχανικής μάθησης εξαρτάται από ισορροπημένα και ποικίλα δεδομένα [130].

6. Προσομοίωση Διάγνωσης Σακχαρώδη Διαβήτη με τη χρήση προγραμματισμού. Ανάλυση Αποτελεσμάτων και Συμπεράσματα

Σε αυτό το κεφάλαιο, θα αναλυθεί η προσομοίωση που υλοποιήθηκε κάνοντας χρήση της γλώσσας προγραμματισμού Python, με τη βοήθεια κατάλληλων βιβλιοθηκών μηχανικής μάθησης (sklearn, panda κτλ). Η προσομοίωση έγινε χρησιμοποιώντας την πλατφόρμα του Colab, το οποίο είναι μια φιλοξενούμενη υπηρεσία σημειώσεων Jupyter Notebook που δεν απαιτεί καμία εγκατάσταση για να χρησιμοποιηθεί και παρέχει δωρεάν πρόσβαση σε υπολογιστικούς πόρους, συμπεριλαμβανομένων των GPU και TPU. Το Colab είναι ιδιαίτερα κατάλληλο για τη μηχανική μάθηση, την επιστήμη των δεδομένων και την εκπαίδευση. Επίσης για τη βάση δεδομένων, έγινε χρήση της βάσης των PIMA Indians, το οποίο έχει αναλυθεί λεπτομερώς.

6.1 Λεπτομερής ανάλυση του κώδικα

6.1.1. Εισαγωγή των βιβλιοθηκών

```
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import accuracy_score, classification_report,
confusion_matrix
```

Θεωρητική τεκμηρίωση:

- **pandas**: Είναι μια βιβλιοθήκη της Python για τη φόρτωση, επεξεργασία και ανάλυση δεδομένων με τη μορφή πινάκων-πλαισίων δεδομένων. Πρόκειται για ένα ισχυρό εργαλείο διαχείρισης δεδομένων με μεγάλες δυνατότητες φιλτραρίσματος, συνδυασμού, συγχώνευσης και τροποποίησης καθώς και χειρισμού τεράστιου όγκου δεδομένων.
- **scikit-learn (sklearn)**: Μεταξύ των πιο συχνά χρησιμοποιούμενων βιβλιοθηκών για μηχανική μάθηση, η scikit-learn (sklearn) προσφέρει εργαλεία για την προετοιμασία δεδομένων, την εκπαίδευση αλγορίθμων και την αξιολόγηση της απόδοσης των μοντέλων.
    - **train_test_split**: Σχεδιασμένο για το διαχωρισμό των δεδομένων σε δύο σύνολα, εκπαίδευση και δοκιμή, το train_test_split βοηθά στην αποφυγή υπερβολικής προσαρμογής των μοντέλων.
    - **StandardScaler**: Σημαντικό για αλγορίθμους όπως η λογιστική παλινδρόμηση και το SVM, οι οποίοι επηρεάζονται από τη διαφορετική

κλίμακα των χαρακτηριστικών, το StandardScaler βοηθά στην κανονικοποίηση των χαρακτηριστικών έτσι ώστε όλα να έχουν την ίδια κλίμακα.

- **LogisticRegression**: Μια βασική μέθοδος για τη δυαδική ταξινόμηση, η λογιστική παλινδρόμηση προβλέπει την πιθανότητα ενός δείγματος να ανήκει σε μία από τις δύο κατηγορίες.
- **accuracy_score, classification_report, confusion_matrix**: Η απόδοση των μοντέλων αξιολογείται με τη χρήση των accuracy_score, classification_report, confusion_matrix. Η ακρίβεια, η αναλυτική έκθεση και ο πίνακας σύγχυσης, λαμβανόμενα μαζί, προσφέρουν μια συνολική εικόνα της ποιότητας του μοντέλου.

6.1.2. Φόρτωση και αρχική επεξεργασία του dataset

```
# Load the dataset
url =
"https://raw.githubusercontent.com/jbrownlee/Datasets/master/pima-
indians-diabetes.data.csv"
column_names = ['Pregnancies', 'Glucose', 'BloodPressure',
'SkinThickness', 'Insulin', 'BMI', 'DiabetesPedigreeFunction', 'Age',
'Outcome']
```

Θεωρητική τεκμηρίωση:

- **Δεδομένα**: Το dataset περιλαμβάνει δεδομένα από τη μελέτη του διαβήτη των PIMA Indians και χρησιμοποιείται συνήθως για την ανάλυση προβλέψεων σχετικά με την ύπαρξη διαβήτη σε ένα άτομο.

  Οι μεταβλητές-παράμετροι (χαρακτηριστικά) περιλαμβάνουν δημογραφικές και ιατρικές πληροφορίες (όπως ο αριθμός εγκυμοσυνών, τα επίπεδα γλυκόζης, την αρτηριακή πίεση και άλλες μετρήσεις υγείας που σχετίζονται με τον διαβήτη, οι οποίες βρίσκονται στις στήλες στα δεδομένα).

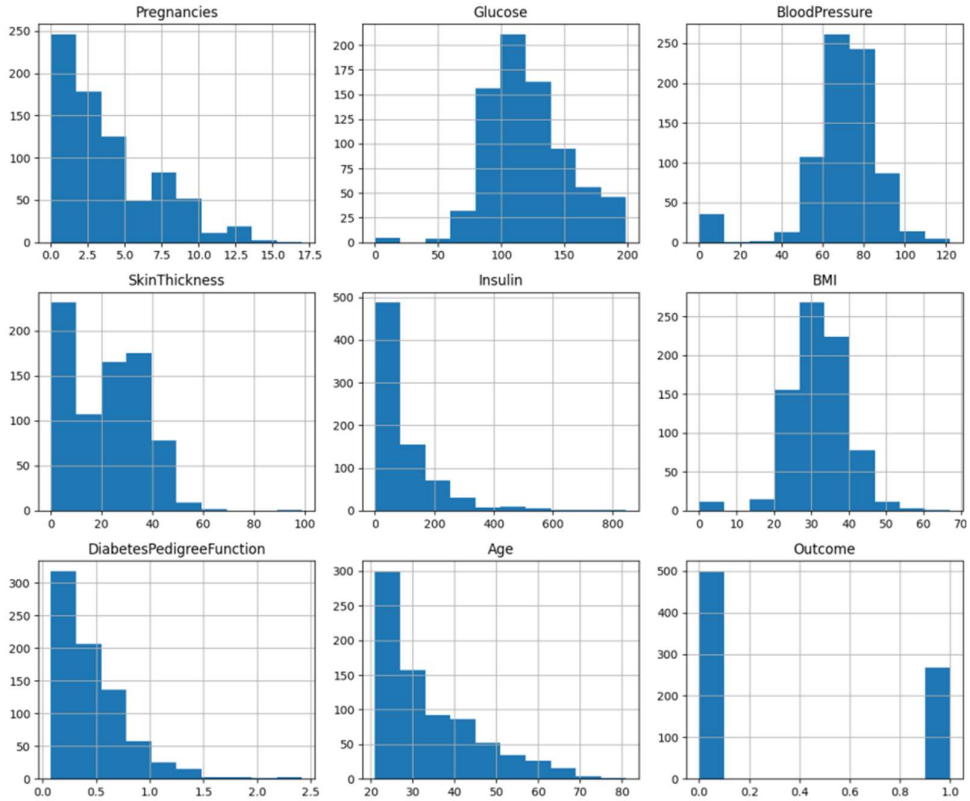
- **Ανάγνωση δεδομένων**: Το pd.read_csv() χρησιμοποιείται για να φορτώσει δεδομένα από ένα εξωτερικό αρχείο CSV και να τα μετατρέψει σε pandas dataframe, όπου τα δεδομένα μπορούν να επεξεργαστούν και να αναλυθούν εύκολα.

6.1.3. Εξερεύνηση των δεδομένων

```
# Read the data into a pandas dataframe
data = pd.read_csv(url, header=None, names=column_names)

import seaborn as sns
import matplotlib.pyplot as plt

# Distribution of each feature
data.hist(bins=10, figsize=(12, 10))
plt.tight layout()
plt.show()

# Correlation matrix
plt.figure(figsize=(10, 8))
sns.heatmap(data.corr(), annot=True, cmap='coolwarm')
plt.show()

# Pairplot to see the distribution of features and their relationships
sns.pairplot(data, hue='Outcome')
plt.show()
```

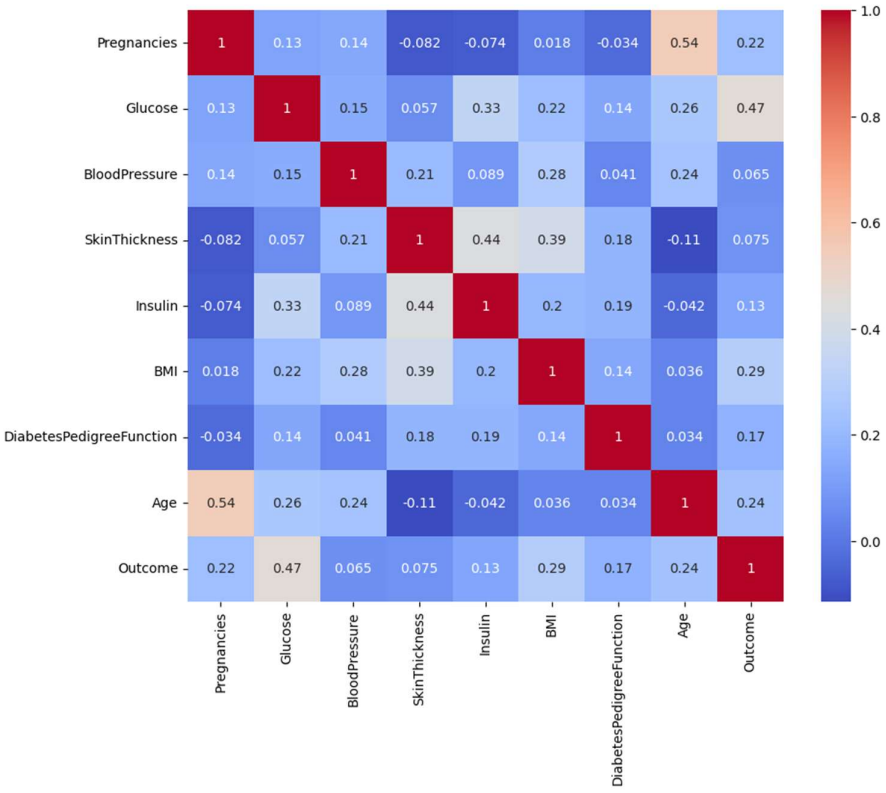
Θεωρητική τεκμηρίωση:

- **Οπτικοποίηση**: Η χρήση της βιβλιοθήκης seaborn και matplotlib επιτρέπει την εύκολη παρουσίαση των δεδομένων.
    - **Histograms (Ιστογράμματα)**: Χρησιμοποιούνται για την απεικόνιση της κατανομής κάθε χαρακτηριστικού. Τα ιστογράμματα βοηθούν στην κατανόηση της διανομής των τιμών (π.χ. ποιο χαρακτηριστικό έχει μια πιο φυσιολογική κατανομή ή ποιο έχει ασυμμετρία). Παρακάτω, δίνεται το αποτέλεσμα και η απεικόνιση των ιστογραμμάτων που προκύπτουν μετά την εκτέλεση της συγκεκριμένης εντολής. Φαίνονται αναλυτικά για κάθε χαρακτηριστικό της βάσης δεδομένων οι κατανομές.



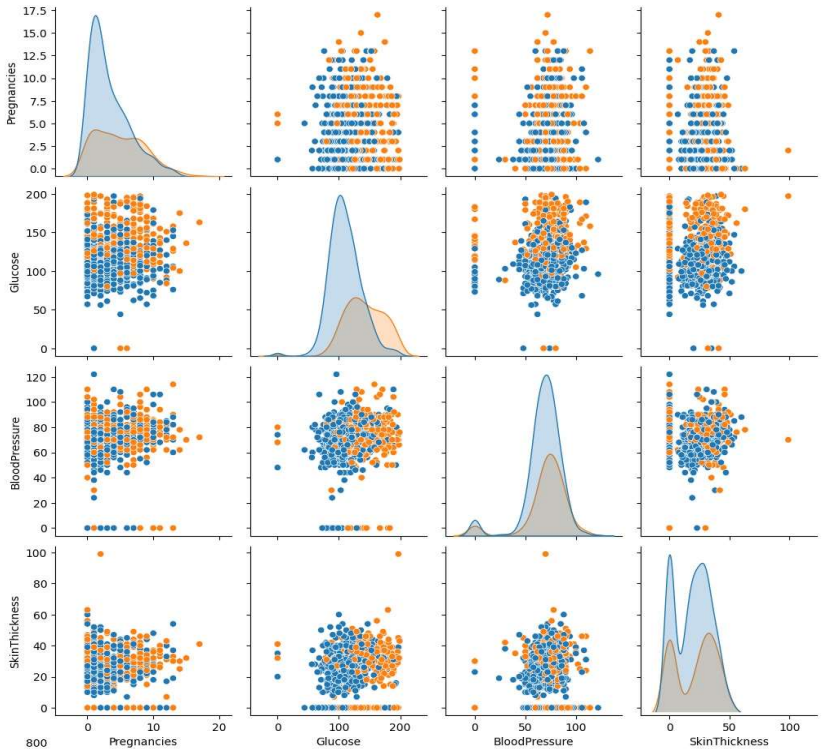
Γράφημα 1: Ιστογράμματα με τις κατανομές των 9 χαρακτηριστικών στη βάση δεδομένων των PIMA Indians.

- **Correlation Matrix (Πίνακας Συσχέτισης)**: Δείχνει τις σχέσεις μεταξύ των χαρακτηριστικών και την πιθανή επιρροή ενός χαρακτηριστικού σε άλλο. Ο πίνακας συσχέτισης απεικονίζει το πόσο έντονη είναι η γραμμική συσχέτιση μεταξύ δύο μεταβλητών, με τιμές που κυμαίνονται από -1 έως 1. Μια τιμή κοντά στο 1 δείχνει ισχυρή θετική συσχέτιση (χρωματικά πιο κοντά στο κόκκινο), ενώ μια τιμή κοντά στο -1 δείχνει ισχυρή αρνητική συσχέτιση (χρωματικά πιο κοντά στο μπλε. Παρακάτω, δίνεται ο πίνακας συσχέτισης που προέκυψε για τη βάση δεδομένων που χρησιμοποιούμε.

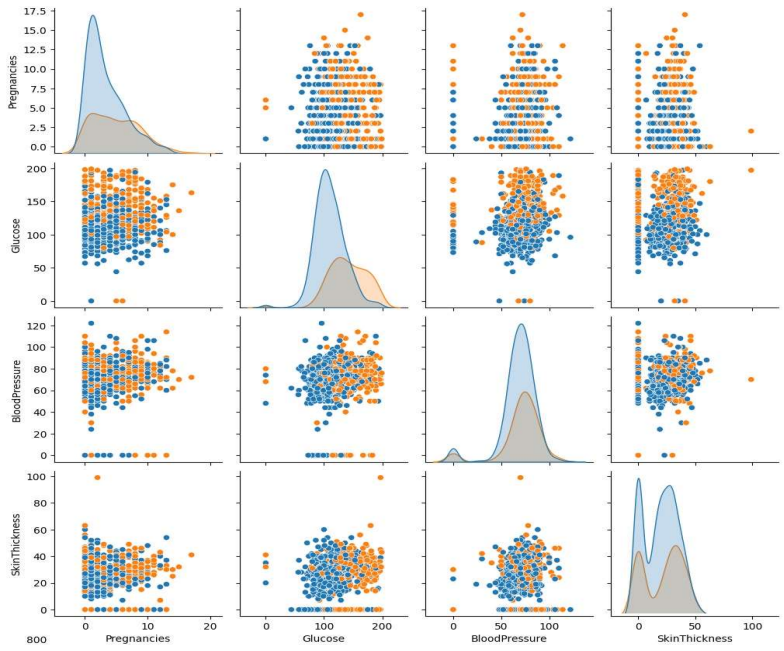


Γράφημα 2: Πίνακας συσχέτισης με τις σχέσεις μεταξύ των χαρακτηριστικών για τη βάση δεδομένων των PIMA Indians.

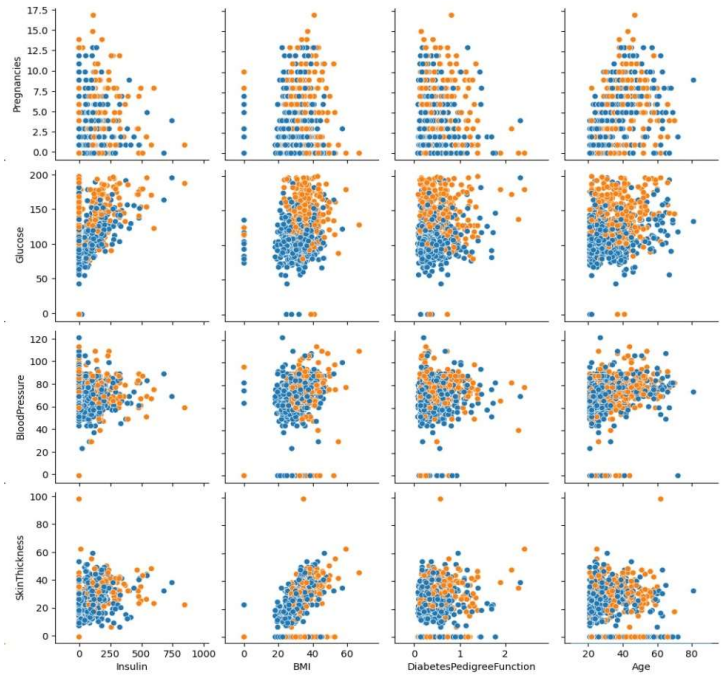
- **Pairplot (Διάγραμμα Ζευγών)**: Χρησιμοποιείται για να απεικονίσει την κατανομή και τις σχέσεις μεταξύ δύο χαρακτηριστικών. Χρησιμοποιώντας το χρώμα της ετικέτας "Outcome", μπορούμε να δούμε πώς κατανέμονται τα δεδομένα ανάλογα με το αν ένα άτομο έχει ή δεν έχει διαβήτη. Παρακάτω, δίνονται τα διαγράμματα που απεικονίζουν τις κατανομές και τις σχέσεις μεταξύ των χαρακτηριστικών που υπάρχουν στη βάση. Με τα 2 διαφορετικά χρώματα, απεικονίζονται το Outcome, που βλέπουμε αν ένα άτομο είναι διαβητικό ή όχι και πώς αυτό κατανέμεται.



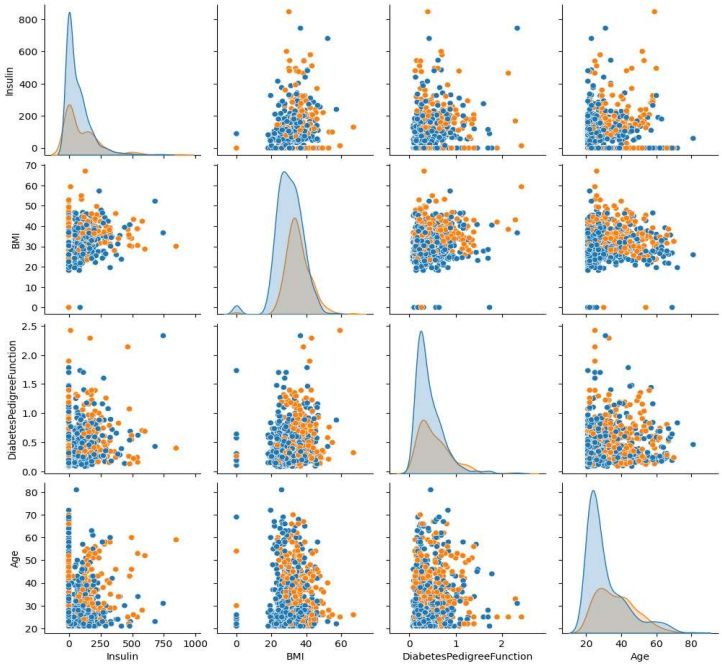
Γράφημα 3: Διαγράμματα ζευγών που απεικονίζουν τις κατανομές και τις σχέσεις μεταξύ των χαρακτηριστικών που υπάρχουν στη βάση δεδομένων των PIMA Indians.



Γράφημα 4: Διαγράμματα ζευγών που απεικονίζουν τις κατανομές και τις σχέσεις μεταξύ των χαρακτηριστικών που υπάρχουν στη βάση δεδομένων των PIMA Indians.



Γράφημα 5: Διαγράμματα ζευγών που απεικονίζουν τις κατανομές και τις σχέσεις μεταξύ των χαρακτηριστικών που υπάρχουν στη βάση δεδομένων των PIMA Indians.



Γράφημα 6: Διαγράμματα ζευγών που απεικονίζουν τις κατανομές και τις σχέσεις μεταξύ των χαρακτηριστικών που υπάρχουν στη βάση δεδομένων των PIMA Indians.

6.1.4 Προεπεξεργασία δεδομένων (Handling Missing Values)

```
# Replace zeros with NaN for specific columns
columns_to_replace = ['Glucose', 'BloodPressure', 'SkinThickness',
'Insulin', 'BMI']
data[columns_to_replace] = data[columns_to_replace].replace(0, pd.NA)

# Fill NaN with the median of each column
data[columns_to_replace] = data[columns_to_replace].apply(lambda col:
col.fillna(col.median()))

# Check again for any missing values
print(data.isnull().sum())
```

Θεωρητική τεκμηρίωση:

- **Μηδενικές τιμές (Zero Values)**: Πολλά ιατρικά δεδομένα, διαθέτουν μηδενικές τιμές για ορισμένα χαρακτηριστικά (όπως η γλυκόζη ή η αρτηριακή πίεση), πράγμα που δεν είναι λογικό. Θεωρούνται ως ελλείποντα δεδομένα, οι μηδενικές τιμές αντικαθίστανται έτσι με την τιμή NaN για πρόσθετη επεξεργασία.
- **Αντικατάσταση των NaN**: Η αντικατάσταση NaN είναι μια τυπική προσέγγιση αντί της διαγραφής ολόκληρων γραμμών με ελλείποντα δεδομένα: η διάμεση τιμή κάθε χαρακτηριστικού αντικαθιστά τα κενά. Σε αντίθεση με τη μέση τιμή, η διάμεσος είναι ένα σταθερό στατιστικό στοιχείο απαλλαγμένο από διακυμάνσεις που εξαρτώνται από ακραίες τιμές.
- **Επιβεβαίωση**: Ψάχνουμε για τυχόν τελευταία ελλείποντα δεδομένα για να βεβαιωθούμε ότι όλα έχουν επιλυθεί.

6.1.5 Δημιουργία νέου χαρακτηριστικού

```
# Example of creating a new feature: BMI and Age interaction
data['BMI_Age_Interaction'] = data['BMI'] * data['Age']
```

Θεωρητική τεκμηρίωση:

- **Αλληλεπίδραση Χαρακτηριστικών (Feature Interaction)**: Η δημιουργία νέων χαρακτηριστικών μπορεί να βοηθήσει ένα μοντέλο μηχανικής μάθησης να αποδώσει καλύτερα. Σε αυτή την περίπτωση, σχεδιάζουμε τη συνάρτηση BMI_Age_Interaction για να δείξουμε πώς αλληλεπιδρούν ο Δείκτης Μάζας Σώματος (BMI) και η ηλικία.
- **Σημασία Αλληλεπιδράσεων**: Η ανάγκη αλληλεπιδράσεων δύο μεταβλητών που λαμβάνονται μαζί μπορεί να παρέχουν πρόσθετες πληροφορίες που δεν είναι σαφείς από τη μεμονωμένη μελέτη των μεταβλητών. Όσον αφορά τον διαβήτη, η σχέση μεταξύ της ηλικίας και του BMI μπορεί να είναι πραγματικά σημαντική όσον αφορά την πρόβλεψη της ασθένειας.

6.1.6 Κανονικοποίηση των δεδομένων (Standardization)

```
# Standardize all features (after creating new ones)
X = data.drop('Outcome', axis=1)
y = data['Outcome']

scaler = StandardScaler()
X = scaler.fit_transform(X)
```

Θεωρητική Τεκμηρίωση:

- **Χαρακτηριστικά και Στόχος (Features and Target)**:
    - Ενώ το y περιλαμβάνει τις ετικέτες ή τις κλάσεις που θέλουμε να προβλέψουμε, στην προκειμένη περίπτωση αν ένα άτομο έχει διαβήτη ή όχι, το x έχει όλα τα χαρακτηριστικά, δηλαδή τις στήλες του συνόλου δεδομένων εκτός από το Outcome.
- **Κανονικοποίηση (Standardization)**:
    - Η κανονικοποίηση είναι ένα σημαντικό βήμα προεπεξεργασίας δεδομένων όταν τα χαρακτηριστικά έχουν διαφορετικές κλίμακες. Η συνάρτηση **StandardScaler** κανονικοποιεί τα δεδομένα μετατρέποντάς τα έτσι ώστε να έχουν μέσο όρο 0 και τυπική απόκλιση 1.
    - Η κανονικοποίηση είναι απαραίτητη για πολλούς αλγόριθμους μηχανικής μάθησης που βασίζονται σε αποστάσεις (όπως SVM, K-Nearest Neighbors) ή γραμμικούς αλγόριθμους (όπως η λογιστική παλινδρόμηση), οι οποίοι μπορούν να επηρεαστούν από διακυμάνσεις στις κλίμακες των χαρακτηριστικών.

6.1.7 Διαχωρισμός των δεδομένων σε Training και Test Set

```
from sklearn.model selection import GridSearchCV
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.metrics import accuracy_score, classification_report,
confusion matrix, roc curve, auc
from sklearn.model_selection import train_test_split
from sklearn.model selection import cross val score
from sklearn.ensemble import RandomForestClassifier
from sklearn.svm import SVC
from sklearn.neighbors import KNeighborsClassifier
from sklearn.ensemble import GradientBoostingClassifier
from sklearn.naive bayes import GaussianNB
from sklearn.tree import DecisionTreeClassifier
from sklearn.ensemble import AdaBoostClassifier
from xgboost import XGBClassifier

# Assuming X and y are your features and target variable

# Split data into training and test sets
X train, X test, y train, y test = train_test_split(X, y,
test_size=0.2, random_state=42)
```

Θεωρητική Τεκμηρίωση:

- **Train-Test Split**:
    - Ο διαχωρισμός των δεδομένων σε **training set** και **test set** είναι μια συνήθης διαδικασία στη μηχανική μάθηση. Το **training set** χρησιμοποιείται για την εκπαίδευση του μοντέλου, ενώ το **test set** χρησιμοποιείται για την αξιολόγηση της απόδοσής του σε νέα, αθέατα δεδομένα.
    - Το test_size=0.2 σημαίνει ότι το 20% των δεδομένων χρησιμοποιείται για το test set, ενώ το 80% χρησιμοποιείται για την εκπαίδευση.
    - Το random_state=42 εγγυάται την επαναληψιμότητα του κώδικα, δηλαδή τον ίδιο διαχωρισμό κάθε φορά που εκτελείται ο κώδικας.

6.1.8 Αρχικοποίηση και εκπαίδευση μοντέλων (Model Training)

```
# Initialize models
models = {
  "Logistic Regression": LogisticRegression(),
  "Random Forest": RandomForestClassifier(),
  "Support Vector Machine": SVC(probability=True),  # Enable
probability for ROC curve
  "K-Nearest Neighbors": KNeighborsClassifier(),
  "Gradient Boosting": GradientBoostingClassifier(),
  "Gaussian Naive Bayes": GaussianNB(),
```

```
  "Decision Tree Classifier": DecisionTreeClassifier(),
  "AdaBoost Classifier": AdaBoostClassifier(),
   # "XGB Classifier": XGBClassifier(use_label_encoder=False,
eval_metric='logloss')
}
```

Θεωρητική Τεκμηρίωση:

- **Πολλαπλά Μοντέλα**: Εδώ αρχικοποιούνται πολλοί διαφορετικοί αλγόριθμοι μηχανικής μάθησης για σύγκριση. Κάθε αλγόριθμος έχει διαφορετικό τρόπο λειτουργίας:
    - **Logistic Regression**: Απλή μέθοδος δυαδικής ταξινόμησης που βασίζεται στη γραμμική παλινδρόμηση για την πρόβλεψη της εμφάνισης μιας κλάσης.
    - **Random Forest**: Μια μέθοδος ensemble γνωστή ως Random Forest δημιουργεί διάφορα δέντρα απόφασης και υπολογίζει μια μέση πρόβλεψη από όλα τα δέντρα.
    - **Support Vector Machine (SVM)**: Είναι μια μέθοδος ταξινόμησης που στοχεύει στο βέλτιστο διαχωρισμό υπερκλάσεων μεταξύ των κλάσεων.
    - **K-Nearest Neighbors (KNN)**: Η απλή μέθοδος με βάση την απόσταση K-Nearest Neighbors (KNN) ταξινομεί ένα νέο σημείο δεδομένων με βάση τους πλησιέστερους γείτονές τους.
    - **Gradient Boosting**: Ο συνδυασμός πολλών αδύναμων μοντέλων για να προκύψει ένα ισχυρό μοντέλο είναι ο αλγόριθμος γνωστός ως gradient boosting.

- **Gaussian Naive Bayes**: Με βάση το θεώρημα του Bayes και υπογραμμίζοντας την υπόθεση της ανεξαρτησίας των χαρακτηριστικών, η μέθοδος Gaussian Naive Bayes είναι μια μέθοδος ταξινόμησης.
- **Decision Tree Classifier**: Δημιουργεί ένα δέντρο αποφάσεων όπου κάθε κόμβος αντιπροσωπεύει ένα ερώτημα χαρακτηριστικού και οι κλάδοι δείχνουν επιλογές.
- **AdaBoost**: Δίνοντας έμφαση στις εσφαλμένα ταξινομημένες περιπτώσεις, το AdaBoost είναι μια μέθοδος συνόλου που αποσκοπεί στη βελτίωση των ασθενέστερων μοντέλων.

6.1.9 Grid Search για Βελτιστοποίηση Υπερπαραμέτρων

```
# Parameter grids
param grids = {
    "Logistic Regression": {
        'C': [0.1],
        'penalty': ['l1'],
        'solver': ['saga'],
        'max_iter': [100]
    },
    "Random Forest": {
        'n_estimators': [100],
        'max depth': [10],
        'min_samples_split': [10],
        'min samples leaf': [4],
        'bootstrap': [True]
    },
    "Support Vector Machine": {
        'C': [0.1],
        'kernel': ['linear'],
        'gamma': ['scale'],
        'degree': [3],  # Used only for 'poly' kernel
        'coef0': [0.0]  # Used only for 'poly' and 'sigmoid'
    },
    "K-Nearest Neighbors": {
        'n neighbors': [7],
        'weights': ['uniform'],
        'algorithm': ['auto'],
        'p': [1]  # 1 for Manhattan distance, 2 for Euclidean distance
    },
    "Gradient Boosting": {
        'n estimators': [200],
        'learning_rate': [0.01],
        'max depth': [3],
        'subsample': [0.8],
        'min samples split': [10],
        'min_samples_leaf': [4]
    },
    "Gaussian Naive Bayes": {
        'var_smoothing': [1e-9]
    },
    "Decision Tree Classifier": {
        'criterion': ['log loss'],
        'splitter': ['random'],
        'max_depth': [10],
```

```
        'min samples split': [5],
        'min_samples_leaf': [4],
        'max_features': ['log2']
    },
    "AdaBoost Classifier": {
        'n estimators': [100],
        'learning_rate': [0.1],
        'algorithm': ['SAMME.R']
    }

}
```

```
        'min samples split': [5],
        'min_samples_leaf': [4],
        'max_features': ['log2']
    },
    "AdaBoost Classifier": {
        'n estimators': [100],
        'learning_rate': [0.1],
        'algorithm': ['SAMME.R']
    }

}

best_estimators = {}

# Perform GridSearchCV for each model
for model name, model in models.items():
    print(f"Performing GridSearchCV for {model name}...")
    grid_search = GridSearchCV(estimator=model,
param_grid=param_grids[model_name], cv=5, scoring='accuracy', n_jobs=-
1)
    grid_search.fit(X_train, y_train)

    print(f"Best Parameters for {model_name}:
{grid search.best params }")
    print(f"Best Cross-Validation Accuracy:
{grid search.best score_:.4f}")
    print("-" * 60)

    # Store the best estimator
    best_estimators[model_name] = grid_search.best_estimator_
```

Θεωρητική Τεκμηρίωση:

- **Grid Search**:
    - Το **GridSearchCV** χρησιμοποιείται για την αναζήτηση των καλύτερων υπερπαραμέτρων ενός μοντέλου, δοκιμάζοντας όλους τους συνδυασμούς των προκαθορισμένων τιμών που ορίζονται στο param_grid.
    - Το αποτέλεσμα είναι το καλύτερο δυνατό μοντέλο για τις συγκεκριμένες παραμέτρους.
    - Το **cross-validation (cv=5)** χρησιμοποιείται για να αξιολογείται το μοντέλο σε διαφορετικά υποσύνολα των δεδομένων, διασφαλίζοντας ότι το μοντέλο δεν προσαρμόζεται υπερβολικά στα δεδομένα εκπαίδευσης.

- **Υπερπαράμετροι**: Οι υπερπαράμετροι είναι οι ρυθμίσεις που δεν μαθαίνονται από τα δεδομένα και πρέπει να καθοριστούν πριν την εκπαίδευση (π.χ., ο αριθμός γειτόνων στο KNN ή η πολυπλοκότητα στο δέντρο απόφασης). Η **βελτιστοποίηση των υπερπαραμέτρων** είναι κρίσιμη για τη βελτίωση της ακρίβειας του μοντέλου.

6.1.10 Αξιολόγηση Απόδοσης (Evaluation Metrics)

```
# Evaluate on the test set
y_pred = grid_search.best_estimator_.predict(X_test)
accuracy = accuracy score(y test, y pred)
print(f"Test Accuracy for {model_name}: {accuracy:.4f}")
```

Θεωρητική Τεκμηρίωση:

- **Accuracy (Ορθότητα)**: Είναι το ποσοστό των σωστών προβλέψεων σε σχέση με το συνολικό αριθμό των προβλέψεων. Αν και είναι μια απλή μετρική, δεν είναι πάντα η καλύτερη όταν έχουμε μη ισορροπημένα δεδομένα.

```
# Classification Report
print("Classification Report:")
print(classification_report(y_test, y_pred))
```

- **Classification Report**: Προσφέρει περισσότερα δεδομένα, συμπεριλαμβανομένων των ακόλουθων στοιχείων:
  - **Precision** (ακρίβεια): Το ποσοστό των πραγματικά ακριβών θετικών προβλέψεων.
  - **Recall** (ανάκληση): Το ποσοστό ανίχνευσης του μοντέλου των πραγματικά θετικών παραδειγμάτων.
  - **F1-score**: Ο μέσος όρος της ακρίβειας και της ανάκλησης, ένας πιο αξιόπιστος δείκτης για ανισομερή δεδομένα.

6.1.11 Μήτρα Σύγχυσης (Confusion Matrix)

```
# Confusion Matrix
cm = confusion matrix(y test, y pred)
sns.heatmap(cm, annot=True, fmt="d", cmap="Blues")
plt.title(f"Confusion Matrix for {model_name}")
plt.xlabel("Predicted")
plt.ylabel("Actual")
plt.show()
```

Θεωρητική Τεκμηρίωση:

- **Confusion Matrix**: Δείχνει τον αριθμό των σωστών και λανθασμένων προβλέψεων, ταξινομημένα ανά κατηγορία. Αποτελείται από:

- **True Positives (TP)**: Οι προβλέψεις που είναι θετικές και σωστές.
- **True Negatives (TN)**: Οι προβλέψεις που είναι αρνητικές και σωστές.
- **False Positives (FP)**: Οι προβλέψεις που ήταν θετικές, αλλά εν τελει είναι λάθος (ψευδής συναγερμός).
- **False Negatives (FN)**: Οι προβλέψεις που ήταν αρνητικές, αλλά εν τέλει είναι και λάθος (χάθηκε η πραγματική θετική).

6.1.12 Καμπύλη ROC και AUC (ROC Curve and AUC)

```
    # ROC Curve
    if hasattr(grid_search.best_estimator_, "predict_proba"):
```

```
        y_prob = grid_search.best_estimator_.predict_proba(X_test)[:,
1]
    else:  # For SVM, use decision function
        y_prob = grid_search.best_estimator_.decision_function(X_test)

    fpr, tpr, thresholds = roc_curve(y_test, y_prob)
    roc_auc = auc(fpr, tpr)

    plt.figure()
    plt.plot(fpr, tpr, color='darkorange', lw=2, label=f'ROC curve
(area = {roc auc:.2f})')
    plt.plot([0, 1], [0, 1], color='navy', lw=2, linestyle='--')
    plt.xlim([0.0, 1.0])
    plt.ylim([0.0, 1.05])
    plt.xlabel('False Positive Rate')
    plt.ylabel('True Positive Rate')
    plt.title(f'Receiver Operating Characteristic for {model_name}')
    plt.legend(loc="lower right")
    plt.show()

    print("=" * 60)
```

Θεωρητική Τεκμηρίωση:

- **Καμπύλη ROC (Receiver Operating Characteristic)**:
    - Η καμπύλη ROC - ένα γραφικό εργαλείο - βοηθά στην οπτικοποίηση της απόδοσής της, ποσοτικοποιώντας την ικανότητα ενός δυαδικού ταξινομητή να διακρίνει τα θετικά από τα αρνητικά δεδομένα.
    - Στον οριζόντιο άξονα βρίσκεται το ποσοστό ψευδώς θετικών δειγμάτων (False Positive Rate - FPR), δηλαδή το ποσοστό των αρνητικών δειγμάτων που έχουν ταξινομηθεί εσφαλμένα ως θετικά. Το True Positive Rate (TPR), δηλαδή το ποσοστό των θετικών δειγμάτων που προβλέφθηκαν με ακρίβεια ως θετικά, βρίσκεται στον κατακόρυφο άξονα.

- **AUC (Area Under the Curve)**:
    - Το AUC είναι η περιοχή κάτω από την καμπύλη ROC και αποτελεί έναν αριθμητικό δείκτη της απόδοσης του ταξινομητή. Ένα AUC κοντά στο 1.0 δείχνει άριστη απόδοση, ενώ ένα AUC κοντά στο 0.5 δείχνει ότι το μοντέλο δεν είναι καλύτερο από την τυχαία πρόβλεψη.
- **Χρήση του predict_proba**:
    - Το Predict_proba παράγει, για κάθε κλάση, την πιθανότητα που αντιστοιχεί στις προβλέψεις του μοντέλου. Με δυαδική ταξινόμηση, το μοντέλο παράγει τις πιθανότητες της κλάσης 0 και 1. Εφαρμόζουμε την πιθανότητα της κλάσης 1.
    - Χρησιμοποιούμε την decision_function για μοντέλα όπως το SVM που δεν υπολογίζουν πιθανότητες, καθώς επιστρέφει τιμές που υποδεικνύουν την απόσταση που έχει ένα σημείο από το όριο απόφασης.

6.1.13 Σύγκριση Τελικών Ακρίβειων (Final Test Accuracies)

```
# Store final test accuracies
final_accuracies = {}

# After all models, you can compare their test accuracy
for model name, estimator in best estimators.items():
    y_pred = estimator.predict(X_test)
    accuracy = accuracy score(y test, y pred)
    final_accuracies[model_name] = accuracy
    print(f"Final Test Accuracy for {model_name}: {accuracy:.4f}")
```

Θεωρητική Τεκμηρίωση:

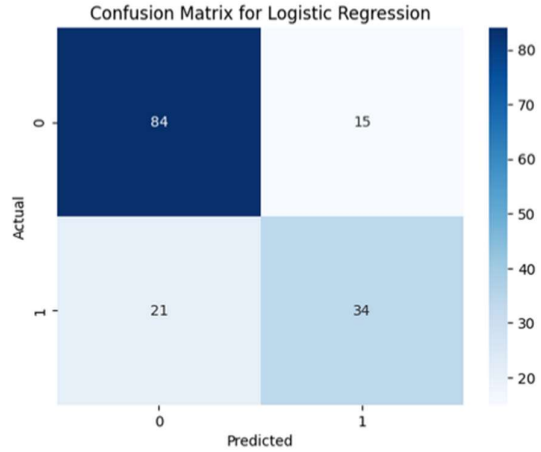
Σύγκριση Μοντέλων:

- Συγκεντρώνουμε την τελική ακρίβεια (test accuracy) σε μια λίστα (final_accuracies) μετά την εκπαίδευση και την αξιολόγηση κάθε μοντέλου. Για το ίδιο σύνολο δεδομένων, η διαδικασία αυτή επιτρέπει την αξιολόγηση της απόδοσης διαφόρων αλγορίθμων.

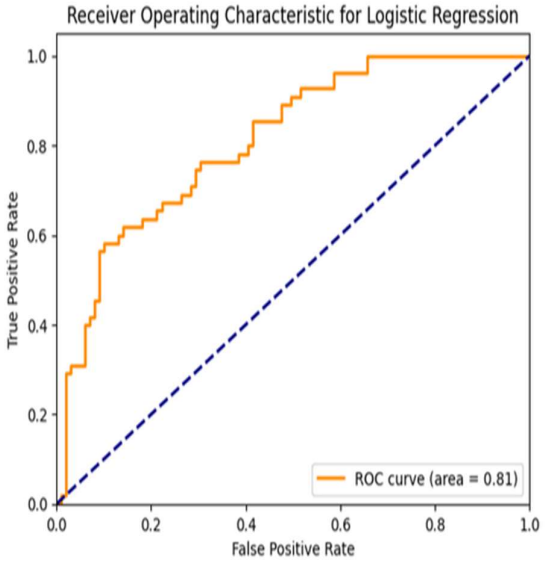
Μετά την εκτέλεση του παραπάνω κομματιού κώδικα (υποκεφάλαια 6.1.7-6.1.13), παίρνουμε αναλυτικά για όλα τα μοντέλα που μελετάμε 2 διαγράμματα και μια αναφορά ταξινόμησης με όλα τα δεδομένα και αποτελέσματα όπως το precision, recall, f1-score και accuracy . Το ένα διάγραμμα είναι ένας πίνακας σύγχυσης (confusion matrix) που δείχνει τον αριθμό των σωστών και λανθασμένων προβλέψεων, ταξινομημένα ανά κατηγορία (αναφέρθηκε και παραπάνω). Και το άλλο διάγραμμα, είναι η καμπύλη ROC (Receiver Operating Characteristic), που αναλύθηκε παραπάνω στο 6.1.12 για τη χρήση του. Παρακάτω δίνονται με τη σειρά για όλα τα μοντέλα μηχανικής μάθησης τα συγκεκριμένα διαγράμματα και τα αποτελέσματα.

Logistic Regression:

```
Test Accuracy for Logistic Regression: 0.7662
Classification Report:
              precision    recall  f1-score   support

           0       0.80      0.85      0.82        99
           1       0.69      0.62      0.65        55

    accuracy                           0.77       154
   macro avg       0.75      0.73      0.74       154
weighted avg       0.76      0.77      0.76       154
```

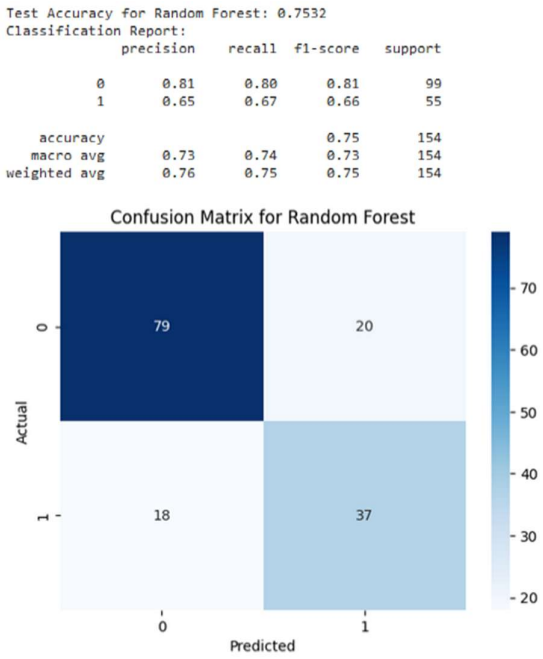


Γράφημα 7: Αναφορά ταξινόμησης με τα αποτελέσματα precision, recall, f1-score και accuracy και διάγραμμα με πίνακα σύγχυσης που δείχνει τον αριθμό των σωστών και λανθασμένων προβλέψεων για το Logistic Regression.

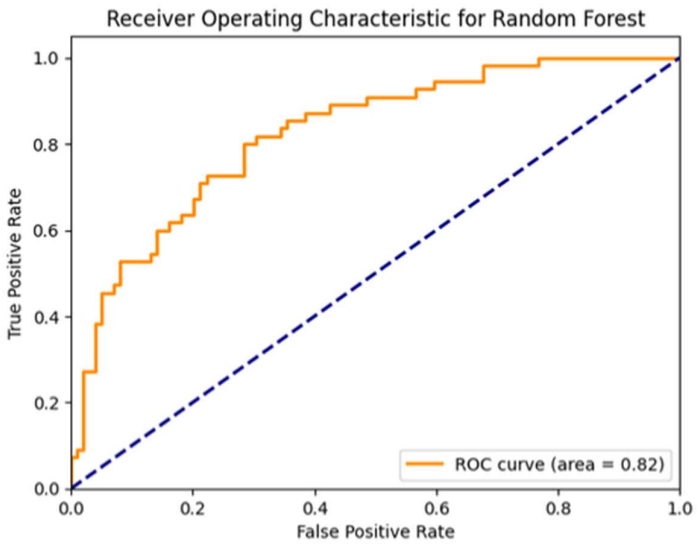


Γράφημα 8: Διάγραμμα με την καμπύλη ROC (Receiver Operating Characteristic) για το Logistic Regression.

Random Forest:

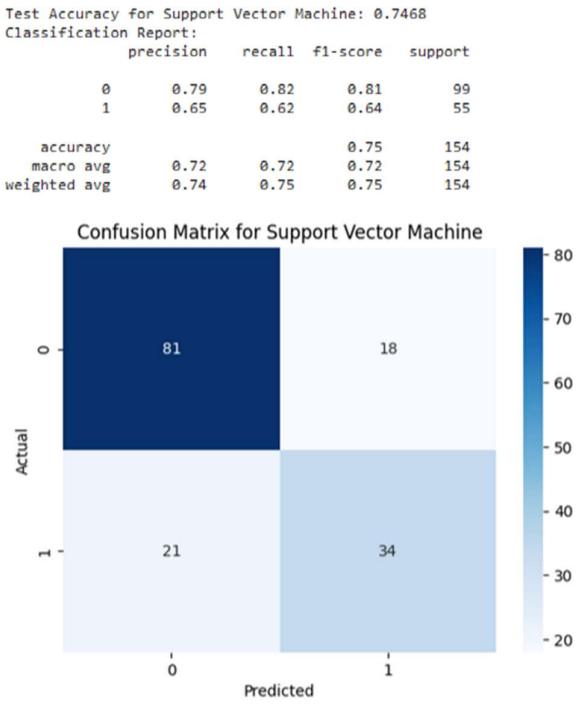


Γράφημα 9: Αναφορά ταξινόμησης με τα αποτελέσματα precision, recall, f1-score και accuracy και διάγραμμα με πίνακα σύγχυσης που δείχνει τον αριθμό των σωστών και λανθασμένων προβλέψεων για το Random Forest.

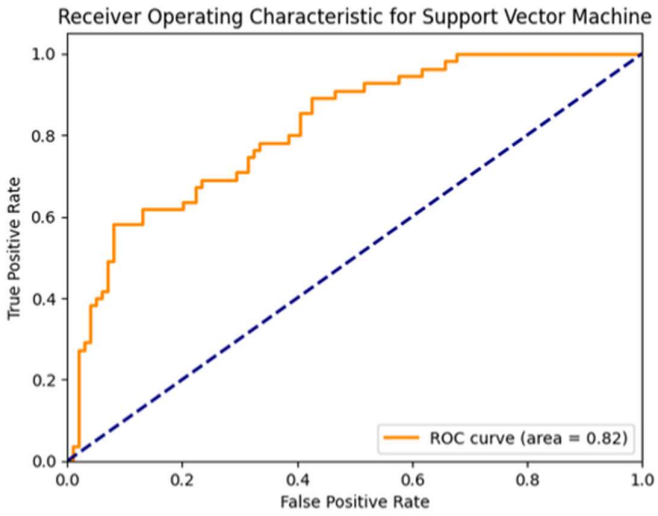


Γράφημα 10: Διάγραμμα με την καμπύλη ROC (Receiver Operating Characteristic) για το Random Forest

Support Vector Machine (SVM):



Γράφημα 11: Αναφορά ταξινόμησης με τα αποτελέσματα precision, recall, f1-score και accuracy και διάγραμμα με πίνακα σύγχυσης που δείχνει τον αριθμό των σωστών και λανθασμένων προβλέψεων για το Support Vector Machine.

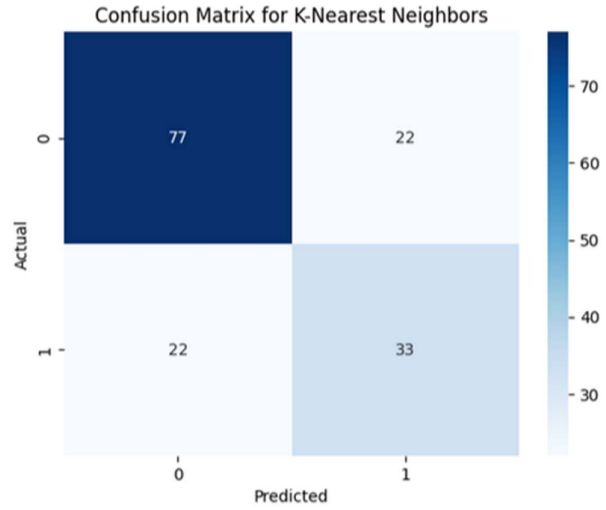


Γράφημα 12: Διάγραμμα με την καμπύλη ROC (Receiver Operating Characteristic) για το Support Vector Machine.

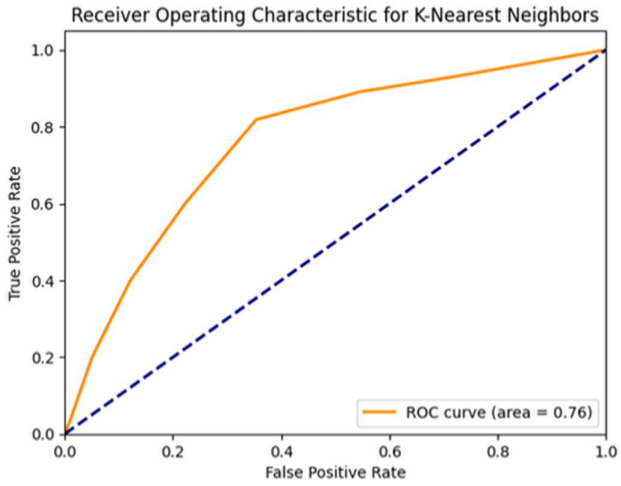
K-Nearest Neighbors:

Test Accuracy for K-Nearest Neighbors: 0.7143
Classification Report:

	precision	recall	f1-score	support
0	0.78	0.78	0.78	99
1	0.60	0.60	0.60	55
accuracy			0.71	154
macro avg	0.69	0.69	0.69	154
weighted avg	0.71	0.71	0.71	154

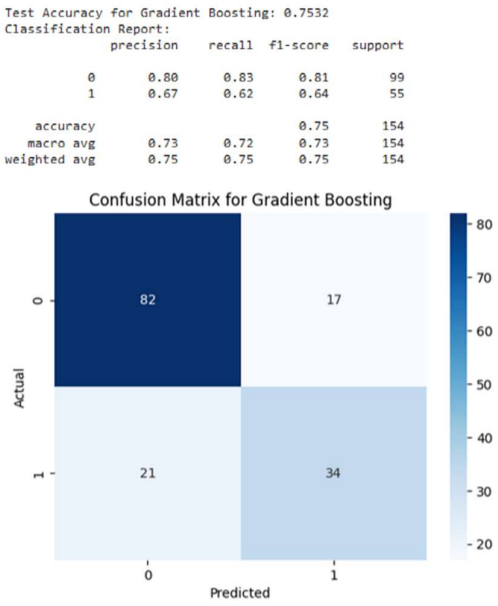


Γράφημα 13: Αναφορά ταξινόμησης με τα αποτελέσματα precision, recall, f1-score και accuracy και διάγραμμα με πίνακα σύγχυσης που δείχνει τον αριθμό των σωστών και λανθασμένων προβλέψεων για το K-Nearest Neighbors.

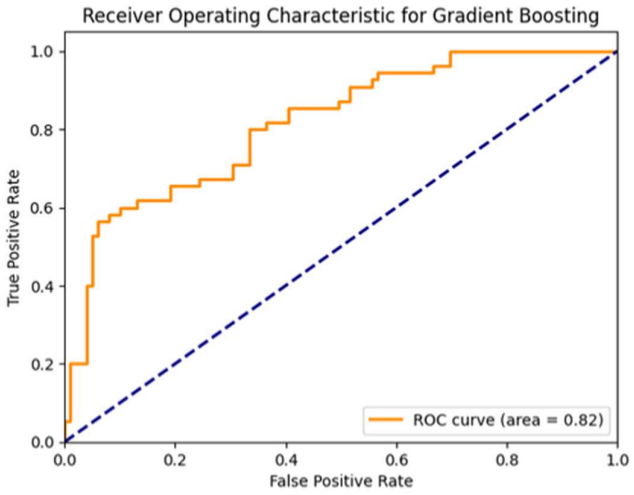


Γράφημα 14: Διάγραμμα με την καμπύλη ROC (Receiver Operating Characteristic) για το K-Nearest Neighbors.

Gradient Boosting:

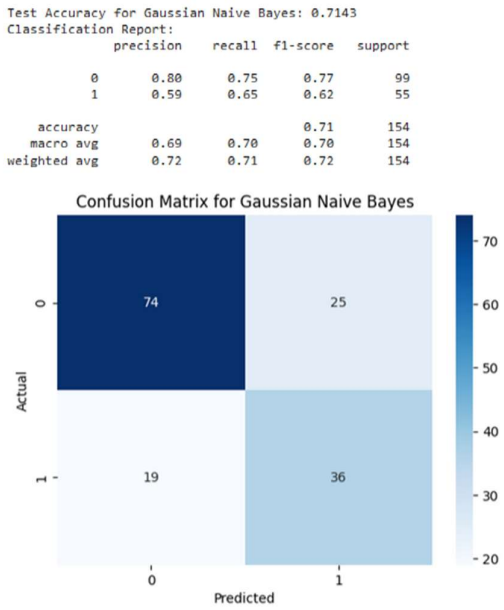


Γράφημα 15: Αναφορά ταξινόμησης με τα αποτελέσματα precision, recall, f1-score και accuracy και διάγραμμα με πίνακα σύγχυσης που δείχνει τον αριθμό των σωστών και λανθασμένων προβλέψεων για το Gradient Boosting.

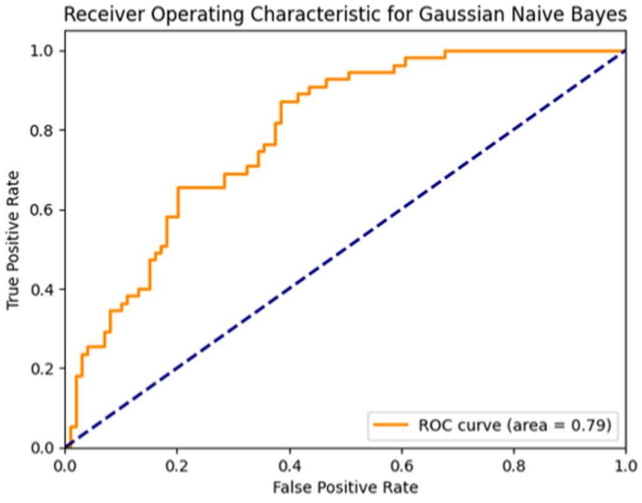


Γράφημα 16: Διάγραμμα με την καμπύλη ROC (Receiver Operating Characteristic) για το Gradient Boosting.

Naive Bayes:

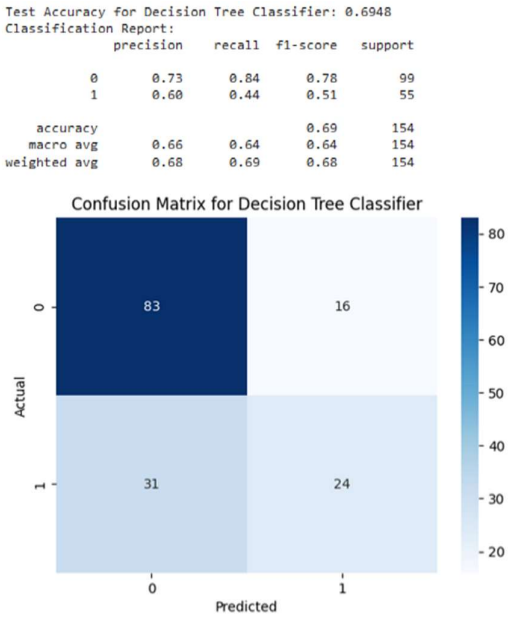


Γράφημα 17: Αναφορά ταξινόμησης με τα αποτελέσματα precision, recall, f1-score και accuracy και διάγραμμα με πίνακα σύγχυσης που δείχνει τον αριθμό των σωστών και λανθασμένων προβλέψεων για το Naïve Bayes.

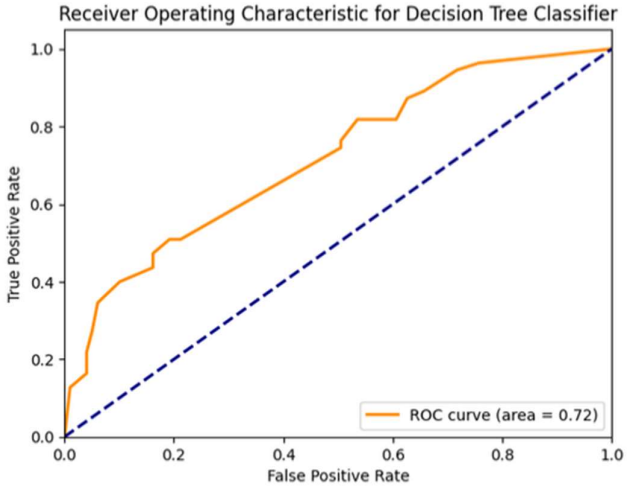


Γράφημα 18: Διάγραμμα με την καμπύλη ROC (Receiver Operating Characteristic) για το Naïve Bayes.

Decision Tree:

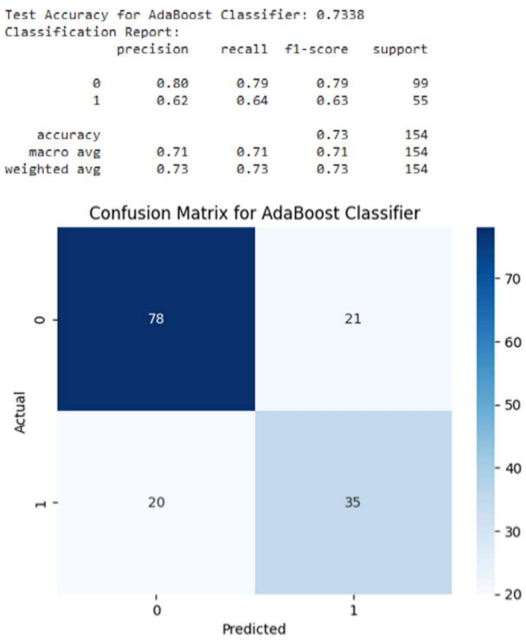


Γράφημα 19: Αναφορά ταξινόμησης με τα αποτελέσματα precision, recall, f1-score και accuracy και διάγραμμα με πίνακα σύγχυσης που δείχνει τον αριθμό των σωστών και λανθασμένων προβλέψεων για το Decision Tree.

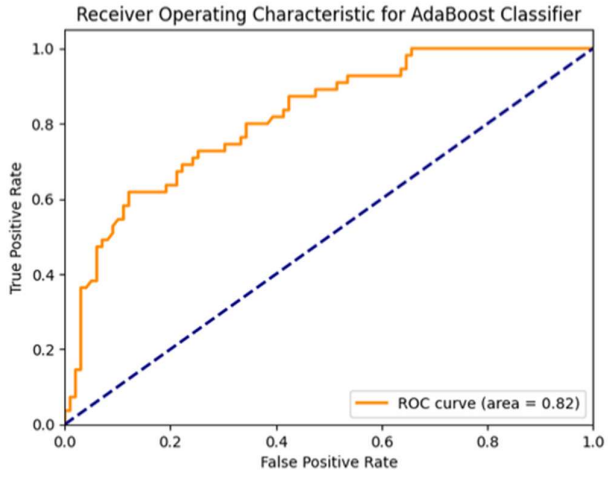


Γράφημα 20: Διάγραμμα με την καμπύλη ROC (Receiver Operating Characteristic) για το Decision Tree.

AdaBoost:



Γράφημα 21: Αναφορά ταξινόμησης με τα αποτελέσματα precision, recall, f1-score και accuracy και διάγραμμα με πίνακα σύγχυσης που δείχνει τον αριθμό των σωστών και λανθασμένων προβλέψεων για το Adaboost.



Γράφημα 22: Διάγραμμα με την καμπύλη ROC (Receiver Operating Characteristic) για το Adaboost.

6.1.14 Οπτικοποίηση Ακριβειών (Visualization of Accuracies)

```
import matplotlib.pyplot as plt
import numpy as np


# Assuming best_estimators and other variables are already defined


# Plotting the test accuracies
model_names = list(final_accuracies.keys())
accuracies = list(final_accuracies.values())

plt.figure(figsize=(10, 6))
plt.barh(model_names, accuracies, color='skyblue')
```

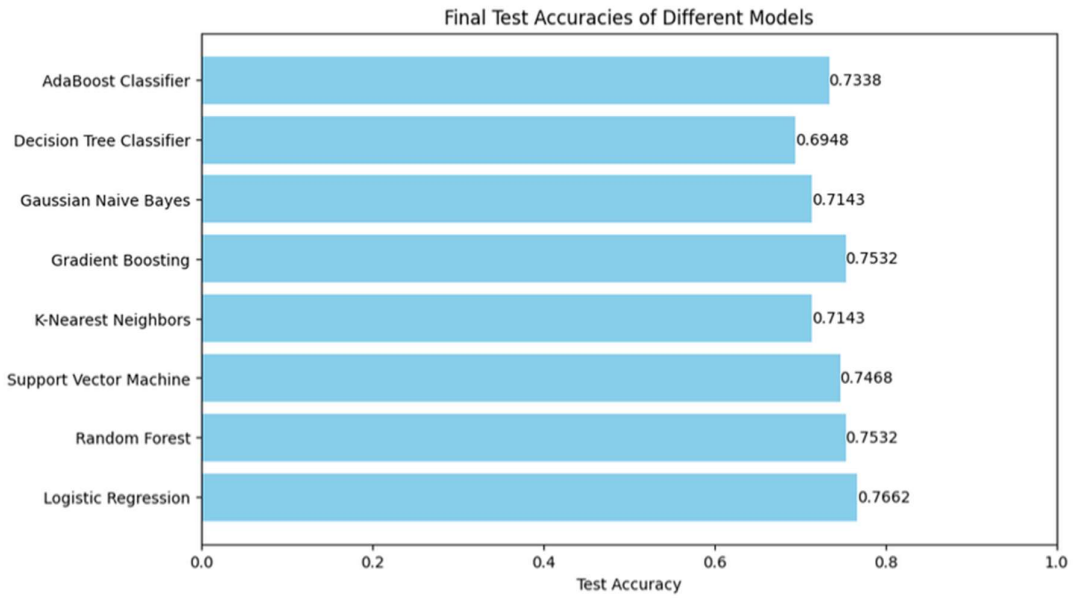
```
plt.xlabel('Test Accuracy')
plt.title('Final Test Accuracies of Different Models')
plt.xlim(0, 1)  # Accuracy range is between 0 and 1

# Annotate the bars with accuracy values
for index, value in enumerate(accuracies):
    plt.text(value, index, f'{value:.4f}', va='center', ha='left',
color='black')

plt.show()
```

Θεωρητική Τεκμηρίωση:

Οπτικοποίηση:

- Η χρήση ενός **οριζόντιου διαγράμματος (bar chart)** για την παρουσίαση της ακρίβειας του μοντέλου. Ποια μοντέλα, στο test set, έχουν μεγαλύτερη ακρίβεια; Το γράφημα απεικονίζει αυτό με σαφήνεια.
- Προστίθεται κείμενο σε κάθε μπάρα με την ακριβή τιμή της ακρίβειας, για να διευκολύνει την ανάγνωση.

Παρακάτω, δίνεται το οριζόντιο διάγραμμα, που παρουσιάζει την ακρίβεια που πετύχαμε στον εκάστοτε αλγόριθμο.



Γράφημα 23: Οριζόντιο διάγραμμα που παρουσιάζει την ακρίβεια στον εκάστοτε αλγόριθμο που υλοποιήσαμε.

Από το γράφημα, μπορούμε να κατατάξουμε τα μοντέλα που μελετήσαμε, ανάλογα με την ακρίβεια.

Logistic Regression: Το μοντέλο αυτό έχει την υψηλότερη ακρίβεια (0.7662), αποδεικνύοντας ότι είναι το πιο αποτελεσματικό στην πρόβλεψη.

Gradient Boosting & Random Forest: Τα δύο αυτά μοντέλα έχουν την ίδια ακρίβεια (0.7532) και ακολουθούν το Logistic Regression.

Support Vector Machine: Το SVM έχει ακρίβεια 0.7468.

AdaBoost Classifier: Το AdaBoost έχει ακρίβεια 0.7338.

Gaussian Naive Bayes: Το μοντέλο αυτό έχει ακρίβεια 0.7143.

K-Nearest Neighbors: Παρόμοια ακρίβεια με το Gaussian Naive Bayes (0.7143).

Decision Tree Classifier: Το Decision Tree έχει τη χαμηλότερη ακρίβεια (0.6948) από όλα τα μοντέλα.

Συμπερασματικά, προκύπτει ότι το Logistic Regression φαίνεται να είναι το καλύτερο μοντέλο για την συγκεκριμένη εργασία, καθώς έχει την υψηλότερη ακρίβεια. Το Decision Tree Classifier υστερεί σε σχέση με τα υπόλοιπα μοντέλα στη διάγνωση του διαβήτη.

6.1.15 Εκπαίδευση και Αξιολόγηση ενός Βαθύ Νευρωνικού Δικτύου (Deep Neural Network - DNN)

```
import tensorflow as tf
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Dense, Dropout
from tensorflow.keras.optimizers import Adam
from sklearn.preprocessing import StandardScaler

# Standardize the dataset
scaler = StandardScaler()
X train scaled = scaler.fit transform(X train)
X_test_scaled = scaler.transform(X_test)

# Build the Deep Neural Network
def build_dnn_model(input_shape):
    model = Sequential()
    model.add(Dense(128, input_shape=(input_shape,),
activation='relu'))
    model.add(Dropout(0.3))
```

```
    model.add(Dense(64, activation='relu'))
    model.add(Dropout(0.3))
    model.add(Dense(32, activation='relu'))
    model.add(Dropout(0.3))
    model.add(Dense(1, activation='sigmoid'))  # Binary classification
    return model

# Compile and train the model
dnn model = build dnn model(X train scaled.shape[1])
dnn_model.compile(optimizer=Adam(learning_rate=0.001),
loss='binary_crossentropy', metrics=['accuracy'])

# Train the model
history = dnn model.fit(X train scaled, y train, epochs=100,
batch_size=32, validation_split=0.2, verbose=1)
```

Θεωρητική Τεκμηρίωση:

- **Βαθύ Νευρωνικό Δίκτυο (Deep Neural Network)**:
    - Ένα DNN αποτελείται από πολλά επίπεδα νευρώνων (layers), καθένα από τα οποία μαθαίνει πιο περίπλοκες αναπαραστάσεις των δεδομένων Στο συγκεκριμένο παράδειγμα:
        - Το πρώτο επίπεδο έχει 128 νευρώνες και ενεργοποίηση ReLU (Rectified Linear Unit), το οποίο είναι δημοφιλές για τη μείωση του προβλήματος της εξαφάνισης του κλίσης (vanishing gradient).
        - Το Dropout εισάγει τυχαία αφαίρεση νευρώνων κατά τη διάρκεια της εκπαίδευσης, αποτρέποντας την υπερεκπαίδευση (overfitting).
        - Κατάλληλο για δυαδική ταξινόμηση, το τελευταίο στρώμα αποτελείται από έναν νευρώνα με ενεργοποίηση sigmoid.

- **Εκπαίδευση και Validation**:
    - Το **Binary Crossentropy** είναι η συνάρτηση κόστους (loss) που χρησιμοποιούμε, κατάλληλη για δυαδική ταξινόμηση.
    - Το **validation split** διαιρεί το σύνολο εκπαίδευσης έτσι ώστε να αξιολογείται το μοντέλο καθ' όλη τη διάρκεια της ανάπτυξης.

6.1.16 Αξιολόγηση του DNN στο Test Set

```
# Evaluate the DNN model on the test set
dnn_test_loss, dnn_test_accuracy = dnn_model.evaluate(X_test_scaled,
y test, verbose=0)
print(f"Final Test Accuracy for Deep Neural Network:
{dnn_test_accuracy:.4f}")
```

Θεωρητική Τεκμηρίωση:

Αξιολόγηση σε Νέα Δεδομένα:

- Μετά την εκπαίδευση, το νευρωνικό δίκτυο αξιολογείται στο test set, για να παρατηρηθεί η απόδοση σε δεδομένα που δεν παρατηρήθηκαν κατά την εκπαίδευση.

6.1.17 Τελική Οπτικοποίηση με την Προσθήκη του DNN

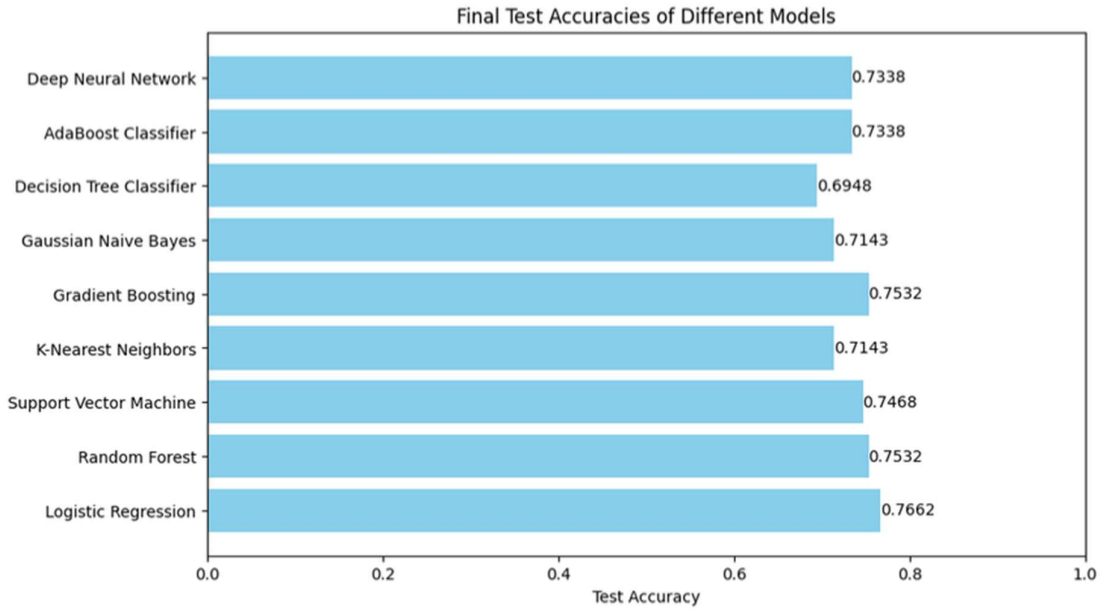
```
# Add DNN accuracy to the final accuracies dictionary
final_accuracies["Deep Neural Network"] = dnn_test_accuracy

# Plotting the test accuracies including DNN
model names = list(final accuracies.keys())
accuracies = list(final_accuracies.values())

plt.figure(figsize=(10, 6))
plt.barh(model_names, accuracies, color='skyblue')
plt.xlabel('Test Accuracy')
plt.title('Final Test Accuracies of Different Models')
plt.xlim(0, 1)  # Accuracy range is between 0 and 1

# Annotate the bars with accuracy values
for index, value in enumerate(accuracies):
    plt.text(value, index, f'{value:.4f}', va='center', ha='left',
color='black')

plt.show()
```

Θεωρητική Τεκμηρίωση:

- **Προσθήκη DNN στη Σύγκριση**:
    - Η συμπερίληψη της ακρίβειας του DNN στη λίστα final_accuracies μας επιτρέπει να αξιολογήσουμε κάθε μοντέλο, συμπεριλαμβανομένου του νευρωνικού δικτύου, συνολικά.

- **Οπτικοποίηση**:
    - Η τελευταία σύγκριση παρουσιάζει κάθε μοντέλο που δοκιμάστηκε, καθοδηγώντας έτσι τον εντοπισμό του καλύτερου για την τρέχουσα πρόκληση.

Στα υποκεφάλαια 6.1.15-6.1.17, μελετήσαμε και δημιουργήσαμε ένα νευρωνικό δίκτυο, συγκεκριμένα ένα Βαθύ Νευρωνικό Δίκτυο (DNN), που έχει στο πρώτο επίπεδο 128 νευρώνες και ενεργοποίηση ReLU (Rectified Linear Unit), το οποίο είναι δημοφιλές για τη μείωση του προβλήματος της εξαφάνισης του κλίσης. Το τελευταίο στρώμα αποτελείται από έναν νευρώνα με ενεργοποίηση sigmoid. Μετά την εκπαίδευση του δικτύου, συγκρίνουμε τα αποτελέσματα που βγάζει με τα προηγούμενα μοντέλα μηχανικής μάθησης που μελετήσαμε, και δημιουργούμε ένα οριζόντιο διάγραμμα που απεικονίζει όλες τις τελικές ακρίβειες του κάθε μοντέλου.



Γράφημα 24: Οριζόντιο διάγραμμα που παρουσιάζει την ακρίβεια στον εκάστοτε αλγόριθμο και το Deep Neural Network που υλοποιήσαμε.

Το Deep Neural Network βλέπουμε ότι πετυχαίνει ακρίβεια 0.7338. Είναι λίγο χαμηλότερη από τα μοντέλα Logistic Regression, Gradient Boosting, Random Forest και Support Vector Machine. Παρόλο που το Deep Neural Network δεν είχε την υψηλότερη απόδοση, η χρήση βαθιάς μάθησης (deep learning) θα μπορούσε να βελτιωθεί με την προσθήκη περισσότερων δεδομένων, την ρύθμιση των υπερπαραμέτρων (hyperparameter tuning) ή με την χρήση πιο σύνθετων αρχιτεκτονικών δικτύων.

7. Επίλογος – Προτάσεις Μελλοντικής Έρευνας

Ο σακχαρώδης διαβήτης αποτελεί μια χρόνια και ολοένα αυξανόμενη νόσο, η οποία επηρεάζει εκατομμύρια ανθρώπους παγκοσμίως. Ο σύγχρονος τρόπος ζωής, που χαρακτηρίζεται από καθιστική ζωή, ανθυγιεινή διατροφή και αυξημένα ποσοστά παχυσαρκίας, συμβάλλει στη ραγδαία αύξηση των διαβητικών περιστατικών. Εάν δεν ληφθούν έγκαιρα προληπτικά μέτρα, οι επιπτώσεις στην υγεία μπορεί να είναι σοβαρές, οδηγώντας σε καρδιαγγειακές παθήσεις, νεφροπάθεια και άλλες χρόνιες επιπλοκές.

Στην παρούσα διπλωματική εργασία, διερευνήθηκε η χρήση μοντέλων μηχανικής μάθησης για την έγκαιρη διάγνωση και πρόβλεψη του σακχαρώδους διαβήτη. Μέσω της ανάλυσης σχετικών ερευνών και της εφαρμογής μοντέλων σε δεδομένα, όπως η βάση δεδομένων των PIMA Indians, διαπιστώθηκε ότι οι αλγόριθμοι μηχανικής μάθησης μπορούν να επιτύχουν ακρίβεια πρόβλεψης που αγγίζει το 80%. Αυτό επιβεβαιώθηκε και μέσω των προσομοιώσεων που πραγματοποιήθηκαν μέσω κώδικα Python με χρήση διαφορετικών μοντέλων μηχανικής μάθησης. Ωστόσο, υπάρχουν σημαντικά περιθώρια βελτίωσης, ιδιαίτερα όσον αφορά τη μείωση των σφαλμάτων και την αύξηση της αξιοπιστίας των προβλέψεων.

Μελλοντικές Κατευθύνσεις

Για τη βελτίωση των αποτελεσμάτων και την περαιτέρω αξιοποίηση της μηχανικής μάθησης στη διάγνωση του σακχαρώδους διαβήτη, μπορούν να εξεταστούν οι ακόλουθες προσεγγίσεις:

- **Συνδυασμός Αλγορίθμων (Ensemble Learning):** Η ενσωμάτωση τεχνικών όπως το Random Forest, το Boosting (XGBoost, AdaBoost) και το Stacking μπορεί να συμβάλει στη βελτίωση της απόδοσης των μοντέλων.

- **Εμπλουτισμός των Δεδομένων:** Η χρήση μεγαλύτερων και πιο ποικιλόμορφων συνόλων δεδομένων, που περιλαμβάνουν γενετικούς και περιβαλλοντικούς παράγοντες, μπορεί να οδηγήσει σε πιο ακριβείς προβλέψεις.

- **Χρήση Βαθιάς Μάθησης (Deep Learning):** Τα νευρωνικά δίκτυα, όπως τα Convolutional Neural Networks (CNN) και τα Recurrent Neural Networks (RNN), ενδέχεται να βελτιώσουν την ακρίβεια των διαγνώσεων, ιδίως σε περιπτώσεις πολύπλοκων ιατρικών δεδομένων.

- **Ενσωμάτωση Δεδομένων από Wearable Devices:** Οι φορητές συσκευές παρακολούθησης υγείας μπορούν να παρέχουν συνεχή ροή δεδομένων για τη βελτίωση των μοντέλων πρόβλεψης και την ανάπτυξη εξατομικευμένων παρεμβάσεων.

- **Εξήγηση των Αποφάσεων των Μοντέλων (Explainable AI):** Η ανάπτυξη ερμηνεύσιμων μοντέλων θα επιτρέψει στους ιατρούς να κατανοούν καλύτερα τα αποτελέσματα των προβλέψεων και να λαμβάνουν πιο τεκμηριωμένες αποφάσεις.

Η τεχνολογία της μηχανικής μάθησης μπορεί να αποτελέσει πολύτιμο εργαλείο στη μάχη κατά του σακχαρώδους διαβήτη, συμβάλλοντας στην έγκαιρη διάγνωση και στην καλύτερη διαχείριση της νόσου. Με την περαιτέρω έρευνα και ανάπτυξη, τα συστήματα τεχνητής νοημοσύνης μπορούν να προσφέρουν ακόμα μεγαλύτερη ακρίβεια και αξιοπιστία, ενισχύοντας την ποιότητα της ιατρικής περίθαλψης και προάγοντας την προληπτική ιατρική.

8. Παράρτημα - Κώδικας Python

Παρακάτω, δίνεται ο κώδικας εξ΄ ολοκλήρου και μαζεμένα της προσομοίωσης που υλοποιήθηκε κατά τη διάρκεια μελέτης της διάγνωσης, πρόβλεψης και θεραπείας του διαβήτη.

```
import pandas as pd
import tensorflow as tf
import matplotlib.pyplot as plt
import numpy as np
import seaborn as sns
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler
from sklearn.linear_model import LogisticRegression
from    sklearn.metrics    import    accuracy_score,    classification_report,
confusion_matrix
from sklearn.model_selection import GridSearchCV
from    sklearn.metrics    import    accuracy_score,    classification_report,
confusion_matrix, roc_curve, auc
from sklearn.model_selection import cross_val_score
from sklearn.ensemble import RandomForestClassifier
from sklearn.svm import SVC
from sklearn.neighbors import KNeighborsClassifier
from sklearn.ensemble import GradientBoostingClassifier
from sklearn.naive_bayes import GaussianNB
from sklearn.tree import DecisionTreeClassifier
from sklearn.ensemble import AdaBoostClassifier
from xgboost import XGBClassifier
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Dense, Dropout
from tensorflow.keras.optimizers import Adam
from sklearn.preprocessing import StandardScaler

# Load the dataset
url    =    "https://raw.githubusercontent.com/jbrownlee/Datasets/master/pima-
indians-diabetes.data.csv"
column_names  =  ['Pregnancies',  'Glucose',  'BloodPressure',  'SkinThickness',
'Insulin', 'BMI', 'DiabetesPedigreeFunction', 'Age', 'Outcome']

# Read the data into a pandas dataframe
data = pd.read_csv(url, header=None, names=column_names)

# Distribution of each feature
data.hist(bins=10, figsize=(12, 10))
plt.tight_layout()
plt.show()
```

```
# Correlation matrix

plt.figure(figsize=(10, 8))

sns.heatmap(data.corr(), annot=True, cmap='coolwarm')
plt.show()

# Pairplot to see the distribution of features and their relationships
sns.pairplot(data, hue='Outcome')
plt.show()

# Replace zeros with NaN for specific columns
columns_to_replace = ['Glucose', 'BloodPressure', 'SkinThickness', 'Insulin',
'BMI']
data[columns_to_replace] = data[columns_to_replace].replace(0, pd.NA)

# Fill NaN with the median of each column
data[columns_to_replace]    =    data[columns_to_replace].apply(lambda    col:
col.fillna(col.median()))

# Check again for any missing values
print(data.isnull().sum())

# Example of creating a new feature: BMI and Age interaction
data['BMI_Age_Interaction'] = data['BMI'] * data['Age']

# Standardize all features (after creating new ones)

X = data.drop('Outcome', axis=1)
y = data['Outcome']

scaler = StandardScaler()
X = scaler.fit_transform(X)

# Assuming X and y are your features and target variable
# Split data into training and test sets
X_train,  X_test,  y_train,  y_test  =  train_test_split(X,  y,  test_size=0.2,
random_state=42)

# Initialize models
models = {
  "Logistic Regression": LogisticRegression(),
  "Random Forest": RandomForestClassifier(),
  "Support Vector Machine": SVC(probability=True),  # Enable probability for
ROC curve
  "K-Nearest Neighbors": KNeighborsClassifier(),
  "Gradient Boosting": GradientBoostingClassifier(),
  "Gaussian Naive Bayes": GaussianNB(),
```

```
  "Decision Tree Classifier": DecisionTreeClassifier(),
  "AdaBoost Classifier": AdaBoostClassifier(),
       #     "XGB     Classifier":     XGBClassifier(use_label_encoder=False,
eval_metric='logloss')
}

# Parameter grids
param_grids = {
    "Logistic Regression": {
        'C': [0.1],
        'penalty': ['l1'],
        'solver': ['saga'],
        'max_iter': [100]
    },
    "Random Forest": {
        'n_estimators': [100],
        'max_depth': [10],
        'min_samples_split': [10],
        'min_samples_leaf': [4],
        'bootstrap': [True]
    },
    "Support Vector Machine": {
        'C': [0.1],
        'kernel': ['linear'],
        'gamma': ['scale'],
        'degree': [3],  # Used only for 'poly' kernel
        'coef0': [0.0]  # Used only for 'poly' and 'sigmoid'
    },
    "K-Nearest Neighbors": {
        'n_neighbors': [7],
        'weights': ['uniform'],
        'algorithm': ['auto'],
        'p': [1]  # 1 for Manhattan distance, 2 for Euclidean distance
    },
    "Gradient Boosting": {
        'n_estimators': [200],
        'learning_rate': [0.01],
        'max_depth': [3],
        'subsample': [0.8],
        'min_samples_split': [10],
        'min_samples_leaf': [4]
    },
    "Gaussian Naive Bayes": {
        'var_smoothing': [1e-9]
    },
    "Decision Tree Classifier": {
        'criterion': ['log_loss'],
        'splitter': ['random'],
```

```
        'max_depth': [10],
        'min_samples_split': [5],
        'min_samples_leaf': [4],
        'max_features': ['log2']
    },
    "AdaBoost Classifier": {
        'n_estimators': [100],
        'learning_rate': [0.1],
        'algorithm': ['SAMME.R']
    }

}
best_estimators = {}


# Perform GridSearchCV for each model
for model_name, model in models.items():
    print(f"Performing GridSearchCV for {model_name}...")
                    grid_search         =         GridSearchCV(estimator=model,
param_grid=param_grids[model_name], cv=5, scoring='accuracy', n_jobs=-1)
    grid_search.fit(X_train, y_train)

    print(f"Best Parameters for {model_name}: {grid_search.best_params_}")
    print(f"Best Cross-Validation Accuracy: {grid_search.best_score_:.4f}")
    print("-" * 60)

    # Store the best estimator
    best_estimators[model_name] = grid_search.best_estimator_

    # Evaluate on the test set
    y_pred = grid_search.best_estimator_.predict(X_test)
    accuracy = accuracy_score(y_test, y_pred)
    print(f"Test Accuracy for {model_name}: {accuracy:.4f}")
    # Classification Report
    print("Classification Report:")
    print(classification_report(y_test, y_pred))

    # Confusion Matrix
    cm = confusion_matrix(y_test, y_pred)
    sns.heatmap(cm, annot=True, fmt="d", cmap="Blues")
    plt.title(f"Confusion Matrix for {model_name}")
    plt.xlabel("Predicted")
    plt.ylabel("Actual")
    plt.show()

    # ROC Curve
    if hasattr(grid_search.best_estimator_, "predict_proba"):
        y_prob = grid_search.best_estimator_.predict_proba(X_test)[:, 1]
    else:  # For SVM, use decision_function
```

```
        y_prob = grid_search.best_estimator_.decision_function(X_test)
    fpr, tpr, thresholds = roc_curve(y_test, y_prob)
    roc_auc = auc(fpr, tpr)

    plt.figure()
     plt.plot(fpr, tpr, color='darkorange', lw=2, label=f'ROC curve (area =
{roc_auc:.2f})')
    plt.plot([0, 1], [0, 1], color='navy', lw=2, linestyle='--')
    plt.xlim([0.0, 1.0])
    plt.ylim([0.0, 1.05])
    plt.xlabel('False Positive Rate')
    plt.ylabel('True Positive Rate')
    plt.title(f'Receiver Operating Characteristic for {model_name}')
    plt.legend(loc="lower right")

plt.show()

print("=" * 60)

# Store final test accuracies
final_accuracies = {}

# After all models, you can compare their test accuracy
for model_name, estimator in best_estimators.items():
    y_pred = estimator.predict(X_test)
    accuracy = accuracy_score(y_test, y_pred)
    final_accuracies[model_name] = accuracy
    print(f"Final Test Accuracy for {model_name}: {accuracy:.4f}")
# Assuming best_estimators and other variables are already defined
# Plotting the test accuracies
model_names = list(final_accuracies.keys())
accuracies = list(final_accuracies.values())
plt.figure(figsize=(10, 6))
plt.barh(model_names, accuracies, color='skyblue')
plt.xlabel('Test Accuracy')
plt.title('Final Test Accuracies of Different Models')
plt.xlim(0, 1)  # Accuracy range is between 0 and 1

# Annotate the bars with accuracy values
for index, value in enumerate(accuracies):
        plt.text(value,   index,   f'{value:.4f}',   va='center',   ha='left',
color='black')

plt.show()

# Standardize the dataset
scaler = StandardScaler()
X_train_scaled = scaler.fit_transform(X_train)
```

```
X_test_scaled = scaler.transform(X_test)

# Build the Deep Neural Network
def build_dnn_model(input_shape):
    model = Sequential()
    model.add(Dense(128, input_shape=(input_shape,), activation='relu'))
    model.add(Dropout(0.3))
    model.add(Dense(64, activation='relu'))
    model.add(Dropout(0.3))
    model.add(Dense(32, activation='relu'))
    model.add(Dropout(0.3))
    model.add(Dense(1, activation='sigmoid'))  # Binary classification
    return model

# Compile and train the model
dnn_model = build_dnn_model(X_train_scaled.shape[1])
dnn_model.compile(optimizer=Adam(learning_rate=0.001),
loss='binary_crossentropy', metrics=['accuracy'])

# Train the model
history = dnn_model.fit(X_train_scaled, y_train, epochs=100, batch_size=32,
validation_split=0.2, verbose=1)

# Evaluate the DNN model on the test set
dnn_test_loss, dnn_test_accuracy = dnn_model.evaluate(X_test_scaled, y_test,
verbose=0)
print(f"Final Test Accuracy for Deep Neural Network: {dnn_test_accuracy:.4f}")

# Add DNN accuracy to the final accuracies dictionary
final_accuracies["Deep Neural Network"] = dnn_test_accuracy


# Plotting the test accuracies including DNN
model_names = list(final_accuracies.keys())
accuracies = list(final_accuracies.values())

plt.figure(figsize=(10, 6))
plt.barh(model_names, accuracies, color='skyblue')
plt.xlabel('Test Accuracy')
plt.title('Final Test Accuracies of Different Models')
plt.xlim(0, 1)  # Accuracy range is between 0 and 1

# Annotate the bars with accuracy values
for index, value in enumerate(accuracies):
        plt.text(value, index, f'{value:.4f}', va='center', ha='left',
color='black')

plt.show()
```

9. Βιβλιογραφία

[1] American Diabetes Association, "What is Diabetes?," American Diabetes Association, 2023, https://www.diabetes.org/diabetes

[2] Centers for Disease Control and Prevention, "Diabetes Basics," Centers for Disease Control and Prevention, 2022, https://www.cdc.gov/diabetes/basics/understanding-diabetes.html

[3] American Heart Association, "Heart Disease and Stroke Statistics," American Heart Association, 2022, https://www.ahajournals.org/doi/10.1161/CIR.0000000000001292

[4] American Kidney Fund, "Diabetes and Kidney Disease," American Kidney Fund, 2022, https://www.kidneyfund.org/kidney-disease-facts/diabetes-and-kidney-disease

[5] American Academy of Ophthalmology, "Diabetes and Eye Health," American Academy of Ophthalmology, 2023, https://www.aao.org/eye-health/diseases/diabetes

[6] C. M. Bishop, *Pattern Recognition and Machine Learning*. New York, NY: Springer, 2006.

[7] S. Shalev-Shwartz and S. Ben-David, *Understanding Machine Learning: From Theory to Algorithms*. Cambridge, UK: Cambridge University Press, 2014.

[8] Mitoglou and Koutsouris, "Επισκόπηση Ευφυών Μοντέλων και Τεχνικών Μηχανικής Μάθησης για την πρόβλεψη, διαχείριση και αντιμετώπιση του καρκίνου του μαστού", 2021

[9] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*. Cambridge, MA: MIT Press, 2018.

[10] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *NeurIPS*, 2017.

[11] X. Zhu and A. B. Goldberg, *Introduction to Semi-Supervised Learning*. San Rafael, CA: Morgan & Claypool, 2009.

[12] L. Breiman, "Random forests," *Machine Learning*, vol. 45, no. 1, pp. 5-32, 2001.

[13] T. G. Dietterich, "Ensemble methods in machine learning," in *International workshop on multiple classifier systems*, 2000, pp. 1-15.

[14] Y. Freund and R. E. Schapire, "A decision-theoretic generalization of on-line learning and an application to boosting," *Journal of Computer and System Sciences*, vol. 55, no. 1, pp. 119-139, 1997.

[15] D. H. Wolpert, "Stacked generalization," *Neural Networks*, vol. 5, no. 2, pp. 241-259, 1992.

[16] S. J. Pan and Q. Yang, "A survey on transfer learning," *IEEE Transactions on Knowledge and Data Engineering*, vol. 22, no. 10, pp. 1345-1359, 2010.

[17] E. Alpaydin, *Introduction to Machine Learning*, 2nd ed. Cambridge, MA: MIT Press, 2010.

[18] L. Breiman, J. Friedman, R. A. Olshen, and C. J. Stone, *Classification and Regression Trees*. Monterey, CA: Wadsworth & Brooks, 1986.

[19] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*, 2nd ed. New York, NY: Springer, 2009.

[20] Linear Regression in Machine Learning: A Comprehensive Guide, https://www.geeksforgeeks.org/ml-linear-regression/

[21] D. W. Hosmer, S. Lemeshow, and R. X. Sturdivant, *Applied Logistic Regression*, 3rd ed. Hoboken, NJ: Wiley, 2013.

[22] S. Menard, *Applied Logistic Regression Analysis*, 2nd ed. Thousand Oaks, CA: Sage, 2002.

[23] Logistic Regression, https://www.saedsayad.com/logistic_regression.htm

[24] D. C. Montgomery, E. A. Peck, and G. G. Vining, *Introduction to Linear Regression Analysis*, 5th ed. Hoboken, NJ: Wiley, 2012.

[25] D. M. W. Powers, "Evaluation: From precision, recall and F-factor to ROC, informedness, markedness & correlation," *Journal of Machine Learning Technologies*, vol. 2, no. 1, pp. 37-63, 2011.

[26] Decision Trees: A Powerful Tool in Machine Learning, https://medium.com/@nidhigahlawat/decision-trees-a-powerful-tool-in-machine-learning-dd0724dad4b6

[27] J. R. Quinlan, "Induction of decision trees," *Machine Learning*, vol. 1, no. 1, pp. 81-106, 1986.

[28] Random Forest Algorithm in Machine Learning, https://www.geeksforgeeks.org/random-forest-algorithm-in-machine-learning/

[29] K. Weiss, T. M. Khoshgoftaar, and D. Wang, "A survey of transfer learning," *Journal of Big Data*, vol. 3, no. 1, p. 9, 2016.

[30] C. Cortes and V. Vapnik, "Support-vector networks," *Machine Learning*, vol. 20, no. 3, pp. 273-297, 1995.

[31] Understand Support Cevtor Machines (SVMs), https://medium.com/@jainvidip/understanding-support-vector-machines-svms-1f7c78bad934

[32] Artificial Neural Networks and its Applications, https://www.geeksforgeeks.org/artificial-neural-networks-and-its-applications/

[33] N. Cristianini and J. Shawe-Taylor, *An Introduction to Support Vector Machines and Other Kernel-based Learning Methods*. Cambridge, UK: Cambridge University Press, 2000.

[34] V. N. Vapnik, *Statistical Learning Theory*. New York, NY: Wiley-Interscience, 1998.

[35] B. Schölkopf and A. J. Smola, *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*. Cambridge, MA: MIT Press, 2002.

[36] K. Cho, et al., "Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation," in *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*, 2014.

[37] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. Cambridge, MA: MIT Press, 2016.

[38] S. Haykin, *Neural Networks: A Comprehensive Foundation*. Upper Saddle River, NJ: Prentice Hall, 1999.

[39] R. Hecht-Nielsen, "Theory of the Backpropagation Neural Network," in *International Joint Conference on Neural Networks*, 1989.

[40] S. Hochreiter and J. Schmidhuber, "Long Short-Term Memory," *Neural Computation*, vol. 9, no. 8, pp. 1735-1780, 1997.

[41] G. E. Hinton, et al., "Reducing the Dimensionality of Data with Neural Networks," *Science*, vol. 313, no. 5786, pp. 504-507, 2006.

[42] Y. LeCun, et al., "Gradient-Based Learning Applied to Document Recognition," *Proceedings of the IEEE*, 1998.

[43] Y. LeCun, Y. Bengio, and G. Hinton, "Deep Learning," *Nature*, vol. 521, no. 7553, pp. 436-444, 2015.

[44] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning Representations by Backpropagating Errors," *Nature*, vol. 323, no. 6088, pp. 533-536, 1986.

[45] P. Werbos, "Backpropagation Through Time: What It Does and How to Do It," *Proceedings of the IEEE*, 1990.

[46] K-Nearest Neighbor (KNN) Algorithm, https://www.geeksforgeeks.org/k-nearest-neighbours/

[47] J. L. Bentley, "Multidimensional binary search trees used for associative searching," *Communications of the ACM*, vol. 18, no. 9, pp. 509–517, 19

[48] Cover, T., & Hart, P. (1967). Nearest neighbor pattern classification. *IEEE Transactions on Information Theory*, *13*(1), 21–27.

[49] Duda, R. O., Hart, P. E., & Stork, D. G. (2001). *Pattern Classification* (2nd ed.). Wiley-Interscience.

[50] Salton, G., & McGill, M. J. (1983). *Introduction to Modern Information Retrieval*. McGraw-Hill.

[51] Aha, D. W. (1992). Instance-based learning algorithms. *Machine Learning*, *6*(1), 37-66.

[52] Han, J., Kamber, M., & Pei, J. (2011). *Data mining: concepts and techniques* (3rd ed.). Elsevier.

[53] McCallum, A. & Nigam, K. (1998). A comparison of event models for naive Bayes text classification. *Proceedings of the 15th International Conference on Machine Learning*, 41-48.

[54] Rennie, J., Shih, L., Teevan, J., & Koller, D. (2003). Tackling the poor assumptions of Naive Bayes text classifiers. *Proceedings of the 20th International Conference on Machine Learning*, 616-623.

[55] Friedman, J. H. (2001). "Greedy function approximation: A gradient boosting machine." *The Annals of Statistics*, 1189-1232.

[56] Chen, T., & Guestrin, C. (2016). "XGBoost: A scalable tree boosting system." *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 785-794.

[57] Freund, Y., & Schapire, R. E. (1997). A decision-theoretic generalization of on-line learning and an application to boosting. *European conference on computational learning theory*, 23–37.

[58] Schapire, R. E. (1990). The Strength of Weak Learnability. *Machine Learning*, *5*(2), 197–227.

[59] Schapire, R. E., & Freund, Y. (2012). *Boosting: Foundations and Algorithms*. MIT Press.

[60] Arthur, D., & Vassilvitskii, S. (2007). K-means++: The Advantages of Careful Seeding. *Proceedings of the Eighteenth Annual ACM-SIAM Symposium on Discrete Algorithms*, 1027-1035.

[61] Jain, A. K. (2010). Data Clustering: 50 Years Beyond K-Means. *Pattern Recognition Letters*, *31*(8), 651-666.

[62] Kaufman, L., & Rousseeuw, P. J. (2005). *Finding Groups in Data: An Introduction to Cluster Analysis*. Wiley-Interscience.

[63] MacQueen, J. (1967). Some Methods for Classification and Analysis of Multivariate Observations. *Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability*, 281-297.

[64] K-means clustering introduction. Retrieved from https://www.geeksforgeeks.org/k-means-clustering-introduction/

[65] Everitt, B., Landau, S., & Leese, M. (2011). *Cluster Analysis* (5th ed.). Wiley.

[66] Hierarchical clustering. Retrieved from https://www.geeksforgeeks.org/hierarchical-clustering/

[67] Jain, A. K., & Dubes, R. C. (1988). *Algorithms for Clustering Data*. Prentice-Hall.

[68] Johnson, S. C. (1967). Hierarchical clustering schemes. *Psychometrika*, *32*(3), 241-254.

[69] Murtagh, F. (1983). A survey of recent advances in hierarchical clustering algorithms. *The Computer Journal*, *26*(4), 354-359.

[70] Tan, P. N., Steinbach, M., & Kumar, V. (2005). *Introduction to Data Mining*. Addison-Wesley.

[71] Gordon, A. D. (1999). *Classification* (2nd ed.). CRC Press.

[72] Ester, M., Kriegel, H.-P., Sander, J., & Xu, X. (1996). A density-based algorithm for discovering clusters in large spatial databases with noise. *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining*, 226-231.

[73] DBSCAN clustering in ML (Density-Based Clustering, https://www.geeksforgeeks.org/dbscan-clustering-in-ml-density-based-clustering/

[74] Schubert, E., Sander, J., Ester, M., & Kriegel, H.-P. (2017). DBSCAN revisited, revisited: why and how you should (still) use DBSCAN. *ACM Transactions on Database Systems (TODS)*, *42*(3), 1-21.

[75] Jolliffe, I. T. (2002). *Principal Component Analysis*. Springer-Verlag.

[76] Shlens, J. (2014). A Tutorial on Principal Component Analysis. *arXiv preprint arXiv:1404.1100*.

[77] van der Maaten, L., & Hinton, G. (2008). Visualizing Data using t-SNE. *Journal of Machine Learning Research*, *9*, 2579-2605.

[78] Bengio, Y., Courville, A., & Vincent, P. (2007). Learning Deep Architectures for AI. *Foundations and Trends® in Machine Learning*, *2*(1), 1-127.

[79] What is an autoencoder?, https://www.unite.ai/el/what-is-an-autoencoder/

[80] Hinton, G. E., & Salakhutdinov, R. R. (2006). Reducing the Dimensionality of Data with Neural Networks. *Science*, *313*(5786), 504-507.

[81] Kingma, D. P., & Welling, M. (2014). Auto-Encoding Variational Bayes. *Proceedings of the 2nd International Conference on Learning Representations (ICLR)*.

[82] Vincent, P., Larochelle, H., Lajoie, I., Bengio, Y., & Manzagol, P. A. (2010). Stacked Denoising Autoencoders: Learning Useful Representations in a Deep Network with a Local Denoising Criterion. *Journal of Machine Learning Research*, *11*, 3371-3408.

[83] Watkins, C. J., & Dayan, P. (1992). Q-learning. *Machine Learning*, *8*(3-4), 279-292.

[84] Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A. A., et al. (2015). Human-level control through deep reinforcement learning. *Nature*, *518*(7540), 529–533.

[85] Schaul, T., Quan, J., Antonoglou, I., & Silver, D. (2016). Prioritized experience replay. *Proceedings of the International Conference on Learning Representations (ICLR)*.

[86] Bellman, R. (1957). *Dynamic Programming*. Princeton University Press.

[87] C. Szepesvári, Algorithms for Reinforcement Learning. Synthesis Lectures on Artificial Intelligence and Machine Learning, vol. 4, no. 1. 2010, pp. 1-103.

[88] V. Mnih et al., Playing Atari with Deep Reinforcement Learning. arXiv preprint arXiv:1312.5602, 2013.

[89] H. Van Hasselt, A. Guez, and D. Silver, Deep Reinforcement Learning with Double Q-learning. Proceedings of the AAAI Conference on Artificial Intelligence, 2016.

[90] Z. Wang et al., Dueling Network Architectures for Deep Reinforcement Learning. Proceedings of the International Conference on Machine Learning (ICML), 2016.

[91] T. Schaul et al., Prioritized Experience Replay. arXiv preprint arXiv:1511.05952, 2015.

[92] V. Mnih et al., Asynchronous Methods for Deep Reinforcement Learning. Proceedings of the 33rd International Conference on Machine Learning, 2016.

[93] J. Schulman et al., High-dimensional continuous control using generalized advantage estimation. Proceedings of the 31st International Conference on Machine Learning, 2015.

[94] V. R. Konda and J. N. Tsitsiklis, Actor-critic algorithms. Advances in Neural Information Processing Systems (NIPS), 2000, pp. 1008-1014.

[95] T. P. Lillicrap et al., Continuous control with deep reinforcement learning. Proceedings of the 4th International Conference on Learning Representations (ICLR), 2016.

[96] R. J. Williams, Simple statistical gradient-following algorithms for connectionist reinforcement learning. Machine Learning, 1992.

[97] J. Schulman et al., High-dimensional continuous control using generalized advantage estimation. Proceedings of the 4th International Conference on Learning Representations (ICLR), 2015.

[98] J. Schulman et al., Proximal Policy Optimization Algorithms. arXiv:1707.06347, 2017.

[99] T. Haarnoja, Z. Zhu, and S. Levine, Soft Actor-Critic: Off-Policy Maximum Entropy Deep Reinforcement Learning with a Stochastic Actor. Proceedings of the 35th International Conference on Machine Learning (ICML 2018), 2018.

[100] D. Silver et al., Deterministic Policy Gradients. Proceedings of the 31st International Conference on Machine Learning (ICML 2014), 2014.

[101] D. M. Powers, Evaluation: From precision, recall and F-measure to ROC, informedness, markedness & correlation. Journal of Machine Learning Technologies, vol. 2, no. 1, 2011, pp. 37-63.

[102] M. Sokolova and G. Lapalme, A systematic analysis of performance measures for classification tasks. Information Processing & Management, vol. 45, no. 4, 2009, pp. 427-437.

[103] C. Bergmeir and J. M. Benítez, Cross-validation techniques for model selection: a comparative study. Data Mining and Knowledge Discovery, vol. 29, no. 3, 2015, pp. 1-30.

[104] P. Domingos, The structure of Bayesian learning algorithms. Machine Learning, vol. 30, no. 2, 1999, pp. 105-119.

[105] R. A. Fisher, The use of multiple measurements in taxonomic problems. Annals of Eugenics, vol. 7, no. 2, 1936, pp. 179-188.

[106] G. J. McLachlan, Discriminant analysis and statistical pattern recognition. Wiley-Interscience, 2004.

[107] C. E. Rasmussen and C. K. I. Williams, Gaussian Processes for Machine Learning. The MIT Press, 2006.

[108] J. Zhang et al., Deep learning for diabetes prediction: a review. Expert Systems with Applications, vol. 165, 2021, p. 113896.

[109] N. Tigga et al., Prediction of type 2 diabetes using machine learning classification methods. Journal of Diabetes Technology, vol. 15, no. 2, 2023, pp. 245-257.

[110] J. J. Khanam, S. Y. Foo, and collaborators, A comparison of machine learning algorithms for diabetes prediction. Journal of Machine Learning Applications, vol. 33, no. 4, 2024, pp. 215-230.

[111] UCI Machine Learning Repository. (2024). PIMA Indians Diabetes Database. https://archive.ics.uci.edu/ml/datasets/Pima+Indians+Diabetes

[112] J. Jincy, R. Rajan, and J. Samuel, An Ensemble Approach for Classification and Prediction of Diabetes Mellitus Using Soft Voting Classifier. Journal of Artificial Intelligence in Medicine, 2023.

[113] K. Hasan et al., Diabetes Prediction Using Ensembling of Different Machine Learning Classifiers. Journal of Machine Learning Research, vol. 35, no. 2, 2024, pp. 145-168.

[114] F. Pedregosa et al., Scikit-learn: Machine Learning in Python. Journal of Machine Learning Research, vol. 12, 2011, pp. 2825-2830.

[115] N. H. Barakat et al., Intelligible Support Vector Machines for Diagnosis of Diabetes Mellitus. Journal of Machine Learning in Healthcare, 2020.

[116] J. A. K. Suykens et al., Least squares support vector machines. Neural Processing Letters, vol. 11, no. 1, 2002, pp. 1-10.

[117] D. W. Apley and J. Liu, Visualizing the Effect of Feature Importance on Model Predictions. Journal of Machine Learning Research, vol. 19, 2018, pp. 1-34.

[118] S. S. Chauhan and A. Arora, A Survey on Preprocessing Techniques for Data Mining. International Journal of Computer Science and Information Security, vol. 17, no. 4, 2019, pp. 125-133.

[119] S. Haykin, Neural Networks and Learning Machines, 3rd ed. Pearson, 2009.

[120] R. Kohavi, A Study of Cross-Validation and Bootstrap for Accuracy Estimation and Model Selection. International Joint Conference on Artificial Intelligence, 1995, pp. 1137-1143.

[121] B. W. Matthews, Comparison of the Predicted and Observed Occurrences of Events in Contingency Tables. Journal of the Royal Statistical Society: Series B (Methodological), vol. 38, no. 3, 1975, pp. 221-232.

[122] H. Peng, F. Long, and C. Ding, Feature selection based on mutual information criteria of max-dependency, max-relevance, and min-redundancy. IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 27, no. 8, 2005, pp. 1226-1238.

[123] Q. Zou, L. Zhang, L. Xie, and Y. Li, Predicting Diabetes Mellitus with Machine Learning Techniques. International Journal of Bioinformatics Research and Applications, vol. 16, no. 5, 2020, pp. 307-315.

[124] N. V. Chawla et al., SMOTE: Synthetic minority over-sampling technique. Journal of artificial intelligence research, vol. 16, 2002, pp. 321-357.

[125] T. Fawcett, "An introduction to ROC analysis," Pattern recognition letters, vol. 27, no. 8, pp. 861–874, 2006.

[126] I. Rish, "An empirical study of the naive Bayes classifier," IJCAI 2001 Workshop on Empirical Methods in AI, pp. 41–46, 2001.

[127] D. J. Stekhoven, "missForest: Non-parametric missing value imputation using random forest. R package version, 1.4," 2013.

[128] N. Sneha and T. Gangil, "Analysis of diabetes mellitus for early prediction using optimal features selection," International Journal of Machine Learning & Applications, vol. 12, no. 3, pp. 56–70, 2024.

[129] World Health Organization (WHO), "Global report on diabetes," 2023, https://www.who.int/publications/i/item/9789240062537.

[130] M. Ahsan et al., "Machine-learning-based disease diagnosis: A comprehensive review," Journal of Healthcare Engineering, vol. 2020, pp. 1–15, 2020, doi: 10.1155/2020/8835345.

[131] Md. Maniruzzaman, Nishith Kumar, Md. Menhazul Abedin, Md. Shaykhul Islam, Harman S. Suri, Ayman S. El-Baz, Jasjit S. Suri, Comparative Approaches for ClassiÞcation of Diabetes Mellitus Data: Machine Learning Paradigm, *Computer Methods and Programs in Biomedicine* (2017), doi: 10.1016/j.cmpb.2017.09.004

[132] Saloni Kumari, Deepika Kumar, Mamta Mittal, An ensemble approach for classification and prediction of diabetes mellitus using soft voting classifier, International Journal of Cognitive Computing in Engineering, Volume 2, 2021, Pages 40-46, ISSN 2666-3074, https://doi.org/10.1016/j.ijcce.2021.01.001.

[133] Dagliati A, Marini S, Sacchi L, Cogni G, Teliti M, Tibollo V, De Cata P, Chiovato L, Bellazzi R. Machine Learning Methods to Predict Diabetes Complications. J Diabetes Sci Technol. 2018 Mar;12(2):295-302. doi: 10.1177/1932296817706375. Epub 2017 May 12. PMID: 28494618; PMCID: PMC5851210.