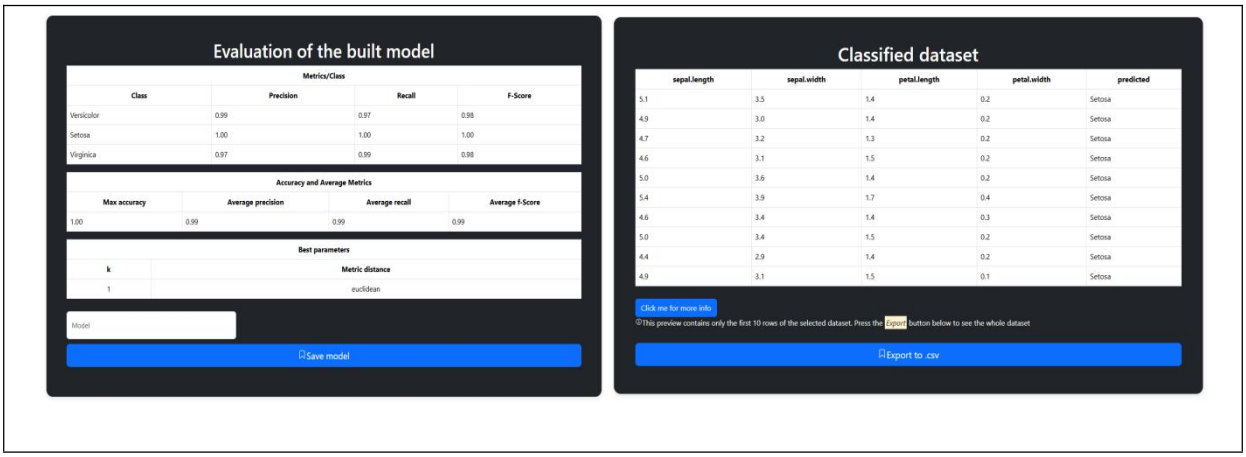
ΔΙΕΘΝΕΣ
ΠΑΝΕΠΙΣΤΗΜΙΟ
ΤΗΣ ΕΛΛΑΔΟΣ

ΣΧΟΛΗ ΜΗΧΑΝΙΚΩΝ

ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΗΛΕΚΤΡΟΝΙΚΩΝ ΣΥΣΤΗΜΑΤΩΝ

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

«Εφαρμογή ιστού για αυτοματοποιημένη κατηγοριοποίηση εγγύτερων γειτόνων»



Του φοιτητή
Κωνσταντίνου Κυριάκου Μπάτσιου
Αρ. Μητρώου: 2019110

Επιβλέπων
Στέφανος Ουγιάρογλου
Επίκουρος Καθηγητής

Ημερομηνία 31/05/2025

Τίτλος Δ.Ε. Εφαρμογή ιστού για αυτοματοποιημένη κατηγοριοποίηση εγγύτερων γειτόνων
Κωδικός Δ.Ε. 24119
Ονοματεπώνυμο φοιτητή/τών Κωνσταντίνος Κυριάκος Μπάτσιος
Ονοματεπώνυμο εισηγητή Στέφανος Ουγιάρογλου
Ημερομηνία ανάληψης Δ.Ε. 10-12-2024
Ημερομηνία περάτωσης Δ.Ε. 31-05-2025

Βεβαιώνω ότι είμαι ο συγγραφέας αυτής της εργασίας και ότι κάθε βοήθεια την οποία είχα για την προετοιμασία της είναι πλήρως αναγνωρισμένη και αναφέρεται στην εργασία. Επίσης, έχω καταγράψει τις όποιες πηγές από τις οποίες έκανα χρήση δεδομένων, ιδεών, εικόνων και κειμένου, είτε αυτές αναφέρονται ακριβώς είτε παραφρασμένες. Επιπλέον, βεβαιώνω ότι αυτή η εργασία προετοιμάστηκε από εμένα προσωπικά, ειδικά ως διπλωματική εργασία, στο Τμήμα Μηχανικών Πληροφορικής και Ηλεκτρονικών Συστημάτων του ΔΙ.ΠΑ.Ε.

Η έγκριση της διπλωματικής εργασίας από το Τμήμα Μηχανικών Πληροφορικής και Ηλεκτρονικών Συστημάτων του Διεθνούς Πανεπιστημίου της Ελλάδος, δεν υποδηλώνει απαραιτήτως και αποδοχή των απόψεων του συγγραφέα, εκ μέρους του Τμήματος.

Πρόλογος

Η εξέλιξη της Τεχνητής Νοημοσύνης και της Μηχανικής Μάθησης έχει οδηγήσει στη δημιουργία πολλών αλγορίθμων κατηγοριοποίησης, με τον k-Nearest Neighbors (k-NN) να αποτελεί έναν από τους πιο γνωστούς και ευρέως χρησιμοποιούμενους. Παρά την απλότητα και την αποτελεσματικότητά του, η εφαρμογή του k-NN απαιτεί συχνά τεχνικές γνώσεις και εγκατάσταση πολύπλοκων εργαλείων, κάτι που καθιστά την πρόσβαση σε αυτόν περιορισμένη για μη ειδικούς χρήστες.

Η παρούσα διπλωματική εργασία στοχεύει στην επίλυση των παραπάνω ζητημάτων, μέσω της ανάπτυξης της διαδικτυακής εφαρμογής AutoKNN. Το AutoKNN προσφέρει μια φιλική και διαδραστική πλατφόρμα, που επιτρέπει τη δημιουργία και αξιολόγηση μοντέλων k-NN χωρίς την ανάγκη για εξειδικευμένο υπόβαθρο. Παράλληλα, παρέχει τη δυνατότητα αποθήκευσης μοντέλων και επαναχρησιμοποίησής τους για μελλοντικές προβλέψεις.

Η εργασία αυτή απευθύνεται σε ερευνητές, φοιτητές, προγραμματιστές, αλλά και σε κάθε ενδιαφερόμενο που επιθυμεί να κατανοήσει και να αξιοποιήσει τον αλγόριθμο k-NN με ευκολία και προσβασιμότητα, μέσω μιας πλήρως ανοιχτής και επεκτάσιμης διαδικτυακής λύσης.

Περίληψη

Η κατηγοριοποίηση (classification) των δεδομένων ορίζεται ως η διαδικασία κατάταξης στοιχείων ενός συνόλου δεδομένων σε ομάδες ή αλλιώς κατηγορίες, βάσει χαρακτηριστικών τους. Για τη διαδικασία της κατηγοριοποίησης, έχουν δημιουργηθεί αρκετοί σχετικοί αλγόριθμοι, ένας εκ των οποίον είναι ο κατηγοριοποιητής k εγγύτερων γειτόνων (k Nearest Neighbors - k-NN). Ο k-NN είναι ένας από τους πιο γνωστούς κατηγοριοποιητές στο χώρο της Εξόρυξης Πληροφορίας, λόγω της εύκολης δομής του και στον απλοϊκό τρόπο λειτουργικότητας του, πράγμα που έχει συμβάλλει στην αξιοποίησή του από πολλές εφαρμογές. Ύστερα από σχετική έρευνα, συνειδητοποιήθηκε το γεγονός ότι υπάρχει δυσκολία χρήσης του k-NN, καθώς οι χρήστες απαιτούται να κατέχουν γνώσεις Μηχανικής Μάθησης και Προγραμματισμού, συμπεραίνοντας στην εύρεση λύσεων λογισμικού. Ωστόσο, παρατηρείται έλλειμα στις λύσεις κι όσες υπάρχουν παρουσιάζουν σημαντικά αρνητικά σημεία και περιορισμούς. Πολλές από αυτές απαιτούν κόστος αγοράς συνδρομής κι αναγκαία εγκατάσταση λογισμικών και βιβλιοθηκών για πλήρη αξιοποίηση, κάνοντας ακόμα πιο δύσκολη τη χρήση του κι υπάρχει περιττή κατανάλωση πόρων από τον υπολογιστή του χρήστη. Αυτή η διπλωματική εργασία αποσκοπεί στην καταπολέμιση των προαναφερόμενων ζητημάτων, μέσω ανάπτυξης μιας διαδικτυακής εφαρμογής εν ονόνατι "AutoKNN", η οποία επιτρέπει σε όλους τους ενδιαφερόμενους χρήστες να επιλέξει επιθυμητά σύνολα δεδομένων και κατάλληλες παραμέτρους του κατηγοριοποιητή, με σκοπό τη δημιουργία μοντέλων k-NN. Μετέπειτα, η αξιολόγηση των αποτελεσμάτων του μοντέλου πραγματοποιείται με τη μέθοδο train test split, όπου και παρουσιάζονται αυτά τα αποτελέσματα μετρικών απόδοσης. Ο χρήστης μπορεί επίσης να αποθηκεύσει αυτό το προ-εκπαιδευμένο μοντέλο, εφόσον το επιθυμεί, και να το αξιοποιήσει μελλοντικά για πρόβλεψη μη κατηγοριοποιημένων στιγμιοτύπων. Το AutoKNN μέσω φιλικού GUI (Graphical User Interface), δίνει την ικανότητα στον χρήστη να χρησιμοποιήσει ελεύθερα κι εύκολα τον αλγόριθμο. Επίσης, μαζί με το GUI, είναι διαθέσιμο στους χρήστες και το διαδικτυακό (Web) API (Application Programming Interface) για την αξιοποίηση του AutoKNN κι από ενδιαφερόμενους προγραμματιστές.

«Web application for automated nearest neighbor classification»

«Konstantinos Kyriakos Batsios»

Abstract

Data classification is defined as the process of classifying elements of a dataset into groups or categories based on their characteristics. For the classification process, several related algorithms have been developed, one of which is the k Nearest Neighbors (k-NN) classifier. The k-NN is one of the most well-known classifiers in the field of Information Mining, due to its easy structure and its simplistic way of functionality, which has contributed to its use by many applications. After some research, it was realized that there is a difficulty in using k-NN, as users are required to have knowledge of Machine Learning and Programming, concluding in finding software solutions. However, there is a lack of solutions and those that do exist have significant negative points and limitations. Many of them require a subscription cost and necessary installation of software and libraries for full use, making it even more difficult to use and there is unnecessary consumption of resources from the user's computer. This thesis aims to combat the mentioned issues by developing a web application called "AutoKNN", which allows all interested users to select desired datasets and appropriate classifier parameters in order to create k-NN models. Subsequently, the evaluation of the model results is performed by the train test split method, where these performance metric results are presented. The user can also save this pre-trained model, if desired, and use it in the future to predict unclassified snapshots. AutoKNN through a friendly GUI (Graphical User Interface), gives the user the ability to freely and easily use the algorithm. Also, along with the GUI, the Web API (Application Programming Interface) is available to users for the use of AutoKNN by interested developers.

Ευχαριστίες

Το έργο αυτό είναι αφιερωμένο στην οικογένειά μου, της οποίας η ακλόνητη πίστη στις δυνατότητές μου υπήρξε ο πυλώνας στον οποίο στηρίχθηκα στις πιο δύσκολες στιγμές. Η συνεχής ενθάρρυνση και η αγάπη σας ήταν πηγή δύναμης και κινήτρων για μένα. Η παρούσα εργασία αφιερώνεται επίσης στον Επίκουρο Καθηγητή, κ. Στέφανο Ουγιαρόγλου, του οποίου η καθοδήγηση και η υποστήριξη ήταν ανεκτίμητες για την ολοκλήρωση της παρούσας μελέτης. Η διορατική κριτική τους, η υπομονετική ενθάρρυνση και τα αυστηρά ακαδημαϊκά πρότυπα έχουν θέσει ένα σημείο αναφοράς που θα προσπαθήσω να εκπληρώσω σε όλη μου την καριέρα.

Περιεχόμενα

Κατάλογος Σχημάτων

Κατάλογος Πινάκων

Κεφάλαιο 1ο: Εισαγωγή

1.1 Κατηγοριοποίηση δεδομένων

Η κατηγοριοποίηση (classification) αποτελεί μια βασική εργασία της Μηχανικής Μάθησης που περιλαμβάνει την ταξινόμηση δεδομένων σε καθορισμένες κατηγορίες, διευκολύνοντας έτσι την αποτελεσματική λήψη αποφάσεων και προβλέψεων. Περιλαμβάνει δύο κύριες μεθόδους: την ταξινόμηση με επίβλεψη (supervised classification), η οποία βασίζεται σε επισημασμένα δεδομένα για να κατευθύνει τη διαδικασία μάθησης, και τη ταξινόμηση χωρίς επίβλεψη (unsupervised classification), που ανακαλύπτει μοτίβα σε δεδομένα χωρίς ετικέτες (labels). Μέθοδοι με επίβλεψη, όπως τα Δέντρα Αποφάσεων (Decision Trees - DT) και οι Μηχανές Διανυσμάτων Υποστήριξης (Support Vector Machines - SVM), έχουν αποδειχθεί αποτελεσματικές στη διαχείριση πολύπλοκων συνόλων δεδομένων, παρέχοντας υψηλή ακρίβεια και αξιοπιστία για εφαρμογές στον πραγματικό κόσμο [1]. Τα προ-εκπαιδευμένα μοντέλα, τα οποία αναπτύσσονται με εκπαίδευση σε εκτεταμένα σύνολα δεδομένων για την εκμάθηση ευρύτερων προτύπων και γνώσεων, προσφέρουν μια σταθερή βάση για συγκεκριμένες εργασίες μέσω της τελειοποίησης σε μικρότερα σύνολα δεδομένων, προσανατολισμένα σε εργασίες. Αυτή η προσέγγιση συντομεύει τη διάρκεια της εκπαίδευσης, βελτιώνει την απόδοση και μειώνει την εξάρτηση από μεγάλες ποσότητες δεδομένων με ετικέτες, καθιστώντας τα προ-εκπαιδευμένα μοντέλα απαραίτητα για τα σύγχρονα συστήματα ταξινόμησης [2].

Η κατηγοριοποίηση δεδομένων είναι εξαιρετικά σημαντική για τη μηχανική μάθηση, επειδή διευκολύνει την πρόβλεψη αποτελεσμάτων με την επισήμανση δεδομένων σύμφωνα με τα χαρακτηριστικά τους, κάτι που είναι σημαντικό για εργασίες που περιλαμβάνουν τόσο κατηγορικές όσο και αριθμητικές πληροφορίες. Οι εφαρμογές της καλύπτουν πολλούς τομείς, όπως της υγειονομικής περίθαλψης για τη διάγνωση ασθενειών, της χρηματοοικονομικής για τον εντοπισμό απάτης, της αναγνώριση εικόνας για τον εντοπισμό αντικειμένων ή προσώπων και της ανάλυσης κειμένου για την εκτίμηση του συναισθήματος, αποδεικνύοντας την προσαρμοστικότητά της στην αντιμετώπιση προκλήσεων του πραγματικού κόσμου και προωθώντας την πρόοδο σε όλους τους τομείς [3]. Όπως επισημαίνεται στο [1], η ταξινόμηση είναι ζωτικής σημασίας για τη διάρθρωση πολύπλοκων συνόλων δεδομένων σε κατηγορίες, διευκολύνοντας την αποτελεσματική λήψη αποφάσεων και ενισχύοντας την ακρίβεια πρόβλεψης σε μεταβαλλόμενα περιβάλλοντα. Επιπλέον, τα προ-εκπαιδευμένα μοντέλα, όπως περιγράφονται στο [2], ενισχύουν την αποτελεσματικότητα της ταξινόμησης με τη αξιοποίηση της υπάρχουσας γνώσης, τη μείωση των υπολογιστικών απαιτήσεων και τη δυνατότητα προσαρμογής των εφαρμογών σε νέα σύνολα δεδομένων με μικρή επανεκπαίδευση, καθιστώντας τα έτσι έναν βασικό πόρο για τη σύγχρονη μηχανική μάθηση.

Στο παρακάτω Σχήμα 1.1 [4], απεικονίζει τις κύριες κατηγορίες της Μηχανικής Μάθησης: μάθηση με επίβλεψη (supervised learning) και μάθηση χωρίς επίβλεψη (unsupervised learning).

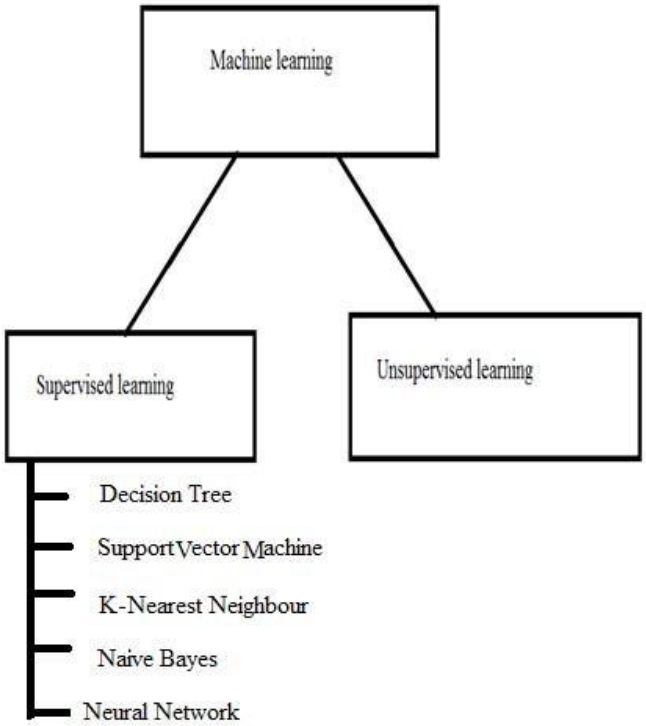


Figure 1-1 Επισκόπηση της μηχανικής μάθησης.

Η μάθηση με επίβλεψη, αποτελεί το θεμελιώδες στοιχείο για τις εργασίες κατηγοριοποίησης, χρησιμοποιώντας σύνολα δεδομένων με ετικέτες για τη συσχέτιση των γνωρισμάτων εισόδου με συγκεκριμένες κλάσεις ή αποτελέσματα. Αυτή η μέθοδος επιτρέπει σε αλγορίθμους με επίβλεψη, όπως οι προαναφερόμενοι (DT και SVM), οι Naive Bayes και οι k Εγγύτερων Γειτόνων (k-Nearest Neighbors - k-NN), να εντοπίζουν συγκεκριμένες συνδέσεις μεταξύ των δεδομένων εισόδου και των αποτελεσμάτων στόχου, διευκολύνοντας την ακριβή κατηγοριοποίηση ή παλινδρόμηση (regression) [1], [3]. Ο κύριος στόχος της μάθησης με επίβλεψη είναι η ανάπτυξη ενός μοντέλου ικανού να προβλέπει με ακρίβεια τις εξόδους από άγνωστα δεδομένα, χρησιμοποιώντας πρότυπα που έχουν αποκτηθεί από δείγματα με ετικέτες. Οι αλγόριθμοι μάθησης με επίβλεψης, συμπεριλαμβανομένων των προαναφερόμενων (Δέντρα Αποφάσεων και Μηχανές Διανυσμάτων Υποστήριξης), αντιμετωπίζουν διάφορες προκλήσεις του πραγματικού κόσμου συνδέοντας χαρακτηριστικά εισόδου με καθορισμένες εξόδους, καθιστώντας τους σχετικούς σε τομείς όπως η ιατρική διάγνωση, η ανίχνευση απάτης και η ταξινόμηση κειμένου [4], [5].

Στο παρακάτω Σχήμα 1.2 [6], απεικονίζεται η διαδικασία μεθόδου μάθησης με επίβλεψη.

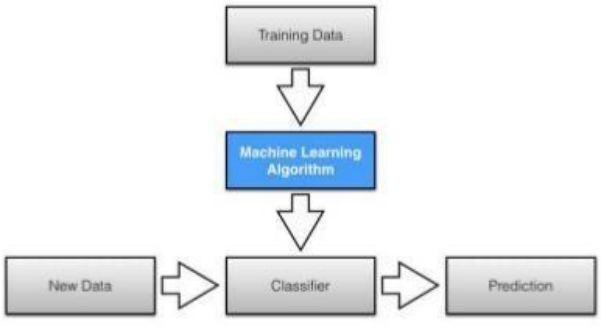


Figure 1-2 Διαδικασία μάθησης με επίβλεψη

Σε αντίθεση με την μάθηση με επίβλεψη, η οποία δίνει έμφαση σε εργασίες κατηγοριοποίησης και χρησιμοποιεί επισημασμένα δεδομένα για την εκπαίδευση αλγορίθμων για την πρόβλεψη συγκεκριμένων κλάσεων, η μάθηση χωρίς επίβλεψη λειτουργεί χωρίς την ανάγκη για επισημασμένα δεδομένα. Αποσκοπεί περισσότερο στην αποκάλυψη κρυφών μοτίβων ή πλαισίων που είναι παρόντα στα ίδια τα δεδομένα. Η μάθηση με επίβλεψη επιδιώκει να συνδέσει τα δεδομένα εισόδου με τις καθορισμένες εξόδους, ενώ η μάθηση χωρίς επίβλεψη διερευνά τη φυσική δομή των δεδομένων, χρησιμοποιώντας συχνά την συσταδοποίηση (clustering) ή την ανάλυση συσχετίσεων. Οι τεχνικές συσταδοποίησης, όπως η K-means, ταξινομούν τα σημεία δεδομένων σε συστάδες ανάλογα με τις ομοιότητές τους, με την K-means να είναι ιδιαίτερα αποτελεσματική για το διαχωρισμό των δεδομένων σε έναν προκαθορισμένο αριθμό συστάδων, που ορίζονται εκ των προτέρων, καθιστώντας την ιδιαίτερα επωφελής στην απουσία επισημασμένων δεδομένων [2]. Η μέθοδος αυτή εφαρμόζεται συχνά σε τομείς όπως τα συστήματα συστάσεων, όπου η συσταδοποίηση αποκαλύπτει μοτίβα στη συμπεριφορά των χρηστών για την πρόβλεψη αποφάσεων [3]. Σε αντίθεση με την κατηγοριοποίηση στην μάθηση με επίβλεψη, όπου οι έξοδοι καθορίζονται εκ των προτέρων, η μάθηση χωρίς επίβλεψη παράγει τις δικές της ετικέτες ή συστάδες, προσφέροντας βαθύτερη κατανόηση της υποκείμενης δομής των δεδομένων [6].

1.2 Εισαγωγή στη κατηγοριοποίηση εγγύτερων γειτόνων

Η κατηγοριοποίηση εγγύτερων γειτόνων είναι μια ευρέως διαδεδομένη τεχνική στη μηχανική μάθηση που χρησιμοποιεί την ομοιότητα μεταξύ σημείων δεδομένων για τη δημιουργία προβλέψεων. Αποτελεί κρίσιμη μέθοδο τόσο για τις εργασίες κατηγοριοποίησης όσο και για τις εργασίες παλινδρόμησης, οι οποίες είναι ζωτικής σημασίας σε πολυάριθμες εφαρμογές του πραγματικού κόσμου, όπως η αναγνώριση εικόνας και ομιλίας, καθώς και η ιατρική διάγνωση [7]. Μια ιδιαίτερη και ευρέως χρησιμοποιούμενη εφαρμογή αυτής της ιδέας είναι ο κατηγοριοποιητής k-NN, ο οποίος έχει αναδειχθεί σε ένα από τα πιο ισχυρά μέσα σε πολλούς τομείς.

Ο κατηγοριοποιητής k-NN είναι μια θεμελιώδης προσέγγιση στην επιβλεπόμενη μηχανική μάθηση, η οποία φημίζεται για την εύκολη της χρήση και την αποτελεσματικότητά της. Ως μη παραμετρική προσέγγιση, δεν προϋποθέτει μια συγκεκριμένη κατανομή για τα δεδομένα, γεγονός που της επιτρέπει να προσαρμόζεται σε διάφορες δραστηριότητες κατηγοριοποίησης και παλινδρόμησης. Ο αλγόριθμος k-NN, που ξεκίνησε το 1951, συνεχίζει να είναι ένας ευρέως μελετημένος και αξιοποιημένος αλγόριθμος σε τομείς όπως η αναγνώριση προτύπων, η ταξινόμηση κειμένων και η ανίχνευση αντικειμένων [8]. Η διαρκή ελκυστικότητά της προκύπτει από την απλή μέθοδο αξιοποίησης της εγγύτητας των δεδομένων για τη δημιουργία προβλέψεων.

Στο παρακάτω Σχήμα 1.3 [8], απεικονίζεται ένα παράδειγμα κατηγοριοποίησης με τον k-NN.

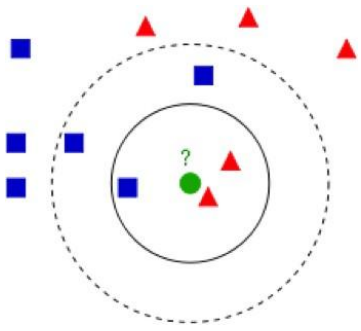


Figure 1-3 Παράδειγμα κατηγοριοποίησης k-NN με K γείτονες K = 3 (συνεχής κύκλος) και K = 5 (διακεκομμένος κύκλος), το μέτρο απόστασης είναι η Euclidean απόσταση.

Ενώ οι ταξινομητές εκμάθησης βάθους συνήθως υπερέχουν έναντι των παραδοσιακών μεθόδων σε εργασίες κατηγοριοποίησης, οι τυπικοί ταξινομητές, όπως ο k-NN, εξακολουθούν να έχουν μεγάλη βαρύτητα, ιδίως σε περιπτώσεις με μικρότερα σύνολα δεδομένων [9]. Η συνεχής σπουδαιότητά τους προκύπτει όχι μόνο από την αμεσότητά τους, αλλά και από την ικανότητά τους να προσφέρουν ιδέες που καθοδηγούν τη δημιουργία πιο περίπλοκων μοντέλων. Επιπλέον, η ικανότητα του k-NN να αντιμετωπίζει μη γραμμικά ζητήματα χωρίς προκαταλήψεις σχετικά με την κατανομή των δεδομένων ενισχύει τη συνεχή εφαρμογή του.

Ο k-NN κατηγοριοποιεί ένα δείγμα (sample) δοκιμής (test) σύμφωνα με την επικρατούσα κλάση (class) των k εγγύτερων γειτόνων του, η οποία βρίσκεται με τη μέτρηση των αποστάσεων μεταξύ του δείγματος δοκιμής και του συνόλου εκπαίδευσης (train). Το δείγμα δοκιμής κατατάσσεται στην κατηγορία που εμφανίζεται συχνότερα μεταξύ των πλησιέστερων γειτόνων του, όπου η τιμή k είναι ένας βασικός παράγοντας που επηρεάζει την ακρίβεια κατηγοριοποίησης. Η μετρική της απόστασης, όπως η Euclidean απόσταση, είναι κρίσιμη για τον προσδιορισμό αυτών των γειτόνων, επηρεάζοντας άμεσα την απόδοση του αλγορίθμου. Ο αλγόριθμος k-NN είναι απλός και πολύ αποδοτικός- ωστόσο, αντιμετωπίζει προβλήματα όσον αφορά την υπολογιστική αποδοτικότητα, ιδίως με μεγάλα σύνολα δεδομένων και χώρους χαρακτηριστικών υψηλών διαστάσεων, επειδή απαιτεί τον υπολογισμό αποστάσεων μεταξύ του δείγματος δοκιμής και κάθε δείγματος εκπαίδευσης [8]. Παρά τις δυσκολίες αυτές, το k-NN συνεχίζει να είναι μια αγαπημένη προσέγγιση λόγω της προσαρμοστικότητάς του, της απλής εφαρμογής του και της αποτελεσματικότητάς του σε διάφορους τομείς [9].

Ο k-NN βασίζεται σημαντικά στις παραμέτρους k (αριθμός γειτόνων) και τη μετρική της απόστασης για την αποτελεσματικότητά του. Η παράμετρος k υποδεικνύει πόσοι πλησιέστεροι γείτονες λαμβάνονται υπόψη κατά την κατηγοριοποίηση ενός νέου σημείου δεδομένων. Ένα μικρό K (όπως το 1) μπορεί να οδηγήσει σε υπερπροσαρμογή, με αποτέλεσμα το μοντέλο να αντιδρά υπερβολικά ευαίσθητα στο θόρυβο, ενώ ένα μεγάλο k μπορεί να εξομαλύνει υπερβολικά το όριο απόφασης, οδηγώντας σε υποπροσαρμογή [7]. Η αύξηση της τιμής k είναι απαραίτητη για την ενίσχυση της απόδοσης του μοντέλου, χρησιμοποιώντας διάφορες τεχνικές όπως η διασταυρούμενη επικύρωση (cross-validation) για τον προσδιορισμό της βέλτιστης τιμής [10].

Εκτός από το k, σημαντική είναι και η μετρική απόσταση, η οποία αξιολογεί την ομοιότητα μεταξύ των σημείων δεδομένων. Οι τυπικές μετρικές αποστάσεων περιλαμβάνουν τις αποστάσεις Euclidean, Manhattan, Chebyshev και Minkowski, με την καθεμία να παρουσιάζει διαφορετικό βαθμό αποτελεσματικότητας ανάλογα με το σύνολο δεδομένων και την περιοχή του προβλήματος [8], [9]. Η επιλογή της κατάλληλης μετρικής μπορεί να επηρεάσει σημαντικά την απόδοση του k-NN, επομένως η βελτιστοποίηση τόσο του k όσο και της μετρικής απόστασης είναι απαραίτητη για την επίτευξη ακριβής και αποτελεσματικής κατηγοριοποίησης.

1.3 Αυτοματοποιημένη Μηχανική Μάθηση

Η Αυτοματοποιημένη Μηχανική Μάθηση (Automated Machine Learning - AutoML) ορίζει την αυτοματοποίηση όλων των δραστηριοτήτων που είναι απαραίτητες για τη δημιουργία μοντέλων μηχανικής μάθησης. Αυτές οι δραστηριότητες περιλαμβάνουν συνήθως την προεπεξεργασία δεδομένων, την ανάπτυξη χαρακτηριστικών, την επιλογή μοντέλου, τη βελτιστοποίηση υπερπαραμέτρων και την αξιολόγηση [11]. Το AutoML καταργεί την ανάγκη άμεσης συμμετοχής και γνώσεων εμπειρογνωμόνων, δίνοντας τη δυνατότητα ακόμη και σε όσους δεν διαθέτουν εμπειρία να δημιουργήσουν αποτελεσματικά λύσεις μηχανικής μάθησης [12]. Ο κύριος στόχος του AutoML είναι να απλοποιήσει τη ροή εργασίας της μηχανικής μάθησης, ενισχύοντας την προσβασιμότητα και

διασφαλίζοντας ότι τα παραγόμενα μοντέλα αποδίδουν στο μέγιστο δυνατό βαθμό [13]. Με την αυτοματοποίηση αυτών των διαδικασιών, τα συστήματα AutoML επιδιώκουν να μειώσουν τόσο τα υπολογιστικά έξοδα όσο και την ανθρώπινη εργασία, ενισχύοντας παράλληλα την ακρίβεια της πρόβλεψης.

Το AutoML λειτουργεί μέσω ενός πλαισίου βελτιστοποίησης δύο επιπέδων, το οποίο αναλύει τις διαμορφώσεις για τις σωληνώσεις μηχανικής μάθησης σε μικρότερα στοιχεία και τα συνδυάζει επανειλημμένα για να ανακαλύψει τις βέλτιστες λύσεις [12], [13]. Αυτή η επαναλαμβανόμενη μέθοδος εγγυάται ότι προσδιορίζονται οι καλύτερες ρυθμίσεις για κάθε εργασία, ενώ παράλληλα τηρούνται οι υπολογιστικοί περιορισμοί [14]. Η ιδέα του AutoML προέκυψε από την αυξανόμενη ζήτηση για κλιμακούμενη, αποτελεσματική και προσιτή μηχανική μάθηση, αντιμετωπίζοντας τις δυσκολίες που σχετίζονται με τις συμβατικές μεθόδους μηχανικής μάθησης.

Το AutoML χρησιμοποιείται σε διάφορους τομείς επειδή αυτοματοποιεί και βελτιώνει τις εργασίες μηχανικής μάθησης, επιτρέποντας στους χρήστες να επικεντρωθούν στην αξιοποίηση των πληροφοριών αντί να χειρίζονται τις επιπλοκές της ανάπτυξης μοντέλων. Οι κύριες χρήσεις του AutoML είναι οι ακόλουθες:

- **Υγειονομική περίθαλψη:** Το AutoML είναι απαραίτητο στην υγειονομική περίθαλψη, καθώς απλοποιεί τη δημιουργία προγνωστικών μοντέλων για τη διάγνωση ασθενειών, τη διαμόρφωση προτιμήσεων για θεραπείες και την πρόβλεψη των αποτελεσμάτων των ασθενών. Για παράδειγμα, τα συστήματα AutoML μπορούν να προσδιορίσουν σημαντικά χαρακτηριστικά από σύνολα ιατρικών δεδομένων, συμπεριλαμβανομένων πληροφοριών απεικόνισης ή αρχείων ασθενών, και να δημιουργήσουν ισχυρά μοντέλα για εργασίες κατηγοριοποίησης ή παλινδρόμησης.
- **Χρηματοοικονομικά:** Στον χρηματοπιστωτικό κλάδο, το AutoML εφαρμόζεται εκτενώς για δραστηριότητες όπως η ανίχνευση απάτης, η βαθμολόγηση της πίστωσης και η αξιολόγηση του κινδύνου. Μέσω της αυτοματοποίησης του μηχανικού σχεδιασμού χαρακτηριστικών και της επιλογής μοντέλων, το AutoML επιτρέπει στους φορείς να ανιχνεύουν πιο αποτελεσματικά δόλιες συναλλαγές ή να προβλέπουν τους κινδύνους αθέτησης πελατών.
- **Λιανικό εμπόριο και Μάρκετινγκ:** Το AutoML βελτιώνει τις εξατομικευμένες συστάσεις, την πρόβλεψη της ζήτησης και την τμηματοποίηση των πελατών στο λιανικό εμπόριο. Αυτά τα συστήματα αυτοματοποιούν τη δημιουργία μοντέλων μηχανικής μάθησης που αναλύουν τις αγοραστικές συμπεριφορές, βελτιστοποιούν τα επίπεδα αποθεμάτων ή προβλέπουν τις προτιμήσεις των πελατών.
- **Μεταφοράς:** Στον τομέα των μεταφορών, το AutoML χρησιμοποιείται για τη βελτίωση του ελέγχου της κυκλοφορίας, του σχεδιασμού των διαδρομών των οχημάτων και των δραστηριοτήτων λογιστικής. Τα συστήματα AutoML που βασίζονται στην ενισχυτική μάθηση έχουν χρησιμοποιηθεί στον έλεγχο των σηματοδοτών κυκλοφορίας, προσαρμόζοντας το χρονοδιάγραμμα των σηματοδοτών σε πραγματικό χρόνο με βάση τα δεδομένα της κυκλοφορίας για την ανακούφιση της κυκλοφοριακής συμφόρησης.

Η ευρεία εφαρμογή του AutoML οφείλεται στην ικανότητά του να προσαρμόζεται σε διαφορετικά σύνολα δεδομένων και εργασίες, παρέχοντας κλιμακούμενες και αποτελεσματικές λύσεις μηχανικής μάθησης. Αυτή η ευελιξία είναι ιδιαίτερα κρίσιμη σε βιομηχανίες όπου απαιτούνται άμεσες αποφάσεις ή σημαντικές προβλέψεις.

Η διαδικασία του AutoML αποτελείται από πολλαπλά κρίσιμα στάδια, καθένα από τα οποία επικεντρώνεται σε ένα ξεχωριστό στοιχείο της ροής εργασίας της μηχανικής μάθησης [11], [13]. Αυτά τα στάδια διευκολύνουν την αυτοματοποίηση δραστηριοτήτων που κανονικά θα απαιτούσαν σημαντική χειρωνακτική εργασία και δεξιότητες, επιτρέποντας την αποτελεσματική και επεκτάσιμη δημιουργία μοντέλων [12], [14]. Τα στάδια διαμορφώνονται ως εξής:

- **Προεπεξεργασία δεδομένων:** Η προεπεξεργασία των δεδομένων αποτελεί το βασικό στάδιο της AutoML. Περιλαμβάνει τη συλλογή συναφών συνόλων δεδομένων, τον εξευγενισμό των δεδομένων με την εξάλειψη των ασυνεπειών και του θορύβου και τη βελτίωση του συνόλου δεδομένων για την ενίσχυση της ανθεκτικότητας του μοντέλου.
- **Μηχανικός σχεδιασμός χαρακτηριστικών:** Το στάδιο αυτό μετατρέπει τα ακατέργαστα δεδομένα σε πολύτιμες εισόδους για τα μοντέλα. Περιλαμβάνει την επιλογή κατάλληλων χαρακτηριστικών, τη δημιουργία νέων και την απόκτηση κρίσιμων πληροφοριών μέσω μεθόδων όπως η PCA (Principal Component Analysis).
- **Επιλογή μοντέλου:** Η επιλογή μοντέλου αυτοματοποιεί την επιλογή του καλύτερου αλγορίθμου για την εργασία. Αξιολογεί αλγορίθμους όπως k-NN, SVM και μοντέλα βαθιάς μάθησης από προκαθορισμένους χώρους αναζήτησης για τον εντοπισμό του πιο αποτελεσματικού.
- **Ρύθμιση υπερπαραμέτρων:** Η βελτιστοποίηση υπερπαραμέτρων βελτιώνει την αναζήτηση των βέλτιστων διαμορφώσεων του μοντέλου χρησιμοποιώντας μεθόδους όπως η βελτιστοποίηση κατά Bayes, η τυχαία αναζήτηση και η αναζήτηση πλέγματος.
- **Αξιολόγηση μοντέλου:** Το τελευταίο βήμα περιλαμβάνει την αξιολόγηση της απόδοσης του μοντέλου μέσω επικύρωσης και διασταυρούμενης επικύρωσης για να επιβεβαιωθεί η γενικευσιμότητα και η αξιοπιστία του σε διάφορες κατανομές δεδομένων.

Το AutoML παρέχει μια πληθώρα θετικών χαρακτηριστικών, καθιστώντας το βασικό πόρο στους σύγχρονους τομείς που είναι προσανατολισμένοι στα δεδομένα [11], [12]. Παρόλα αυτά, όπως κάθε τεχνολογική εξέλιξη, έχει και αυτή τα μειονεκτήματά της [13], [14]. Ακολουθούν τα κύρια πλεονεκτήματα και μειονεκτήματα του AutoML που προκύπτουν από τις αναφερόμενες πηγές.

Πλεονεκτήματα

1. **Προσβασιμότητα:** Το AutoML διευκολύνει τη μηχανική μάθηση μειώνοντας την ανάγκη για γνώση ειδικών, επιτρέποντας σε χρήστες με ελάχιστη τεχνική εμπειρία να αναπτύξουν και να χρησιμοποιήσουν αποτελεσματικά μοντέλα.
2. **Αποδοτικότητα:** Η αυτοματοποίηση δραστηριοτήτων όπως η προεπεξεργασία δεδομένων, ο μηχανικός σχεδιασμός των χαρακτηριστικών και η ρύθμιση των υπερπαραμέτρων μειώνει σημαντικά τον χρόνο που απαιτείται για τη δημιουργία μοντέλων μηχανικής μάθησης.
3. **Επεκτασιμότητα:** Η AutoML μπορεί να χειριστεί μεγάλα σύνολα δεδομένων και πολύπλοκες εργασίες, καθιστώντας την κατάλληλη για κλάδους με υψηλές απαιτήσεις σε υπολογισμούς και δεδομένα, όπως η υγειονομική περίθαλψη και η χρηματοοικονομική.
4. **Επίδοση:** Με τη διερεύνηση εκτεταμένων χώρων αναζήτησης αλγορίθμων και υπερπαραμέτρων, το AutoML επιτυγχάνει συχνά αξιοσημείωτη ακρίβεια και αξιοπιστία στις προβλέψεις του.
5. **Ενσωμάτωση:** Το AutoML λειτουργεί αποτελεσματικά παράλληλα με καθιερωμένα πλαίσια όπως το Google Cloud AutoML και το H2O.ai, επιτρέποντας την ομαλή υλοποίηση σε πρακτικές εφαρμογές.

Μειονεκτήματα

1. **Υπολογιστικά έξοδα:** Η αυτοματοποίηση της βελτιστοποίησης υπερπαραμέτρων, της επιλογής μοντέλου και άλλων εργασιών απαιτεί συνήθως σημαντική υπολογιστική ισχύ, ιδίως για εκτεταμένα σύνολα δεδομένων ή περίπλοκα μοντέλα βαθιάς μάθησης.
2. **Περιορισμένη αντίληψη:** Τα συστήματα AutoML λειτουργούν συχνά ως «μαύρα κουτιά», καθιστώντας δύσκολη την κατανόηση της λογικής πίσω από τις προβλέψεις τους. Το ζήτημα αυτό είναι ιδιαίτερα ανησυχητικό σε τομείς όπως η υγειονομική περίθαλψη, όπου η διαφάνεια είναι απαραίτητη.
3. **Κίνδυνος υπερπροσαρμογής:** Αν δεν υπάρχουν επαρκείς περιορισμοί, τα συστήματα AutoML μπορούν να προσαρμοστούν υπερβολικά στα δεδομένα εκπαίδευσης, περιορίζοντας την ικανότητά τους να προσαρμόζονται σε νέα σύνολα δεδομένων.

4. **Εξάρτηση από καθιερωμένους χώρους:** Η αποδοτικότητα των αποτελεσμάτων του AutoML εξαρτάται σημαντικά από τη διαμόρφωση των χώρων αναζήτησης. Οι ασαφώς καθορισμένες περιοχές μπορεί να οδηγήσουν σε λιγότερο από ιδανικές επιδόσεις του μοντέλου.
5. **Ζητήματα δεοντολογίας και μεροληψίας:** Τα αυτοματοποιημένα συστήματα ενδέχεται να συντηρήσουν ακούσια τις υπάρχουσες προκαταλήψεις στα δεδομένα, γεγονός που θα μπορούσε να οδηγήσει σε ανήθικα αποτελέσματα.

Παρόλο που το AutoML προσφέρει σημαντικά οφέλη στον εξορθολογισμό και την ενίσχυση των διαδικασιών μηχανικής μάθησης, είναι απαραίτητο να αντιμετωπιστούν οι περιορισμοί του για να εξασφαλιστεί η υπεύθυνη και αποτελεσματική εφαρμογή του.

1.4 Κίνητρο

Σε αυτό το εισαγωγικό κεφάλαιο της διπλωματικής εργασίας, έχει πραγματοποιηθεί εκτενής αναφορά σε βασικές έννοιες, όπως είναι η κατηγοριοποίηση δεδομένων, η κατηγοριοποίηση εγγύτερων γειτόνων και το AutoML. Ωστόσο, η κύρια έμφαση της παρούσας διπλωματικής εργασίας είναι στον κατηγοριοποιητή k-NN. Ο k-NN, όπως προαναφέρθηκε περιληπτικά νωρίτερα και θα αναλυθεί περαιτέρω στη συνέχεια, αποτελεί μια δημοφιλής προσέγγιση, με ιδιαίτερη ευκολία στη χρήση του και με υψηλή αποτελεσματικότητα και με μεγάλη ποικιλία εφαρμογών.

Ανεξαρτήτως από την ευκολία στη χρήση του, ένα από τα κύρια εμπόδια που αντιμετωπίζει η εφαρμογή του k-NN είναι η ανάγκη κατοχής εξειδικευμένων γνώσεων από τους χρήστες σε τομείς όπως η Εξόρυξη Δεδομένων και η Μηχανική Μάθηση. Επιπρόσθετα, έχει παρατηρηθεί το γεγονός ότι δεν υπάρχει αντίστοιχη εφαρμογή, δηλαδή εφαρμογή όπου όχι μόνο αξιοποιεί πλήρως τον κατηγοριοποιητή k-NN, αλλά και να στηρίζεται σε βασικά θεμέλια του AutoML. Έτσι, οι χρήστες που επιθυμούν να κατανοήσουν και να παρατηρήσουν τη λειτουργικότητα του κατηγοριοποιητή χωρίς να χρειαστεί να κατέχουν εξειδικευμένες εμπειρίες και γνώσεις πάνω στους εμπλεκόμενους τομείς, να είναι ικανοί να μπορούν να χρησιμοποιήσουν την εφαρμογή και να εκτελούν διάφορες εργασίες χωρίς εμπόδια. Είναι ταυτόχρονα απαιρήτη η αναφορά της ύπαρξης εφαρμογών και λογισμικών που εφαρμόζουν και τον συγκεκριμένο κατηγοριοποιητή, ωστόσο τα μειονεκτήματα υπερέχουν τα πλεονεκτήματα, οδηγώντας στην περιορισμένη χρήση του κατηγοριοποιητή. Ένα από τα κύρια μειονεκτήματα είναι το γεγονός ότι η χρήση των υπαρκτών εφαρμογών δεν είναι δωρεάν, αλλά είναι αναγκαία η καταβολή ορισμένου, και πολλές φορές μεγάλου, ποσού για πλήρη πρόσβαση στον κατηγοριοποιητή. Επίσης, σε ορισμένες από αυτές τις εφαρμογές απαιτείται η εγκατάσταση όχι μόνο του ίδιου του εργαλείου, αλλά και επιπλέον βιβλιοθηκών και πακέτων, με αποτέλεσμα η εφαρμογή να είναι όχι μόνο δυσνόητη και πολύπλοκη στη χρήση της, αλλά και να καταλαμβάνει σημαντικό χώρο στον ηλεκτρονικό ή φορητό υπολογιστή του χρήστη.

1.5 Συνεισφορά

Η αντιμετώπιση των ζητημάτων που προαναφέρονται στο προηγούμενο κεφάλαιο (1.4), αποσκοπεί στο να επιλύσει η παρούσα διπλωματική εργασία μέσω της ανάπτυξης μιας διαδικτυακής εφαρμογής, που ονομάζεται "AutoKNN". Το AutoKNN είναι λογισμικό ανοιχτού πηγαίου κώδικα, όπου όλοι οι ενδιαφερόμενοι χρήστες είναι ικανοί να αξιοποιήσουν πλήρως, τον κατηγοριοποιητή k-NN. Αναλυτικά, το AutoKNN επιτρέπει στους χρήστες να επιλέξουν επιθυμητά σύνολα δεδομένων, όπου βάσει αυτών να μπορέσουν να δημιουργήσουν προεκπαιδευμένα μοντέλα. Πριν την δημιουργία των μοντέλων, εμφανίζονται κύριες παραμέτρους όπου ο χρήστης μπορεί είτε να επιλέξει τις τιμές που επιθυμεί, είτε να τις αφήσει στις προκαθορισμένες τους τιμές, πράγμα που τον βοηθάει πολύ στην περίπτωση που δεν κατέχει σχετικές γνώσεις, για να μπορέσει να δημιουργήσει τα προεκπαιδευμένα μοντέλα. Μαζί με τον κατηγοριοποιητή k-NN, αξιοποιείται και η τεχνική διαχωρισμού συνόλου

δεδομένων σε σύνολα εκπαίδευσης και δοκιμής (train test split) έτσι ώστε να πραγματοποιηθεί αξιολόγηση αποτελεσμάτων του νέου μοντέλου, όπου εμφανίζεται αναλυτική αναφορά σχετική με μετρικές απόδοσης, τόσο για κάθε κλάση, όσο και για το μοντέλο στο σύνολο. Ύστερα, ο χρήστης μπορεί να αποθηκεύσει το καινούργιο μοντέλο, όπου έχει την επιλογή να το χρησιμοποιήσει για να προβλέψει μη κατηγοριοποιημένα στιγμιότυπα.

Όλες οι λειτουργίες της διαδικτυακής εφαρμογής AutoKNN παρέχονται μέσω μιας διαδικτυακής Διεπαφής Προγραμματισμού Εφαρμογών (Application Programming Interface - API), όπου είναι ελεύθερο για όλους όσους ενδιαφέρονται. Έτσι, είναι δυνατή η χρήση της από την κοινότητα των προγραμματιστών, όπου, για παράδειγμα, μπορούν να την επεκτήνουν περαιτέρω, να δημιουργήσουν νέες εφαρμογές και να αξιοποιήσουν ότι επιθυμούν από το AutoKNN. Επιπρόσθετα, υπάρχει σύγχρονη γραφική διεπαφή χρήστη (Graphical User Interface - GUI), όπου ο χρήστης μπορεί να παρατηρήσει και να εκτελέσει τις λειτουργίες του AutoKNN. Η συγκεκριμένη εφαρμογή φιλοξενείται από έναν διακομιστή (server) του Τμήματος Μηχανικών Πληροφορικής και Ηλεκτρονικών Συστημάτων, αλλά ο πηγαίος κώδικας της εφαρμογής είναι διαθέσιμος στη πλατφόρμα GitHub, με αποτέλεσμα να μπορούν οι χρήστες να την αξιοποιήσουν και να τη φιλοξενήσουν σε εξωτερικό διακομιστή, για τις δικές τους ανάγκες. Τέλος, το συμπέρασμα είναι ότι πρόκειται για ένα λογισμικό όπου αντιμετωπίζει τους περιορισμούς των ήδη υπαρκτών εφαρμογών, και μπορεί να αξιοποιηθεί πλήρως από χρήστες, όπως ερευνητές, φοιτητές, προγραμματιστές, ανθρώπους ειδικούς στην Εξόρυξη Δεδομένων και Μηχανική Μάθηση, αλλά και σε απλούς ενδιαφερόμενους στους σχετικούς τομείς.

Η εφαρμογή AutoKNN είναι διαθέσιμη στο παρακάτω σύνδεσμο:

https://kclusterhub.iee.ihu.gr/autoknn

Ο κώδικας της εφαρμογής είναι διαθέσιμος στον ακόλουθο σύνδεσμο της GitHub:

https://github.com/KostasKyriakosBatsios/AutoKNN

1.6 Οργάνωση εργασίας

Η παρούσα διπλωματική εργασία οργανώνεται σε εφτά κεφάλαια. Το πρώτο κεφάλαιο, όπου μόλις ολοκληρώθηκε, αφορά την εισαγωγή σε θεμελιώδεις έννοιες όπως είναι η κατηγοριοποίηση δεδομένων, η κατηγοριοποίηση εγγύτερων γειτόνων, και το AutoML. Επιπλέον, έγινε ανάλυση τόσο του κίνητρου αυτής της διπλωματικής εργασίας, όσο και στη συνεισφορά της. Παρακάτω γίνεται μια συνοπτική αναφορά των ανερχόμενων κεφαλαίων.

Το δεύτερο κεφάλαιο αφορά μια αναλυτική εμβάθυνση στον k-NN, όπου επισημαίνονται ο ορισμός της, η μεθοδολογία της, πλεονεκτήματα και μειονεκτήτατα, οι μετρικές απόστασεις, καθώς και για την παράμετρο K επιπλέον πληροφορίες και πειράματα.

Το τρίτο κεφάλαιο σχετίζεται με τις γλώσσες προγραμματισμού και τις τεχνολογίες που εμπλέκονται στην ανάπτυξη της εφαρμογής, και στο επίπεδο του front-end, αλλά και στο επίπεδο back-end.

Το τέταρτο κεφάλαιο αφορά τον τρόπο σχεδιασμού και υλοποίησης της διαδικτυακής εφαρμογής AutoKNN. Αρχικά, παρουσιάζονται οι λειτουργικές απαιτήσεις, καθώς και η αρχιτεκτονική του AutoKNN. Ύστερα αναλύονται λειτουργίες του front-end και back-end. Τέλος, εμφάνιση σχετικών στιγμιοτύπων προγραμματιστικού κώδικα.

Το πέμπτο κεφάλαιο αφορά την παρουσίαση των λειτουργιών του AutoKNN μέσω του GUI, όπου συμβάλλει στη καλύτερη κατανόηση των χρηστών για την πλήρη και ορθή χρήση της διαδικτυακής εφαρμογής.

Το έκτο κεφάλαιο αφορά την αξιολόγηση του AutoKNN που πραγματοποιείται μέσω σχετικού ερωτηματολογίου τύπου Κλίμακας Χρηστικότητας του Συστήματος (System Usability Scale - SUS). Παρουσιάζονται λεπτομερώς τα αποτελέσματα από τις μετρήσεις που αφορούν την εμπειρία του χρήστη.

Το έβδομο κεφάλαιο, όπου με αυτό ολοκληρώνεται η διπλωματική εργασία, αφορά τα συμπεράσματα που προκύπτουν από αυτή την εργασία, καθώς και τις πιθανές μελλοντικές επεκτάσεις της εφαρμογής.

Κεφάλαιο 2ο: Κατηγοριοποίηση μέσω αναζήτησης εγγύτερων γειτόνων

2.1 Ο κατηγοριοποιητής k εγγύτερων γειτόνων

Ο κατηγοριοποιητής k-NN, όπως προαναφέραμε και στο κεφάλαιο 1.2, είναι ένας βασικός και απλός αλγόριθμος στον τομέα της Μηχανικής Μάθησης, ο οποίος είναι ευρέως αναγνωρισμένος για την ευκολία χρήσης και την προσαρμοστικότητά του. Ο k-NN, που είναι μια μέθοδος που στηρίζεται σε περιπτώσεις και δεν είναι παραμετρική, εξαρτάται από τη διατήρηση του πλήρους συνόλου δεδομένων εκπαίδευσης και παράγει προβλέψεις εντοπίζοντας τις K πλησιέστερες επισημειωμένες περιπτώσεις σε ένα συγκεκριμένο δείγμα δοκιμής στο χώρο των χαρακτηριστικών [9], [10]. Σε αντίθεση με τους παραμετρικούς αλγορίθμους που δημιουργούν ρητά μοντέλα κατά τη διάρκεια της εκπαίδευσης, ο k-NN λειτουργεί χρησιμοποιώντας μια προσέγγιση τεμπέλικης (lazy) μάθησης, αναβάλλοντας τους υπολογισμούς μέχρι να χρειαστεί ταξινόμηση ή παλινδρόμηση [7]. Η μέθοδος αυτή είναι πολύ ευέλικτη σε διαφορετικές κατανομές δεδομένων, καθώς δεν βασίζεται σε υποθέσεις σχετικά με τα θεμελιώδη πρότυπα των δεδομένων [8]. Ο k-NN χρησιμοποιεί μετρικές αποστάσεων όπως η Euclidean, η Manhattan ή η Minkowski για να αξιολογήσει την ομοιότητα μεταξύ των σημείων δεδομένων, εντοπίζοντας τους πλησιέστερους γείτονες και χρησιμοποιώντας τις ετικέτες ή τις τιμές τους για την πραγματοποίηση προβλέψεων. Ο απλός σχεδιασμός και η ευκολία εφαρμογής του έχουν καθιερώσει ο k-NN ως βασικό στοιχείο της μηχανικής μάθησης, ειδικά σε θέματα που απαιτούν ισχυρές επιδόσεις με μικρή προεπεξεργασία.

Η μέθοδος k-NN ταξινομεί ένα δείγμα δοκιμής προσδιορίζοντας την πλειοψηφική κλάση μεταξύ των k πλησιέστερων γειτόνων του στα δεδομένα εκπαίδευσης. Η διαδικασία ταξινόμησης ξεκινά με τον προσδιορισμό της απόστασης μεταξύ του δείγματος δοκιμής και κάθε σημείου δεδομένων στο σύνολο εκπαίδευσης χρησιμοποιώντας μια επιλεγμένη μετρική απόστασης, όπως η Euclidean, η Manhattan ή η Minkowski. Αυτές οι αποστάσεις καθορίζουν πόσο κοντά είναι τα δείγματα εκπαίδευσης στο δείγμα δοκιμής. Στη συνέχεια, ο αλγόριθμος ταξινομεί αυτές τις αποστάσεις και επιλέγει τις k μικρότερες τιμές, οι οποίες αντιπροσωπεύουν τους k πλησιέστερους γείτονες. Στις εργασίες κατηγοριοποίησης, το δείγμα δοκιμής κατηγοριοποιείται στην πιο κοινή κλάση μεταξύ των γειτόνων του. Στις εργασίες παλινδρόμησης, ο αλγόριθμος δίνει συνήθως στο δείγμα δοκιμής μια τιμή που προκύπτει από το μέσο όρο των τιμών των γειτονικών του δειγμάτων[15]. Αξιολογώντας δυναμικά τους πλησιέστερους γείτονες για κάθε πρόβλεψη, ο k-NN χρησιμοποιεί την τοπική διάταξη των δεδομένων, καθιστώντας τον πολύ αποτελεσματικό για προκλήσεις που περιλαμβάνουν περίπλοκα, μη γραμμικά όρια [16].

Παρακάτω, στον Αλγόριθμο 1 [8], βλέπουμε την επεξήγηση του k-NN με πολύ συνωπτικό τρόπο.

Algorithm 1 Βασικός KNN Αλγόριθμος

Require: Δείγματα εκπαίδευσης D, Δείγμα δοκιμής d, K

Ensure: Ετικέτα κλάσης του δείγματος δοκιμής

1. Υπολογίστε την απόσταση μεταξή του d και κάθε δείγματος στο D
2. Επιλέξτε τα K δείγματα στο D που είναι πλησιέστερα στο d; συμβολίστε το σύνολο με P ($\in$ D)
3. Αναθέστε d την κλάση που είναι η πιο συχνή κλάση (ή η κλάση πλειοψηφίας)

Ο Πίνακας 2.1 [8], απεικονίζει ένα παράδειγμα τρόπου υπολογισμού του k-NN, με βάσει και τον Αλγόριθμο 1, όπου έχουμε τρία δείγματα εκπαίδευσης και ένα δείγμα δοκιμής. Επίσης, η μετρική απόσταση στο συγκεκριμένο παράδειγμα είναι η Euclidean απόσταση, και με βάσει αυτή, υπολογίζεται η απόσταση ανάμεσα στο δείγμα δοκιμής και σε κάθε δείγμα εκπαίδευσης. Η κλάση στην οποία ανήκει το δείγμα δοκιμής, εξαρτάται από το K που έχουμε λάβει υπόψην, καθώς από αυτό βρίσκουμε και τη μικρότερη απόσταση. Για παράδειγμα, αν το k είναι 1, ανήκει στη κλάση 1. Αλλιώς, αν το k είναι 3, ανήκει στη κλάση 2.

Samples	X1	X2	X3	class	Distance
Training sample(1)	5	4	3	1	$D = \sqrt{(4-5)^2-(4-4)^2-(2-3)^2} = 1.4$
Training sample(2)	1	2	2	2	$D = \sqrt{(4-1)^2-(4-2)^2-(2-2)^2} = 3.6$
Training sample(3)	1	2	3	2	$D = \sqrt{(4-1)^2-(4-2)^2-(2-3)^2} = 3.7$
Test sample	4	4	2	?	

Table 2-1 Παραδείγματα δεδομένων εκπαίδευσης και δοκιμής με αποστάσεις.

Η λειτουργία του κατηγοριοποιητή k-NN επικεντρώνεται στην ικανότητά του να κάνει προβλέψεις με βάση την εγγύτητα των σημείων δεδομένων, βασιζόμενος στην ιδέα ότι παρόμοια παραδείγματα βρίσκονται κοντά το ένα στο άλλο στο χώρο των χαρακτηριστικών. Στην ταξινόμηση, ο αλγόριθμος προσδιορίζει τους K πλησιέστερους γείτονες ενός δείγματος δοκιμής χρησιμοποιώντας μια καθορισμένη μετρική απόστασης, όπως η Euclidean απόσταση, η απόσταση Manhattan ή η απόσταση Chebyshev, και στη συνέχεια αναθέτει το δείγμα δοκιμής στην κυρίαρχη κλάση μεταξύ των γειτόνων του [17]. Σε εργασίες παλινδρόμησης, ο k-NN εκτιμά μια αριθμητική τιμή με τον μέσο όρο των αποτελεσμάτων των K πλησιέστερων γειτόνων. Η προσαρμοστικότητά του αναδεικνύεται από την ικανότητά του να τροποποιεί διαφορετικές μετρικές απόστασης και τιμές του K, επιτρέποντάς του να διαχειρίζεται ένα εύρος συνόλων δεδομένων με διάφορες κατανομές και χαρακτηριστικά. Επιπλέον, η ικανότητα του k-NN να δημιουργεί τοπικά όρια αποφάσεων του επιτρέπει να εντοπίζει αποτελεσματικά μη γραμμικά μοτίβα και να παρέχει αξιόπιστες προβλέψεις σε διάφορους τομείς.

Ο αλγόριθμος k-NN παρουσιάζει διάφορα πλεονεκτήματα που συμβάλλουν στη διάδοσή του στη μηχανική μάθηση, ωστόσο, όπως και κάθε τεχνολογία, έχει τους δικούς της περιορισμούς που θα πρέπει να λαμβάνονται υπόψη κατά τη χρήση της για ζητήματα του πραγματικού κόσμου. Τα επόμενα σημεία αναδεικνύουν τα κύρια πλεονεκτήματα και μειονεκτήματα του k-NN.

Πλεονεκτήματα

1. **Μη παραμετρική προσαρμοστικότητα:** Ο k-NN δεν επιβάλλει υποθέσεις σχετικά με την υποκείμενη κατανομή των δεδομένων, επιτρέποντάς του να διαχειρίζεται με επάρκεια μη γραμμικές και περίπλοκες κατανομές δεδομένων.
2. **Ποικίλες μετρικές απόστασης:** Υποδέχεται μια σειρά από μετρικές αποστάσεων, συμπεριλαμβανομένων των Euclidean, Manhattan και Minkowski, επιτρέποντας προσαρμογές με βάση διαφορετικά χαρακτηριστικά συνόλου δεδομένων.
3. **Σαφήνεια και κατανόηση:** Ο αλγόριθμος είναι εύκολος στην κατανόηση και στην εκτέλεση, καθιστώντας τον διαθέσιμο τόσο για αρχάριους όσο και για εμπειρογνώμονες.
4. **Ευρεία δυνατότητα χρήσης:** Ο k-NN εφαρμόζεται σε διάφορους τομείς, όπως η ταξινόμηση κειμένου, η ανίχνευση εισβολών και οι προκλήσεις παλινδρόμησης.
5. **Προσαρμοστική λήψη αποφάσεων:** Προσαρμόζεται αποτελεσματικά στις τοπικές διαμορφώσεις των δεδομένων, διευκολύνοντας την ακριβή ταξινόμηση και παλινδρόμηση σε σύνολα δεδομένων με μη γραμμικά όρια απόφασης.

Μειονεκτήματα

1. **Υπολογιστική αναποτελεσματικότητα:** Ο k-NN απαιτεί την αποθήκευση και τον υπολογισμό των αποστάσεων για κάθε σημείο δεδομένων εκπαίδευσης για κάθε πρόβλεψη, καθιστώντας τον έτσι υπολογιστικά δαπανηρό για εκτεταμένα σύνολα δεδομένων.
2. **Κατάρα της διαστατικότητας:** Σε χώρους με πολλές διαστάσεις, οι αποστάσεις χάνουν τη σημασία τους, με αποτέλεσμα να μειώνεται ενδεχομένως η αποτελεσματικότητα του αλγορίθμου, εκτός αν εφαρμοστεί μείωση διαστάσεων ή επιλογή χαρακτηριστικών.
3. **Ευαισθησία σε θορυβώδη ή ασήμαντα χαρακτηριστικά:** Οι θορυβώδεις ή ασήμαντες πληροφορίες μπορούν να διαταράξουν τις εκτιμήσεις απόστασης, με αποτέλεσμα τη μείωση της ακρίβειας, εκτός εάν πραγματοποιείται ενδελεχής προεπεξεργασία.
4. **Εξάρτηση από το k και τη μετρική απόσταση:** Η αποτελεσματικότητα του k-NN επηρεάζεται σε μεγάλο βαθμό από την επιλογή του k (αριθμός γειτόνων) και του μέτρου απόστασης, καθιστώντας αναγκαία την προσαρμογή και την επαλήθευση για τη βελτίωση των αποτελεσμάτων.
5. **Κατανάλωση μνήμης:** Ο k-NN πρέπει να αποθηκεύσει το πλήρες σύνολο δεδομένων εκπαίδευσης στη μνήμη, γεγονός που μπορεί να είναι απαιτητικό από άποψη πόρων για εκτεταμένα σύνολα δεδομένων.

2.2 Μετρικές απόστασης

Οι μετρικές απόστασης είναι ζωτικής σημασίας ως παράμετρος του κατηγοριοποιητή k-NN, επηρεάζοντας άμεσα την ικανότητά του να αξιολογεί την ομοιότητα μεταξύ των σημείων δεδομένων. Δεδομένου ότι ο k-NN λειτουργεί με την εύρεση των k πλησιέστερων γειτόνων εντός του χώρου των χαρακτηριστικών, η επιλεγμένη μετρική απόστασης καθορίζει τον τρόπο μέτρησης της «εγγύτητας». Οι τυπικές μετρικές περιλαμβάνουν τις μετρικές Euclidean, Manhattan, Chebyshev και Minkowski, κάθε μία από τις οποίες είναι προσαρμοσμένη για συγκεκριμένους τύπους δεδομένων και προβληματικές περιοχές [7], [18]. Αυτές οι μετρικές καθορίζουν την εγγύτητα μεταξύ των σημείων δεδομένων σύμφωνα με τα χαρακτηριστικά τους, επιτρέποντας στον k-NN να κατανέμει ετικέτες ή να παράγει προβλέψεις [19].

Η δυνατότητα επιλογής μιας μετρικής απόστασης επιτρέπει στον k-NN να προσαρμόζεται σε διαφορετικές κατανομές δεδομένων και τύπους χαρακτηριστικών. Για παράδειγμα, η Euclidean απόσταση είναι κατάλληλη για συνεχή, κανονικοποιημένα δεδομένα, ενώ η Manhattan αποδίδει καλύτερα σε διαμορφώσεις υψηλών διαστάσεων ή σε διαμορφώσεις που μοιάζουν με πλέγμα [8], [20]. Επιπλέον, εξελιγμένες μετρικές όπως η Minkowski παρέχουν ρυθμιζόμενες παραμέτρους για τη διαμόρφωση της ευαισθησίας των μετρήσεων απόστασης [18]. Αυτή η ευελιξία αναδεικνύει την ανάγκη επιλογής μιας κατάλληλης μετρικής για την καλύτερη απόδοση του κατηγοριοποιητή. Ωστόσο, η εξάρτηση από τις μετρικές απόστασης καθιστά τον k-NN ευάλωτο σε άσχετα ή θορυβώδη χαρακτηριστικά, υπογραμμίζοντας την ανάγκη για σχολαστική προεπεξεργασία δεδομένων και επιλογή χαρακτηριστικών [9].

Η επιλογή μιας κατάλληλης μετρικής απόστασης είναι ζωτικής σημασίας για την ενίσχυση της αποτελεσματικότητας του ταξινομητή k-NN. Κάθε μετρική παρέχει ξεχωριστά πλεονεκτήματα που ταιριάζουν σε συγκεκριμένους τύπους δεδομένων και προβληματικές περιοχές, επιτρέποντας στο k-NN να προσαρμόζεται σε διάφορες εφαρμογές [18], [19]. Παρόλα αυτά, η επιτυχία αυτών των μετρικών εξαρτάται από την κατάλληλη προεπεξεργασία, όπως η κανονικοποίηση και η επιλογή χαρακτηριστικών, διασφαλίζοντας ότι η επιλεγμένη μετρική απόσταση αντικατοπτρίζει τις σημαντικές σχέσεις εντός του συνόλου δεδομένων [20]. Η κατανόηση των χαρακτηριστικών και των εφαρμογών των αποστάσεων Ευκλείδειας, Manhattan, Chebyshev και Minkowski επιτρέπει στους ειδικούς να

κάνουν τεκμηριωμένες επιλογές, βελτιώνοντας έτσι την ακρίβεια και την ανθεκτικότητα του κατηγοριοποιητή.

2.2.1 Ευκλείδεια Απόσταση

Μια μετρική απόστασης που χρησιμοποιείται συνήθως στον k-NN είναι η Ευκλείδεια απόσταση. Υπολογίζει την ευθεία απόσταση μεταξύ δύο θέσεων στον Ευκλείδειο χώρο. Είναι ιδιαίτερα αποτελεσματική όταν η κατανομή των δεδομένων είναι ομαλή και τα χαρακτηριστικά έχουν τυποποιηθεί. Αυτή η μετρική είναι κατάλληλη για πολυάριθμες παραδοσιακές εργασίες Μηχανικής Μάθησης, όπου τα δεδομένα έχουν εγγενή γεωμετρική σημασία. Ο αλγόριθμος υπολογίζει την ευκλείδεια απόσταση μεταξύ του σημείου δοκιμής και κάθε σημείου εκπαίδευσης, προσδιορίζει τους k πλησιέστερους γείτονες και ταξινομεί το δείγμα δοκιμής σύμφωνα με την επικρατούσα κλάση μεταξύ αυτών των γειτόνων. Η μαθηματική αναπαράσταση της ευκλείδειας απόστασης έχει ως εξής:

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

όπου τα x και y αντιπροσωπεύουν διανύσματα στον n-διάστατο χώρο.

2.2.2 Manhattan (City Block) Απόσταση

Η απόσταση Manhattan, που αναφέρεται ως απόσταση οικοδομικού τετραγώνου (City Block), προσδιορίζει το σύνολο των απόλυτων διαφορών μεταξύ των καρτεσιανών συντεταγμένων δύο θέσεων. Αυτή η απόσταση είναι ιδιαίτερα επωφελής σε σύνολα δεδομένων που μοιάζουν με πλέγμα ή σε καταστάσεις όπου η κίνηση γίνεται κατά μήκος αξόνων, όπως στους δρόμους της πόλης όπου η κίνηση ακολουθεί ορθογώνιες διαδρομές. Σε αντίθεση με την ευκλείδεια απόσταση, η οποία μετράει την άμεση απόσταση, η απόσταση Μανχάταν αξιολογεί τον διαχωρισμό μεταξύ δύο σημείων διατρέχοντας το πλέγμα, καθιστώντας την πιο ισχυρή έναντι ακραίων τιμών. Στον k-NN, η απόσταση Manhattan χρησιμοποιείται για την αξιολόγηση της εγγύτητας κάθε δείγματος εκπαίδευσης με το δείγμα δοκιμής, διευκολύνοντας τον εντοπισμό των πλησιέστερων γειτόνων. Η απόσταση Manhattan περιγράφεται ως εξής:

$$d(x, y) = \sum_{i=1}^{n} |x_i - y_i|$$

όπου τα x και y αντιπροσωπεύουν διανύσματα στον n-διάστατο χώρο.

2.2.3 Chebyshev Απόσταση

Η απόσταση Chebyshev υπολογίζει τη μέγιστη απόλυτη διαφορά μεταξύ των συντεταγμένων δύο σημείων. Αυτή η μετρική είναι ιδανική για καταστάσεις όπου όλες οι διαστάσεις έχουν την ίδια σημασία και απαιτούν ιεράρχηση. Εφαρμόζεται συνήθως σε περιπτώσεις όπου η μεγαλύτερη διακύμανση προς οποιαδήποτε κατεύθυνση είναι το πιο σημαντικό στοιχείο. Στο πλαίσιο του k-NN, η απόσταση Chebyshev αξιολογεί τη μεγαλύτερη διαφορά μεταξύ του δείγματος δοκιμής και των

δειγμάτων εκπαίδευσης, η οποία αποδεικνύεται ιδιαίτερα επωφελής όταν εξετάζονται πολυδιάστατα δεδομένα με χαρακτηριστικά που μπορεί να έχουν σημαντικά διαφορετικά εύρη. Η απόσταση Chebyshev περιγράφεται ως εξής:

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

όπου τα x και y αντιπροσωπεύουν διανύσματα, και το i αντιπροσωπεύει τον δείκτη της μεγαλύτερης διαφοράς μεταξύ των αντίστοιχων συνιστωσών τους.

2.2.4 Minkowski Απόσταση

Η απόσταση Minkowski χρησιμεύει ως μια γενική μετρική για την ποσοτική μέτρηση της απόστασης, που περιλαμβάνει τόσο την Ευκλείδεια απόσταση όσο και την απόσταση Manhattan ως ειδικές περιπτώσεις. Η μετρική προσφέρει προσαρμοστικότητα με την αλλαγή της παραμέτρου p, η οποία αλλάζει την προσέγγιση για τον υπολογισμό της απόστασης. Για p=1, η απόσταση Minkowski ισούται με την απόσταση Manhattan, ενώ για p=2, με την Ευκλείδεια απόσταση. Αυτή η ευελιξία καθιστά την απόσταση Minkowski κατάλληλη για διάφορα είδη δεδομένων και κατανομών. Στη μέθοδο k-NN, η παράμετρος p μπορεί να ρυθμιστεί ώστε να αντιπροσωπεύει διάφορους βαθμούς ευαισθησίας στις διαφορές μεταξύ των σημείων δεδομένων. Ο τυπικός τύπος για την απόσταση Μινκόφσκι είναι:

$$d(x, y) = \left(\sum_{i=1}^{n} |x_i - y_i|^p \right)^{\frac{1}{p}}$$

όπου το p καθορίζει τον τύπο της απόστασης (Manhattan για p=1, Ευκλείδεια για p=2).

2.3 Η παράμετρος k

Η παράμετρος k στον αλγόριθμο k-NN υποδεικνύει πόσοι πλησιέστεροι γείτονες λαμβάνονται υπόψη κατά τον προσδιορισμό της ταξινόμησης ή την πρόβλεψη της τιμής ενός νέου σημείου δεδομένων. Αποτελεί κρίσιμο παράγοντα για τη διαμόρφωση του τρόπου με τον οποίο ο αλγόριθμος αντιλαμβάνεται την τοπική δομή του συνόλου δεδομένων [21]. Στις εργασίες κατηγοριοποίησης, το k δηλώνει τον αριθμό των γειτονικών σημείων των οποίων η επικρατούσα κλάση καθορίζει την ετικέτα του δείγματος δοκιμής [22]. Στις εργασίες παλινδρόμησης, το k καθορίζει πόσες κοντινές τιμές υπολογίζονται κατά μέσο όρο για την πρόβλεψη της εξόδου. Με την επιλογή του k, ο αλγόριθμος ρυθμίζει την επίδραση των κοντινών σε σύγκριση με τα μακρινά σημεία δεδομένων, γεγονός που επηρεάζει άμεσα την ανταπόκρισή του στις τοπικές σε σύγκριση με τις παγκόσμιες τάσεις στο σύνολο δεδομένων.

Ο ρόλος της παραμέτρου k στον αλγόριθμο k-NN είναι να λειτουργεί ως παράγοντας ελέγχου για τον καθορισμό της έκτασης της γειτονιάς που χρησιμοποιείται στη λήψη αποφάσεων. Επιλέγοντας k, ο αλγόριθμος αξιολογεί τα k πλησιέστερα σημεία δεδομένων σε ένα συγκεκριμένο δείγμα δοκιμής χρησιμοποιώντας μια επιλεγμένη μετρική απόστασης, όπως η Ευκλείδεια απόσταση ή η απόσταση Manhattan [22]. Στις εργασίες κατηγοριοποίησης, το k επιτρέπει στον αλγόριθμο να κατηγοριοποιήσει το δείγμα δοκιμής στην πιο συχνή κλάση μεταξύ των k γειτόνων, λαμβάνοντας έτσι υπόψη τα τοπικά

μοτίβα στα δεδομένα. Στις εργασίες παλινδρόμησης, το k βοηθά στον υπολογισμό της εξόδου με τη μέση τιμή των τιμών των πλησιέστερων γειτόνων, χρησιμοποιώντας την εγγύτητα για αριθμητικές προβλέψεις [21]. Η επιλογή του k επηρεάζει την απόδοση του μοντέλου: χαμηλότερες τιμές του k αυξάνουν την ευαισθησία στο θόρυβο και τις τοπικές διακυμάνσεις, ενώ υψηλότερες τιμές του k αποδίδουν πιο ομαλές προβλέψεις, αλλά μπορεί να χάσουν πολύπλοκα μοτίβα.

Στο παρακάτω Σχήμα 2.1 [23], απεικονίζεται ολοκληρωμένα η κατηγοριοποίηση με τη χρήση του αλγορίθμου k-NN, στη περίπτωση που η μετρική απόστασης είναι η Ευκλείδεια.

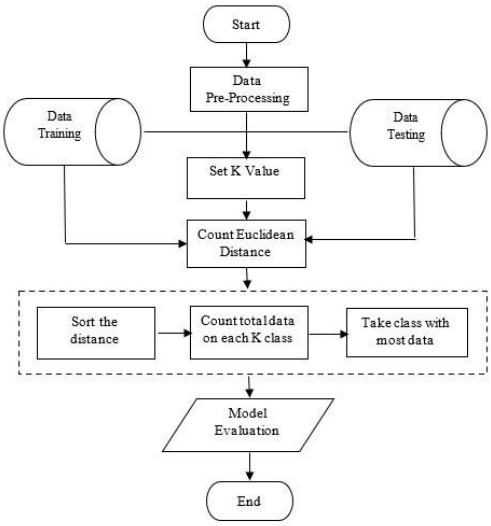


Figure 2-1 Διάγραμμα ροής της κατηγοριοποίησης με χρήση K-NN

2.3.1 Ρύθμιση παραμέτρου k

Η ρύθμιση (tuning) της παραμέτρου k είναι μια κρίσιμη διαδικασία για την ενίσχυση της απόδοσης του κατηγοριοποιητή k-NN. Η τιμή του k επηρεάζει άμεσα την ικανότητα του μοντέλου να διαχειρίζεται αποτελεσματικά τη μεροληψία και τη διακύμανση. Οι χαμηλές τιμές k παράγουν πολύ τοπικές οριοθετήσεις αποφάσεων που κατανοούν τις διαφοροποιήσεις των δεδομένων, αλλά διατρέχουν τον κίνδυνο υπερπροσαρμογής (overfitting). Ωστόσο, οι υψηλές τιμές k μπορεί να εξομαλύνουν τα όρια της απόφασης, παραλείποντας ενδεχομένως κρίσιμες πληροφορίες και οδηγώντας σε υποπροσαρμογή (underfitting). Κατά τη διαδικασία ρύθμισης, οι πιθανές τιμές k διερευνώνται συστηματικά χρησιμοποιώντας τεχνικές όπως η αναζήτηση πλέγματος (grid search) και η διασταυρούμενη (cross validation) [24]. Προσδιορίζοντας το βέλτιστο k που ταιριάζει στην κατανομή και την οργάνωση του συνόλου δεδομένων, αυτές οι μέθοδοι εξασφαλίζουν την καλύτερη δυνατή ισορροπία μεταξύ της προβλεπτικής απόδοσης και της πολυπλοκότητας του μοντέλου.

Για τη σωστή ρύθμιση της παραμέτρου k στον αλγόριθμο k-NN, είναι απαραίτητο να ληφθούν υπόψη διάφοροι παράγοντες που συνδέονται μεταξύ τους και επηρεάζουν την απόδοση και την ακρίβεια του μοντέλου [21]. Τα στοιχεία αυτά επηρεάζουν την ικανότητα του αλγορίθμου να γενικεύει σε διάφορα σύνολα δεδομένων και να αντιμετωπίζει την ισορροπία μεταξύ υπερπροσαρμογής και υποπροσαρμογής [22]. Με την εξέταση της διάταξης του συνόλου δεδομένων, των χαρακτηριστικών γνωρισμάτων και των περιορισμών επεξεργασίας, οι επαγγελματίες μπορούν να κατευθύνουν τη διαδικασία προσαρμογής ώστε να ανακαλύψουν ένα ιδανικό k [24]. Η διαδικασία αυτή απαιτεί σχολαστικό σχεδιασμό και την εκτέλεση συγκεκριμένων όρων που εγγυώνται αξιόπιστα και αποτελεσματικά αποτελέσματα συντονισμού.

- **Εξισορρόπηση μεροληψίας και διακύμανσης:** Μια μικρή τιμή k επιτρέπει στο μοντέλο να δημιουργεί πολύ συγκεκριμένα όρια απόφασης που μπορούν να εντοπίσουν μικρές διαφορές στα δεδομένα, αλλά μπορεί να είναι επιρρεπή σε υπερπροσαρμογή θορύβου ή ανωμαλιών. Αντίθετα, οι μεγαλύτερες τιμές k οδηγούν σε ομαλότερα όρια απόφασης με τη μέση εκτίμηση της επίδρασης των απομακρυσμένων γειτόνων, μειώνοντας τη διακύμανση και αυξάνοντας ταυτόχρονα τον κίνδυνο υποπροσαρμογής. Η εύρεση της σωστής ισορροπίας απαιτεί τον προσδιορισμό μιας τιμής k που λαμβάνει υπόψη το εγγενές επίπεδο θορύβου και την πολυπλοκότητα του συνόλου δεδομένων.
- **Προετοιμασία συνόλου δεδομένων:** Οι υπολογισμοί ομοιόμορφων αποστάσεων είναι απαραίτητοι στον k-NN, ιδίως όταν τα χαρακτηριστικά αξιολογούνται σε διαφορετικές κλίμακες. Η κανονικοποίηση ή η ισχυρή κλιμάκωση εγγυάται ότι όλα τα χαρακτηριστικά έχουν ίσο αντίκτυπο στους υπολογισμούς της απόστασης, αποτρέποντας τα χαρακτηριστικά με μεγαλύτερα εύρη από το να επισκιάζουν τη μετρική της απόστασης. Η προεπεξεργασία είναι ιδιαίτερα σημαντική σε σύνολα δεδομένων που περιέχουν διάφορους τύπους δεδομένων ή έχουν υψηλή διαστατικότητα.
- **Υπολογιστική βιωσιμότητα:** Η προσαρμογή του k απαιτεί συχνά υπολογιστικά απαιτητικές τεχνικές, όπως η διασταυρούμενη επικύρωση ή η αναζήτηση πλέγματος. Για εκτεταμένα σύνολα δεδομένων, αυτές οι διαδικασίες μπορεί να μετατραπούν σε συμφόρηση, απαιτώντας επεκτάσιμους αλγορίθμους ή ευρετικές μεθόδους για την ενίσχυση της αποδοτικότητας των υπολογισμών. Εξελιγμένες μέθοδοι, όπως οι προσεγγιστικές αναζητήσεις του πλησιέστερου γείτονα, μπορούν να βοηθήσουν στην ελαχιστοποίηση της επιβάρυνσης που συνδέεται με τον συντονισμό.
- **Συμβουλές συγκεκριμένου τομέα:** Η εξοικείωση με το σύνολο δεδομένων ή τα σχετικά ζητήματα μπορεί να βοηθήσει σημαντικά στην επιλογή του κατάλληλου εύρους τιμών k. Ευρετικές μέθοδοι που προέρχονται από προηγούμενες έρευνες ή βιομηχανικά πρότυπα μπορούν να περιορίσουν την περιοχή αναζήτησης, ελαχιστοποιώντας την ανάγκη για εκτεταμένες δοκιμές. Για παράδειγμα, μια τιμή k μεταξύ 3 και 10 είναι συνήθως ένα κατάλληλο σημείο εκκίνησης για εργασίες ταξινόμησης, ενώ υψηλότερες τιμές k μπορεί να αποδίδουν καλύτερα για θέματα παλινδρόμησης.

Η ρύθμιση της παραμέτρου k στον αλγόριθμο k-NN είναι φυσικά ένα δύσκολο και απαιτητικό εγχείρημα, κυρίως λόγω των επαναληπτικών και υπολογιστικά εντατικών χαρακτηριστικών της διαδικασίας. Κάθε πιθανή τιμή k πρέπει να αξιολογείται μεθοδικά, συχνά σε διάφορες φορές διασταυρούμενης επικύρωσης, καθιστώντας τη διαδικασία ιδιαίτερα χρονοβόρα για εκτεταμένα σύνολα δεδομένων με υψηλή διαστατικότητα [22], [24]. Επιπλέον, η έλλειψη καθολικά σχετικών τιμών k απαιτεί από τους επαγγελματίες να ασχοληθούν με εκτεταμένο πειραματισμό, γεγονός που επιμηκύνει τη διαδικασία [21]. Αυτά τα εμπόδια υπογραμμίζουν την ανάγκη για ισχυρές υπολογιστικές ικανότητες και σχολαστική οργάνωση για την αποτελεσματική ενίσχυση του k.

- **Υπολογιστικό κόστος:** Ο προσδιορισμός του καλύτερου k απαιτεί την επανειλημμένη αξιολόγηση διαφόρων τιμών μέσω τεχνικών όπως η αναζήτηση πλέγματος ή η διασταυρούμενη επικύρωση. Αυτές οι μέθοδοι απαιτούν τον υπολογισμό των αποστάσεων μεταξύ των σημείων δεδομένων σε κάθε υποσύνολο του συνόλου δεδομένων, οδηγώντας σε εκθετική αύξηση του χρόνου υπολογισμού καθώς αυξάνεται είτε το μέγεθος του συνόλου δεδομένων είτε η διαστατικότητά του. Για εκτεταμένες εφαρμογές, η διαδικασία αυτή μπορεί να απαιτήσει ώρες ή ακόμη και ημέρες, ανάλογα με τους διαθέσιμους υπολογιστικούς πόρους. Οι εξελιγμένες μέθοδοι, όπως η παραλληλοποίηση ή η αναζήτηση του πλησιέστερου γείτονα, μπορούν να μειώσουν τον χρόνο εκτέλεσης, αλλά μπορεί να αντιμετωπίζουν προκλήσεις με δεδομένα εξαιρετικά υψηλών διαστάσεων.
- **Έλλειψη του καθολικά καλύτερου k:** Σε αντίθεση με άλλες υπερπαραμέτρους, το k δεν έχει μια καθολικά βέλτιστη τιμή, απαιτώντας από τους επαγγελματίες να δοκιμάζουν διάφορες ρυθμίσεις για κάθε συγκεκριμένο σύνολο δεδομένων. Αυτή η μέθοδος δοκιμής και σφάλματος είναι εγγενώς χρονοβόρα, καθώς απαιτεί την εξέταση της σχέσης μεταξύ k, μεγέθους

δεδομένων, κατανομής χαρακτηριστικών και πολυπλοκότητας εργασιών. Για παράδειγμα, ένα μικρό k μπορεί να υπερέχει σε πυκνά, χαμηλής διάστασης σύνολα δεδομένων, ενώ ένα μεγαλύτερο k είναι καταλληλότερο για αραιές ή πιο ομοιόμορφες κατανομές δεδομένων. Αυτή η διαφοροποίηση απαιτεί δοκιμή και επαλήθευση για κάθε διαμόρφωση.

- **Ευαισθησία κατανομής δεδομένων:** Τα μη ισορροπημένα ή ανομοιογενή σύνολα δεδομένων περιπλέκουν τη βελτιστοποίηση k, διαστρεβλώνοντας τα αποτελέσματα. Ο θόρυβος, οι ακραίες τιμές ή οι επικρατούσες κλάσεις μπορούν να μειώσουν την απόδοση του k, καθιστώντας αναγκαία διάφορα μέτρα προεπεξεργασίας, όπως η μείωση της διαστατικότητας, η επανεξισορρόπηση ή η κανονικοποίηση (normalization). Αυτά τα πρόσθετα βήματα αυξάνουν το χρόνο και τη συνολική πολυπλοκότητα που απαιτείται για την προσαρμογή του k. Σε ορισμένες περιπτώσεις, μπορεί να είναι απαραίτητη η επαναληπτική βελτίωση του συνόλου δεδομένων και των παραμέτρων για την επίτευξη αξιόπιστων αποτελεσμάτων, γεγονός που περιπλέκει περαιτέρω το πρόβλημα συνολικά.

2.3.2 Πείραμα ρύθμισης παραμέτρου k

Ένα σημαντικό στάδιο στη κατηγοριοποίηση είναι η αξιολόγηση της απόδοσης (performance evaluation) της. Ουσιαστικά, αξιολογεί την ακρίβεια (Accuracy) και την αποτελεσματικότητα του κατηγοριοποιητή, συμβάλλοντας στους ερευνητές στο να προσδιορίσουν την αποδοτικότητα ενός μοντέλου [23].

Μια από τις μεθόδους αξιολόγησης της απόδησης είναι η χρήση πίνακα σύγχυσης (confusion matrix). Ο συγκεκριμένος πίνακας συγκρίνει προβλεπόμενες και πραγματικές κατηγοριοποιήσεις. Επίσης, απεικονίζεται στο Σχήμα 2.2 [25].

		Actual	
		Positive	Negative
Predicted	Positive	True Positive	False Positive
	Negative	False Negative	True Negative

Figure 2-2 Διάγραμμα ροής της κατηγοριοποίησης με χρήση K-NN

Όπως μπορούμε να παρατηρήσουμε από το παραπάνω πίνακα, εμφανίζονται 4 αποτελέσματα:

- **Αληθώς Θετικό (True Positive, TP)**, όπου τα δεδομένα είναι ορθά ταξινομημένα ως θετικά.
- **Ψευδώς Θετικό (False Positive, FP)**, όπου τα δεδομένα είναι σφαλμένα ταξινομημένα ως θετικά.
- **Ψευδώς Αρνητικό (False Negative, FN)**, όπου τα δεδομένα είναι σφαλμένα ταξινομημένα ως αρνητικά.
- **Αληθώς Αρνητικό (True Negative, TN)**, όπου τα δεδομένα είναι ορθά ταξινομημένα ως αρνητικά.

Για να γνωρίζουμε τη τιμή της αξιολόγησης της απόδοσης, είναι απαραίτητος ο υπολογισμός της ακρίβειας. Η ακρίβεια μετράει το πόσο καλά ένα μοντέλο κατηγοριοποίησης προβλέπει ορθά και τις θετικές, αλλά και τις αρνητικές περιπτώσεις σε σχέση με το συνολικό αριθμό περιπτώσεων. Ο μαθηματικός τύπος φαίνεται παρακάτω [25].

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

Πραγματοποίησα πείραμα όπου εκτέλεσα τον k-NN για τα σύνολα δεδομένων magic.csv και letter.csv, τα οποία περιέχουν 19.020 και 20.000 δεδομένα αντίστοιχα, για όλες τις προαναφερόμενες μετρικές απόστασης (Ευκλείδεια, Manhattan, Chenyshev και Minkowski (p=3 και p=4)), και όλα τα γνωρίσματα (features) και την αντίστοιχη κλάση (class), για να παρατηρήσω τη σχέση ανάμεσα σε k και ακρίβειας. Οι τιμές του k είναι από 1 μέχρι και 50. Πριν την εκτέλεση του κατηγοριοποιητή, πραγματοποιήθηκε διαχωρισμός των συνόλων σε σύνολα εκπαίδευσης (70%) και δοκιμής (30%). Υπάρχει η επιλογή μετά το διαχωρισμό, να εφαρμοστεί στο σύνολο εκπαίδευσης (δηλαδή στο 70%) στρωματοποιημένη δειγματοληψία (stratified sampling) σε 1.000 δείγματα, όπου εξασφαλίζεται το γεγονός ότι διάφορες κατηγορίες αντιπροσωπεύονται ορθά κατά την εκπαίδευση του μοντέλου. Έτσι, αποφεύγεται η ανισορροπία κλάσεων, όπου η αξιολόγηση απόδοσης είναι πιο αξιόπιστη. Για το πείραμα, για κάθε σύνολο δεδομένων, εκτελώ μια περίπτωση όπου εφαρμόζεται στρωματοποιημένη δειγματοληψία και μια περίπτωση όπου δεν εφαρμόζται. Τα σύνολα δεδομένων φαίνονται στα Σχήματα 2.3 και 2.4.

```
1   FLength,FWidth,FSize,FConc,FConc1,FAsym,FM3Long,FM3Trans,FAlpha,FDist,Class
2   28.7967,16.0021,2.6449,0.3918,0.1982,27.7004,22.011,-8.2027,40.092,81.8828,g
3   31.6036,11.7235,2.5185,0.5303,0.3773,26.2722,23.8238,-9.9574,6.3609,205.261,g
4   162.052,136.031,4.0612,0.0374,0.0187,116.741,-64.858,-45.216,76.96,256.788,g
5   23.8172,9.5728,2.3385,0.6147,0.3922,27.2107,-6.4633,-7.1513,10.449,116.737,g
6   75.1362,30.9205,3.1611,0.3168,0.1832,-5.5277,28.5525,21.8393,4.648,356.462,g
7   51.624,21.1502,2.9085,0.242,0.134,50.8761,43.1887,9.8145,3.613,238.098,g
8   48.2468,17.3565,3.0332,0.2529,0.1515,8.573,38.0957,10.5868,4.792,219.087,g
9   26.7897,13.7595,2.5521,0.4236,0.2174,29.6339,20.456,-2.9292,0.812,237.134,g
10  96.2327,46.5165,4.154,0.0779,0.039,110.355,85.0486,43.1844,4.854,248.226,g
```

Figure 2-3 magic.csv

```
1   X-box,Y-box,Width,High,Onpix,X-bar,Y-bar,X2bar,Y2bar,Xybar,X2ybr,Xy2br,X-ege,Xegvy,Y-ege,Yegvx,Class
2   2,4,4,3,2,7,8,2,9,11,7,7,1,8,5,6,Z
3   4,7,5,5,5,5,9,6,4,8,7,9,2,9,7,10,P
4   7,10,8,7,4,8,8,5,10,11,2,8,2,5,5,10,S
5   4,9,5,7,4,7,7,13,1,7,6,8,3,8,0,8,H
6   6,7,8,5,4,7,6,3,7,10,7,9,3,8,3,7,H
7   4,7,5,5,3,4,12,2,5,13,7,5,1,10,1,7,F
8   6,10,8,8,4,7,8,2,5,10,7,8,5,8,1,8,N
9   1,0,2,0,1,6,10,7,2,7,5,8,2,7,4,9,R
10  5,9,7,6,7,7,7,2,4,9,8,9,7,6,2,8,M
```

Figure 2-4 letter.csv

Στα παρακάτω Σχήματα (2.5 με 2.8), φαίνονται τα διαγράμματα της σχέσης k με ακρίβειας μετά την εκτέλεση του k-NN για τα σύνολα δεδομένων magic και letter. Κάθε γραμμή αντιπροσωπεύει τις προαναφερόμενες μετρικές αποστάσεις, και για κάθε σύνολο δεδομένων υπάρχουν 2 διαγράμματα: ένα που εφαρμόζει στρωματοποιημένη δειγματοληψία κι ένα που δεν εφαρμόζει στρωματοποιημένη δειγματοληψία.

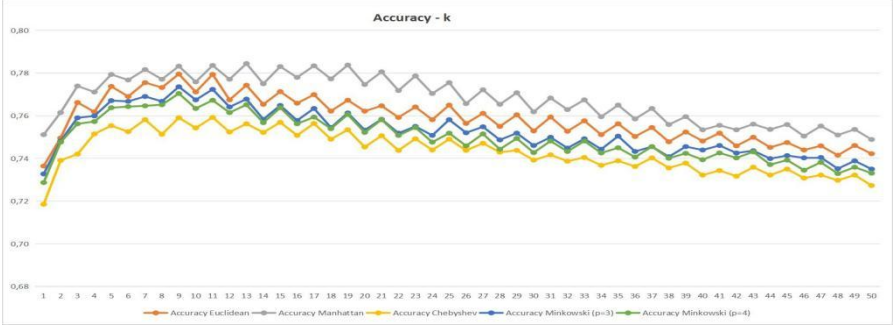


Figure 2-5 Γράφημα Accuracy - k για magic.csv κι όταν εφαρμόζεται στρωματοποιημένη δειγματοληψία

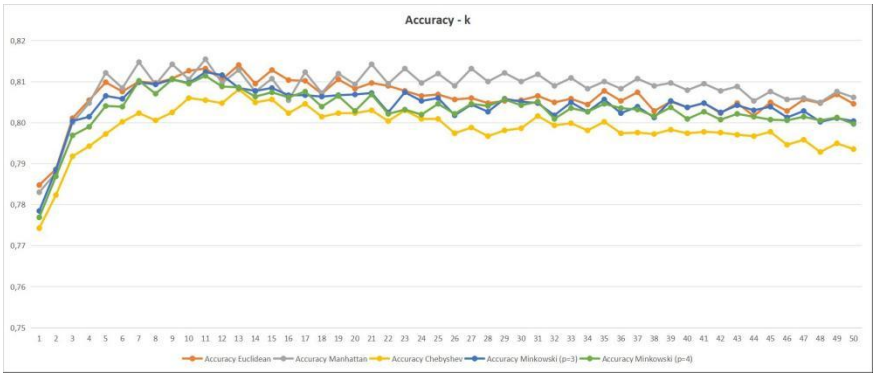


Figure 2-6 Γράφημα Accuracy - k για magic.csv κι όταν δεν εφαρμόζεται στρωματοποιημένη δειγματοληψία

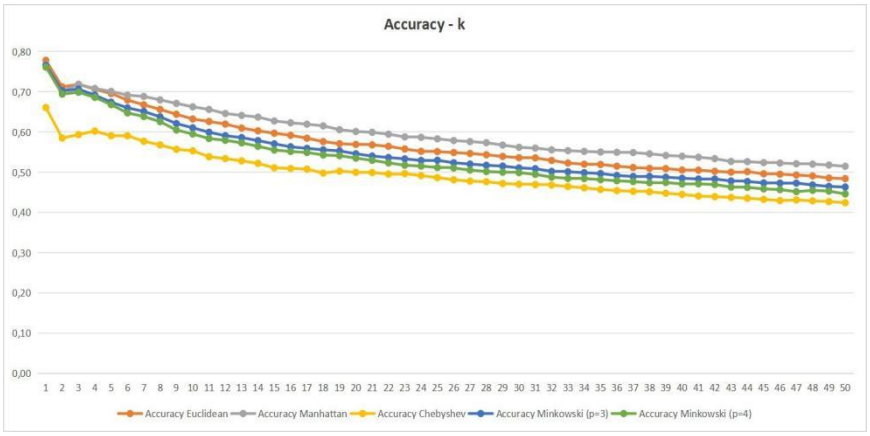


Figure 2-7 Γράφημα Accuracy - k για letter.csv κι όταν εφαρμόζεται στρωματοποιημένη δειγματοληψία

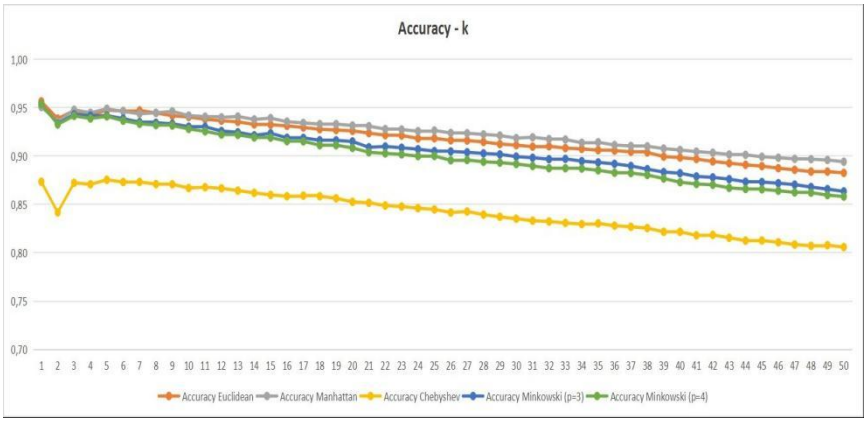


Figure 2-8 Γράφημα Accuracy - k για letter.csv κι όταν δεν εφαρμόζεται στρωματοποιημένη δειγματοληψία

Κεφάλαιο 3ο: Τεχνολογίες

3.1 Εισαγωγή

Για την ανάπτυξη μιας εφαρμογής, από προγραμματιστικής άποψης, εμπλέκονται δύο επίπεδα: το Back-end και το Front-end. Το επίπεδο Back-end, είναι εκείνο που αφορά τη πλευρά του διακομιστή, δεν είναι ορατό από το χρήστη και ο ρόλος του είναι η λειτουργικότητα της εφαρμογής. Από την άλλη, το Front-end αφορά τη πλευρά του περιηγητή (web browser), το οποίο όχι μόνο είναι ορατό από το χρήστη, αλλά αλληλεπιδράει με αυτό μέσω της γραφικής διεπαφής χρήστη. Παρακάτω αναλύονται οι γλώσσες προγραμματισμού και γενικά οι τεχνολογίες που εμπλέκονται στα προαναφερόμενα επίπεδα για την ανάπτυξη της εφαρμογής.

3.2 Back-end

3.2.1 PHP

Η PHP (Hypertext Preprocessor) είναι μια ευρέως χρησιμοποιούμενη γλώσσα σεναρίων (script) από την πλευρά του διακομιστή που δημιουργήθηκε κυρίως για την ανάπτυξη ιστοσελίδων, αν και είναι επίσης κατάλληλη για προγραμματισμό γενικού σκοπού. Ως μία από τις πιο ευρέως χρησιμοποιούμενες γλώσσες προγραμματισμού ιστού, η PHP είναι η κινητήρια δύναμη πάνω από το 80% των ιστότοπων παγκοσμίως, συμπεριλαμβανομένων εξέχουσων πλατφορμών όπως το WordPress, το Drupal και το phpBB [26]. Είναι μια διερμηνευμένη γλώσσα ανοικτού κώδικα που επιτρέπει στους προγραμματιστές να εισάγουν κώδικα στην HTML για την αποτελεσματική δημιουργία δυναμικών ιστοσελίδων. Η PHP φιλοξενεί διάφορες βάσεις δεδομένων, όπως η MySQL και η PostgreSQL, και έχει αναπτυχθεί σημαντικά με την πάροδο των ετών, ενσωματώνοντας έννοιες αντικειμενοστραφούς προγραμματισμού (Object-Oriented Programming - OOP) για τη βελτίωση της συντηρησιμότητας και της ασφάλειας [27]. Η γλώσσα έχει εξελιχθεί μέσω πλαισίων όπως το Symfony, το Laravel και το CodeIgniter, τα οποία παρέχουν δομημένες προσεγγίσεις ανάπτυξης, βελτιωμένη απόδοση και μεγαλύτερη ασφάλεια. Ακόμη και με τις ανησυχίες σχετικά με την επεκτασιμότητά της για μεγαλύτερες εφαρμογές, η PHP παραμένει μια σημαντική δύναμη στην ανάπτυξη ιστοσελίδων, εισάγοντας συνεχώς νέα χαρακτηριστικά και βελτιώνοντας την ασφάλεια.

Η PHP διαθέτει αρκετά βασικά χαρακτηριστικά που την καθιστούν μια από τις πιο δημοφιλείς γλώσσες για την ανάπτυξη ιστοσελίδων:

- **Επεξεργασία από την πλευρά του διακομιστή:** Η PHP λειτουργεί στον διακομιστή, εκτελώντας κώδικα πριν από τη μετάδοση της τελικής εξόδου HTML στον πελάτη, επιτρέποντας την ανάπτυξη δυναμικών ιστοσελίδων.
- **Ανοιχτού κώδικα:** Η PHP είναι προσβάσιμη δωρεάν και διαθέτει μια μεγάλη κοινότητα προγραμματιστών που βοηθούν στη συνεχή βελτίωσή της και στις ενημερώσεις ασφαλείας.
- **Συμβατότητα μεταξύ διαφορετικών πλατφορμών:** Η PHP λειτουργεί σε διάφορα λειτουργικά συστήματα (Windows, Linux, macOS) και είναι συμβατή με διάφορους διακομιστές ιστού (web servers), όπως ο Apache και ο Nginx.
- **Ενσωμάτωση βάσης δεδομένων:** Η PHP υποστηρίζει εγγενώς βάσεις δεδομένων όπως η MySQL, η PostgreSQL και η SQLite, γεγονός που την καθιστά ιδανική για διαδικτυακές εφαρμογές που επικεντρώνονται σε δεδομένα.
- **Δυναμική τυποποίηση:** Οι μεταβλητές PHP δεν χρειάζονται ειδικές δηλώσεις τύπου και μπορούν να αλλάζουν τύπο κατά την εκτέλεση, αυξάνοντας την ευελιξία, αλλά ενδεχομένως προκαλώντας προβλήματα εντοπισμού σφαλμάτων.

- **OOP:** Η σύγχρονη PHP ενσωματώνει τις αρχές της OOP, βελτιώνοντας τη δομή του κώδικα, τη συντήρηση και την ασφάλεια.
- **Ολοκληρωμένο πλαίσιο βοήθειας:** Γνωστά πλαίσια PHP, όπως το Laravel, το Symfony και το CodeIgniter, παρέχουν οργανωμένη ανάπτυξη, καλύτερη ασφάλεια και αυξημένες επιδόσεις.
- **Συνεχής ανάπτυξη:** Η PHP έχει σημειώσει σημαντικές εξελίξεις όλα αυτά τα χρόνια, δίνοντας προτεραιότητα στη βελτίωση της απόδοσης, την ενίσχυση της ασφάλειας και την αρθρωτότητα για να φιλοξενήσει τις σύγχρονες διαδικτυακές εφαρμογές.

Ο κώδικας PHP γενικά περικλείεται μέσα σε ετικέτες PHP (<?php ... ?>), οι οποίες υποδηλώνουν την αρχή και το τέλος της PHP σεναρίου μέσα σε ένα αρχείο HTML. Αυτό επιτρέπει στους προγραμματιστές να συνδυάζουν εύκολα την PHP με την HTML για δυναμικό περιεχόμενο ιστού. Τα αρχεία PHP χρησιμοποιούν την επέκταση αρχείου '.php', η οποία εγγυάται ότι ο διακομιστής χειρίζεται το αρχείο ως PHP αντί για κανονική HTML. Στην PHP, οι μεταβλητές ορίζονται με το σύμβολο $ (π.χ., $name = "John";) και δεν υπάρχει ανάγκη για ρητές δηλώσεις τύπου, καθώς η PHP είναι μια γλώσσα δυναμικής τυποποίησης. Οι εντολές της PHP συνήθως ολοκληρώνονται με ερωτηματικό (;), όπως και σε άλλες γλώσσες που βασίζονται στη C, γεγονός που εγγυάται την ορθή εκτέλεση του κώδικα. Οι συμβολοσειρές (strings) μπορούν να δημιουργηθούν χρησιμοποιώντας είτε απλά (') είτε διπλά (") εισαγωγικά, όπου τα διπλά εισαγωγικά επιτρέπουν την παρεμβολή μεταβλητών. Η PHP προσφέρει επίσης ενσωματωμένες συναρτήσεις και δομές ελέγχου (όπως οι if, for και foreach) για την αποτελεσματική διαχείριση της λογικής και των βρόχων [27]. Επιπλέον, η PHP επιτρέπει τη χρήση υπερ-παγκόσμιων πινάκων, όπως οι $_GET, $_POST και $_SESSION, για τη διαχείριση της εισόδου του χρήστη, την επεξεργασία των υποβολών φορμών και τη διαχείριση των συνόδων [26]. Χάρη στην απλή σύνταξη και το ευπροσάρμοστο πλαίσιο, η PHP προσφέρει μια στιβαρή και φιλική προς το χρήστη προσέγγιση για τη δημιουργία εφαρμογών ιστού.

Ο Πίνακας 3.1 παρακάτω, απεικονίζει ένα παράδειγμα σύνταξης κώδικα PHP.

```php
<?php
    // Start session to access session variables
    session_start();

    // Include database connection
    require_once "../db_connection.php";

    header('Content-Type: application/json');

    // Ensure the request is a POST request
    if ($_SERVER['REQUEST_METHOD'] !== 'POST') {
        http_response_code(405);
        echo json_encode(["status" => "danger", "message" => "Only POST requests are allowed"]);
        exit;
    }

    // Retrieve the POST data
    $data = json_decode(file_get_contents("php://input"), true);

```

```
20      // Check if required parameters are present
21      if (!isset($data['email']) || !isset($data['password'])) {
22          http_response_code(400);
23          echo json_encode(["status" => "danger", "message" => "Missing required parameters"]);
24          exit;
25      }
26
27      $email = $data['email'];
28      $password = $data['password'];
29
30      // Rest of code
31  ?>
```

Table 3-1 Παράδειγμα κώδικα PHP

3.2.2 Python

Η Python είναι μια υψηλού επιπέδου, διερμηνευμένη γλώσσα προγραμματισμού που αναγνωρίζεται για την ευκολία χρήσης, τη σαφήνεια και την ευελιξία της. Χρησιμοποιείται συνήθως σε πολλούς τομείς, όπως η ανάπτυξη ιστοσελίδων, η επιστήμη των δεδομένων, η τεχνητή νοημοσύνη και οι επιστημονικοί υπολογισμοί, λόγω της ολοκληρωμένης τυποποιημένης βιβλιοθήκης της και της ισχυρής υποστήριξης της κοινότητας [28]. Το συντακτικό της Python έχει διαμορφωθεί έτσι ώστε να είναι απλό, καταργώντας την απαίτηση για σαφείς δηλώσεις μεταβλητών και άνω και κάτω τελεία, και αντ' αυτού βασιζόμενο στην εσοχή για την οργάνωση του κώδικα [29]. Η Python, ως γλώσσα ανοικτού κώδικα, ευνοεί τη συνεργασία και τη δημιουργικότητα, γεγονός που την καθιστά ευνοϊκή επιλογή τόσο για τους νεοεισερχόμενους όσο και για τους έμπειρους προγραμματιστές.

Τα κύρια χαρακτηριστικά τα οποία η Python κατέχει, είναι τα εξής:

- **Σαφήνεια και ευκολία:** Η σύνταξη της Python είναι απλή και συνοπτική, γεγονός που διευκολύνει την ανάγνωση και τη συγγραφή.
- **Διερμηνευόμενη γλώσσα:** Η Python δεν χρειάζεται μεταγλώττιση- ο διερμηνέας εκτελεί τον κώδικα διαδοχικά.
- **Δυναμική τυποποίηση:** Η ρητή δήλωση μεταβλητών είναι περιττή- η Python προσδιορίζει τον τύπο κατά τη διάρκεια της εκτέλεσης.
- **Σύνταξη βασισμένη στην εσοχή:** Αντί να χρησιμοποιεί αγκύλες `{}` ή ερωτηματικό `;`, η Python χρησιμοποιεί εσοχές για να δημιουργήσει μπλοκ κώδικα.
- **Ολοκληρωμένη τυποποιημένη βιβλιοθήκη:** Η Python προσφέρει ολοκληρωμένες ενότητες για δραστηριότητες όπως η διαχείριση αρχείων, οι μαθηματικοί υπολογισμοί και η επεξεργασία δεδομένων.
- **Συμβατότητα σε διαφορετικές πλατφόρμες:** Η Python λειτουργεί σε διάφορα λειτουργικά συστήματα (Windows, macOS, Linux) χωρίς να χρειάζεται τροποποιήσεις.
- **Αντικειμενοστρεφής και λειτουργική:** Η Python υποστηρίζει τόσο τον αντικειμενοστραφή προγραμματισμό (OOP) όσο και τις προσεγγίσεις του λειτουργικού προγραμματισμού.
- **Τεράστια κοινότητα και ανοιχτός κώδικας:** Η Python μπορεί να υπερηφανεύεται για τη δυναμική της κοινότητα που προσφέρει ολοκληρωμένη τεκμηρίωση, πλαίσια και βιβλιοθήκες.

Η Python έχει σχεδιαστεί για να είναι μια εύκολα κατανοητή και προσαρμόσιμη γλώσσα προγραμματισμού, με ένα απλό και ευανάγνωστο συντακτικό που εξαλείφει την ανάγκη για άνω και κάτω τελεία (`;`) και χρησιμοποιεί εσοχές για να οριοθετεί τα τμήματα κώδικα. Στην Python, οι

μεταβλητές είναι δυναμικά τυποποιημένες, πράγμα που σημαίνει ότι δεν απαιτούν ρητή δήλωση και ο τύπος τους προκύπτει αυτόματα κατά την εκτέλεση. Δομές ελέγχου όπως οι `if`, `elif` και `else` διευκολύνουν τη λήψη αποφάσεων κατά την εκτέλεση του κώδικα, επιτρέποντας λειτουργίες που βασίζονται σε λογικές εκφράσεις. Η Python προσφέρει επίσης δομές βρόχων όπως οι βρόχοι `for` που διατρέχουν ακολουθίες όπως λίστες και πλειάδες (tuples), καθώς και βρόχους `while` που συνεχίζουν να εκτελούνται όσο μια δεδομένη συνθήκη ισχύει [29]. Επιπλέον, η Python υποστηρίζει τόσο τον αντικειμενοστραφή προγραμματισμό (OOP) όσο και το λειτουργικό στυλ προγραμματισμού, καθιστώντας την πολύ ευέλικτη για μια σειρά από χρήσεις, όπως οι επιστημονικοί υπολογισμοί και η ανάλυση δεδομένων [28].

Στον Πίνακα 3.2 απεικονίζεται ένα παράδειγμα σύνταξης κώδικα Python.

```python
# Check if the class column is provided and valid
if class_name != 'None' and class_name in dataset.columns:
    # Calculate metrics
    class_label = dataset[class_name]
    labels = class_label.unique().tolist()  # Get unique labels

    # Calculate accuracy
    accuracy = round(metrics.accuracy_score(class_label, predicted_values), 2)

    # Calculate precision, recall, and f1-score per label
    precision_per_label, recall_per_label, fscore_per_label, _ = metrics.precision_recall_fscore_support(
        class_label, predicted_values, average=None, labels=labels, zero_division=0
    )
    precision_per_label = [round(p, 2) for p in precision_per_label]
    recall_per_label = [round(r, 2) for r in recall_per_label]
    fscore_per_label = [round(f, 2) for f in fscore_per_label]

    # Calculate average precision, recall, and f1-score
    average_precision, average_recall, average_fscore, _ = metrics.precision_recall_fscore_support(
        class_label, predicted_values, average='macro', zero_division=0
    )
    average_precision, average_recall, average_fscore = (
        round(average_precision, 2),
        round(average_recall, 2),
        round(average_fscore, 2),
    )

    # Rest of code
```

Table 3-2 Παράδειγμα κώδικα Python

3.2.3 MySQL

Η MySQL είναι ένα δημοφιλές σύστημα διαχείρισης σχεσιακών βάσεων δεδομένων (Relational Database Management System - RDBMS) ανοικτού κώδικα που διευκολύνει την αποτελεσματική αποθήκευση, ανάκτηση και διαχείριση δεδομένων μέσω της δομημένης γλώσσας ερωτήσεων

(Structured Query Language - SQL). Η MySQL, που δημιουργήθηκε και εποπτεύεται από την Oracle Corporation, αναγνωρίζεται για τις εξαιρετικές επιδόσεις, την προσαρμοστικότητα και την επεκτασιμότητά της, καθιστώντας την ευνοημένη επιλογή για διαδικτυακές εφαρμογές, επιχειρηματικές λύσεις και συστήματα με επίκεντρο τα δεδομένα [30]. Η MySQL είναι απαραίτητη για την επεξεργασία και βελτιστοποίηση ερωτημάτων, ιδίως σε εφαρμογές που χρειάζονται δυναμικές συστάσεις ερωτημάτων, όπως φαίνεται σε μια συγκριτική ανάλυση της αντιστοίχισης προτύπων MySQL και της ομοιότητας Jaccard. Η έρευνα τονίζει την αποτελεσματικότητα της MySQL στη διαχείριση των συστάσεων ερωτημάτων, καθώς η αντιστοίχιση προτύπων αποδίδει ταχύτερα αποτελέσματα αναζήτησης, ενώ η ομοιότητα Jaccard βελτιώνει την ακρίβεια στην απόκτηση σχετικών πληροφοριών [31]. Αυτά τα χαρακτηριστικά καθιστούν τη MySQL ένα αποτελεσματικό μέσο για το χειρισμό δομημένων δεδομένων σε πολλαπλά πεδία.

Τα κύρια χαρακτηριστικά της MySQL είναι τα παρακάτω:

- **Δωρεάν & Ανοιχτού Κώδικα:** Η MySQL είναι ένα RDBMS ανοικτού κώδικα, το οποίο έχει αδειοδοτηθεί με τη Γενική Άδεια Δημόσιας Χρήσης GNU (General Public License - GPL), επιτρέποντας στους προγραμματιστές να το χρησιμοποιούν, να το τροποποιούν και να το μοιράζονται χωρίς κόστος.
- **Σχεσιακό μοντέλο βάσης δεδομένων:** Δομεί τα δεδομένα σε οργανωμένους πίνακες με γραμμές και στήλες, διευκολύνοντας την αποτελεσματική διαχείριση και ανάκτηση πληροφοριών.
- **Ενισχυμένη απόδοση:** Η MySQL είναι ρυθμισμένη για ταχεία επεξεργασία, γεγονός που την καθιστά ιδανική για τη διαχείριση εκτεταμένων συνόλων δεδομένων και εφαρμογών με μεγάλη επισκεψιμότητα.
- **Συμμόρφωση ACID (Atomicity Consistency Isolation Durability):** Διατηρεί τις συναλλαγές Ατομικότητας, Συνέπειας, Απομόνωσης και Ανθεκτικότητας, εγγυώμενη την αξιοπιστία και την ακεραιότητα των δεδομένων.
- **Βασισμένο σε SQL:** Η MySQL χρησιμοποιεί τη SQL για την εκτέλεση λειτουργιών της βάσης δεδομένων, όπως `SELECT`, `INSERT`, `UPDATE` και `DELETE`.
- **Βελτιστοποίηση ερωτήσεων:** Περιλαμβάνει ολοκληρωμένες μεθόδους βελτιστοποίησης, όπως η ευρετηρίαση και η προσωρινή αποθήκευση, για τη βελτίωση της αποτελεσματικότητας της αναζήτησης και της ανάκτησης.
- **Δυνατότητες για αντιστοίχιση μοτίβων:** Η αντιστοίχιση μοτίβων της MySQL, που χρησιμοποιείται συχνά στις συστάσεις ερωτημάτων, ενισχύει την ακρίβεια αναζήτησης και την ταχύτητα επεξεργασίας, όπως προκύπτει από τις αντιπαραθέσεις με την ομοιότητα Jaccard.
- **Επεκτασιμότητα & Ευελιξία:** Η MySQL είναι ικανή να διαχειρίζεται τόσο μικρές εφαρμογές όσο και μεγάλα εταιρικά συστήματα, επιτρέποντας την αντιγραφή, την ομαδοποίηση και την ανάπτυξη στο σύννεφο.

Στη MySQL, τα δεδομένα είναι δομημένα σε πίνακες, που αποτελούνται από γραμμές (εγγραφές) και στήλες (πεδία) για κάθε πίνακα. Οι σειρές περιέχουν τις ξεχωριστές καταχωρήσεις δεδομένων που ευθυγραμμίζονται με τα χαρακτηριστικά που καθορίζονται από τις στήλες, συμπεριλαμβανομένων των ονομάτων, των ημερομηνιών και των αριθμητικών στοιχείων. Οι χρήστες μπορούν να ασχοληθούν αποτελεσματικά με αυτούς τους πίνακες μέσω εντολών SQL. Η εντολή `SELECT` αντλεί συγκεκριμένα δεδομένα από έναν πίνακα σύμφωνα με όρους, γεγονός που την καθιστά ζωτικής σημασίας για τα ερωτήματα σε βάσεις δεδομένων. Η εντολή `INSERT` προσθέτει νέες γραμμές δεδομένων, ενώ η εντολή `UPDATE` αλλάζει τις τρέχουσες εγγραφές, διασφαλίζοντας ότι οι πληροφορίες παραμένουν σωστές και τρέχουσες. Η εντολή `DELETE` επιτρέπει στους χρήστες να εξαλείψουν συγκεκριμένες γραμμές από έναν πίνακα όταν τα δεδομένα δεν είναι πλέον απαραίτητα [30]. Αυτές οι βασικές λειτουργίες καθιστούν τη MySQL ένα ισχυρό και ευέλικτο σύστημα διαχείρισης βάσεων δεδομένων για τη διαχείριση δομημένων δεδομένων σε πολυάριθμες εφαρμογές.

Στον Πίνακα 3.3 απεικονίζεται ένα παράδειγμα σύνταξης κώδικα SQL.

```
DROP TABLE IF EXISTS `verify_account`;
CREATE TABLE IF NOT EXISTS `verify_account` (
    `id_of_user` int(11) NOT NULL,
    `verification_key` varchar(100) NOT NULL,
    `creation_time` timestamp NULL DEFAULT current_timestamp(),
    PRIMARY KEY (`id_of_user`) USING BTREE,
    CONSTRAINT `FK_verify_account_users` FOREIGN KEY (`id_of_user`) REFERENCES `users` (`id`)
        ON DELETE CASCADE
        ON UPDATE NO ACTION
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_bin;
```

Table 3-3 Παράδειγμα κώδικα SQL

3.2.4 Scikit-Learn

Το Scikit-learn είναι μια εκτεταμένη βιβλιοθήκη Python ανοικτού κώδικα για μηχανική μάθηση που προσφέρει πολυάριθμα εργαλεία για την ανάλυση και μοντελοποίηση δεδομένων. Στόχος του είναι να βελτιώσει τις εργασίες μηχανικής μάθησης παρέχοντας αποτελεσματικές υλοποιήσεις διαφορετικών αλγορίθμων τόσο για επιβλεπόμενη όσο και για μη επιβλεπόμενη μάθηση, μαζί με την προεπεξεργασία, την αξιολόγηση και την επιλογή μοντέλων. Βασισμένο σε βασικές βιβλιοθήκες επιστημονικών υπολογιστών, όπως οι NumPy και SciPy, το Scikit-learn εγγυάται αυξημένες επιδόσεις, διατηρώντας παράλληλα τη φιλικότητα προς το χρήστη μέσω ενός συνεκτικού API και συνεπών μορφών δεδομένων [32]. Η ισχυρή υποστήριξη της κοινότητας, η εκτενής τεκμηρίωση και η εύκολη συμβατότητα με άλλες βιβλιοθήκες Python την καθιστούν μια ισχυρή και προσιτή εργαλειοθήκη για ερευνητές, επιστήμονες δεδομένων και προγραμματιστές.

Το Scikit-learn διακρίνεται για μια σειρά σημαντικών χαρακτηριστικών που το καθιστούν μια ευέλικτη και αποτελεσματική βιβλιοθήκη μηχανικής μάθησης. Προσφέρει μια εκτεταμένη σειρά αλγορίθμων μηχανικής μάθησης, με μεθόδους ταξινόμησης, παλινδρόμησης, ομαδοποίησης και μείωσης διαστάσεων. Η βιβλιοθήκη είναι σχεδιασμένη για βέλτιστη απόδοση, χρησιμοποιώντας μεταγλωττισμένες γλώσσες όπως η C και η Fortran για αποδοτικότητα, ενώ παρέχει μια εύχρηστη διεπαφή Python. Το Scikit-learn διατηρεί έναν ομοιόμορφο σχεδιασμό API, επιτρέποντας την εύκολη μετάβαση μεταξύ διαφόρων μοντέλων με ελάχιστες αλλαγές στον κώδικα. Επιπλέον, περιλαμβάνει ισχυρά εργαλεία για την προεπεξεργασία δεδομένων, την επιλογή χαρακτηριστικών και την αξιολόγηση μοντέλων, συμπεριλαμβανομένης της διασταυρούμενης επικύρωσης και της βελτιστοποίησης υπερπαραμέτρων [32]. Λόγω της ισχυρής υποστήριξης της κοινότητας, της ολοκληρωμένης τεκμηρίωσης και της ομαλής ενσωμάτωσης με άλλες βιβλιοθήκες Python, όπως οι NumPy και Pandas, το Scikit-learn χρησιμοποιείται ευρέως τόσο για ακαδημαϊκή έρευνα όσο και για εφαρμογές στον πραγματικό κόσμο.

Στον Πίνακα 3.4 απεικονίζεται η αξιοποίηση της βιβλιοθήκης Scikit-learn για την εκτέλεση του κατηγοριοποιητή k-NN και της μεθόδου διαχωρισμού train-test.

```python
from sklearn.model_selection import train_test_split
from sklearn.neighbors import KNeighborsClassifier
from sklearn.metrics import accuracy_score, precision_score, recall_score, f1_score, classification_report

def evaluate_knn(X_train, y_train, X_test, y_test, k, distance, p=None):
    clf = KNeighborsClassifier(n_neighbors=k, metric=distance, p=p)
    clf.fit(X_train, y_train)
    y_pred = clf.predict(X_test)

    # Overall metrics
    accuracy = accuracy_score(y_test, y_pred)
    precision = precision_score(y_test, y_pred, average='weighted', zero_division=0)
    recall = recall_score(y_test, y_pred, average='weighted', zero_division=0)
    f1 = f1_score(y_test, y_pred, average='weighted', zero_division=0)

    # Class-wise metrics
    class_report = classification_report(y_test, y_pred, output_dict=True)

    return accuracy, precision, recall, f1, class_report

if len(sys.argv) != 9:
    print("Usage: python knn_train_test.py <file> <features> <target> <k_values> <distances> <p_value> <stratified_sampling> <results_file_path>")
    sys.exit(1)

# The script arguments passed from PHP
file = sys.argv[1]
features = sys.argv[2].split(",")  # Comma-separated string -> list
target = sys.argv[3]
k_values = [int(k) for k in sys.argv[4].split(",")]
distance_values = sys.argv[5].split(",")

p_value = sys.argv[6] if sys.argv[6] and sys.argv[6] != 'null' else None
if p_value:
    p_value = [int(p) for p in sys.argv[6].split(",")]

stratified_sampling = sys.argv[7].lower() == 'true'
results_path = sys.argv[8]

# Load dataset
dataset = pd.read_csv(file)

# Split the features and target variables
X = dataset[features].values
y = dataset[target].values

# Perform train test split (70% train, 30% test)
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3, random_state=42)

# Check if stratified sampling is needed
if stratified_sampling:
    X_sampled, _, y_sampled, _ = train_test_split(X_train, y_train, train_size=1000, random_state=42, stratify=y_train)
else:
    X_sampled, y_sampled = X_train, y_train
```

Table 3-4 Παράδειγμα κώδικα Python για ορισμό και χρήση βιβλιοθήκης Scikit-Learn

3.2.5 Composer

Το Composer είναι ένα βοηθητικό πρόγραμμα για την εποπτεία των εξαρτήσεων στην PHP που απλοποιεί τη διαχείριση των βιβλιοθηκών και των πακέτων που απαιτούνται για ένα έργο. Επιτρέπει στους προγραμματιστές να ορίζουν τις εξαρτήσεις του έργου τους μέσα σε ένα αρχείο `composer.json` που περιγράφει λεπτομερώς τα απαιτούμενα πακέτα και τις εκδόσεις τους. Ο συνθέτης συγκεντρώνει αυτές τις εξαρτήσεις από το Packagist, διασφαλίζοντας ότι όλα τα απαραίτητα συστατικά έχουν εγκατασταθεί και ενημερωθεί σωστά. Ένα βασικό χαρακτηριστικό της είναι η αυτόματη διαχείριση εξαρτήσεων, η οποία αποτρέπει τις συγκρούσεις εκδόσεων και τα προβλήματα συμβατότητας μεταξύ διαφορετικών βιβλιοθηκών [33]. Επιπλέον, το Composer λειτουργεί απρόσκοπτα με πλαίσια PHP όπως το Magento 2, βελτιώνοντας τη διαχείριση πακέτων και αυξάνοντας την αποτελεσματικότητα της ανάπτυξης.

Το Composer διαθέτει αρκετά σημαντικά χαρακτηριστικά που το καθιστούν ένα ζωτικής σημασίας εργαλείο για την ανάπτυξη PHP. Προσφέρει αποτελεσματική διαχείριση εξαρτήσεων, επιτρέποντας στους προγραμματιστές να ορίζουν τα απαραίτητα πακέτα στο αρχείο `composer.json`, τα οποία το Composer στη συνέχεια επιλύει και εγκαθιστά αυτόματα. Επιτρέπει τον έλεγχο εκδόσεων, διατηρώντας τη συμβατότητα μεταξύ των διαφόρων βιβλιοθηκών με το χειρισμό των εξαρτήσεων με βάση τη σημασιολογική έκδοση. Το Composer επιτρέπει επίσης λειτουργίες αυτόματης φόρτωσης}, διευκολύνοντας την ενσωμάτωση εξωτερικών βιβλιοθηκών χωρίς την ανάγκη χειροκίνητης ενσωμάτωσης αρχείων. Επιπλέον, συμμορφώνεται με τα Πρότυπα Διαλειτουργικότητας Πλαισίου PHP (PSR), προωθώντας την ομοιομορφία και τη συμβατότητα μεταξύ των έργων PHP [33]. Συνδεδεμένο με το Packagist, το Composer προσφέρει πρόσβαση σε ένα μεγάλο αποθετήριο πακέτων PHP, καθιστώντας το ένα ισχυρό και προσαρμόσιμο εργαλείο για το χειρισμό των εξαρτήσεων του έργου.

Η χρήση του Composer συνεπάγεται μια οργανωμένη προσέγγιση που βελτιώνει το χειρισμό εξαρτήσεων σε έργα PHP. Αρχικά, οι προγραμματιστές παραθέτουν τις απαραίτητες βιβλιοθήκες και εξαρτήσεις σε ένα αρχείο `composer.json`, αναφέροντας λεπτομερώς τα ονόματα των πακέτων και τους περιορισμούς των εκδόσεων. Ο συνθέτης αναζητά αυτά τα πακέτα στο Packagist, τον τυπικό κατάλογο πακέτων, και επιλύει όλες τις εξαρτήσεις για να εγγυηθεί τη συμβατότητα μεταξύ των διαφόρων βιβλιοθηκών. Αφού εντοπίσει τις εξαρτήσεις, το Composer ανακτά και εγκαθιστά τα απαραίτητα αρχεία τοπικά στον κατάλογο «vendor», διασφαλίζοντας ότι το έργο περιέχει όλα τα βασικά συστατικά [33]. Αυτή η μέθοδος εξασφαλίζει ότι οι προγραμματιστές χρησιμοποιούν τις κατάλληλες εκδόσεις βιβλιοθηκών και αποτρέπουν τις συγκρούσεις, ενισχύοντας έτσι την αποτελεσματικότητα και τη διαχειρισιμότητα της διαδικασίας ανάπτυξης.

Στον Πίνακα 3.5 παρουσιάζεται ένα παράδειγμα δήλωσης εξαρτήσεων, όπου πρώτα δημιουργείται το `composer.json` κι ύστερα πραγματοποιείται κλήση της εντολής `composer install`.

```
{
    "require": {
        "phpmailer/phpmailer": "^6.5"
    }
}
```

Table 3-5 Παράδειγμα ορισμού Composer

3.3 Front-end

3.3.1 HTML

Η HTML (HyperText Markup Language) είναι η τυπική γλώσσα κωδικοποίησης που χρησιμοποιείται για τη δημιουργία και το σχεδιασμό ιστοσελίδων. Οργανώνει τις πληροφορίες στο διαδίκτυο και γίνεται κατανοητό από τους φυλλομετρητές ιστού για την εμφάνιση των στοιχείων μιας ιστοσελίδας [34]. Η HTML είναι μια γλώσσα σήμανσης και όχι μια γλώσσα προγραμματισμού, η οποία καθορίζει την οργάνωση του περιεχομένου του διαδικτύου μέσω στοιχείων που περιβάλλονται από ετικέτες.

Τα κύρια χαρακτηριστικά της HTML είναι τα ακόλουθα:

- **Γλώσσα σήμανσης:** Η HTML χρησιμοποιεί ετικέτες για την οργάνωση του περιεχομένου του ιστού. Οι ετικέτες προσδιορίζουν στοιχεία όπως επικεφαλίδες, παραγράφους, συνδέσμους, εικόνες, πίνακες και άλλα.
- **Ανεξάρτητη από πλατφόρμες:** Ο κώδικας HTML μπορεί να αναπτυχθεί σε οποιοδήποτε λειτουργικό σύστημα και είναι συμβατός με όλους τους σύγχρονους φυλλομετρητές ιστού.
- **Στατικό και δυναμικό περιεχόμενο:** Η HTML είναι ικανή να παράγει στατικές ιστοσελίδες και, όταν συνδυάζεται με CSS (δείτε στην ενότητα 3.3.4 παρακάτω) και JavaScript, μπορεί να βελτιώσει τη διαδραστικότητα και την αισθητική.
- **Ιεραρχία και φωλιασμός:** Τα στοιχεία HTML μπορούν να ενσωματωθούν το ένα μέσα στο άλλο, καθορίζοντας τη διάταξη μιας ιστοσελίδας.
- **Υπερσύνδεση:** Η HTML επιτρέπει συνδέσεις μεταξύ διαφόρων ιστοσελίδων και εξωτερικών πόρων μέσω της χρήσης ετικετών άγκυρας.
- **Ενσωμάτωση πολυμέσων:** Η HTML επιτρέπει τη συμπερίληψη εικόνων, βίντεο και ηχητικών στοιχείων για τη βελτίωση της παρουσίασης του περιεχομένου.

3.3.2 JavaScript

Η JavaScript είναι μια ευέλικτη, προσανατολισμένη στο πρωτότυπο γλώσσα σεναρίων (scripting) που χρησιμοποιείται κυρίως για την ανάπτυξη ιστοσελίδων, προωθώντας διαδραστικές και ευέλικτες εμπειρίες χρήστη. Ξεκίνησε αρχικά από τη Netscape το 1995 ως μια φιλική προς το χρήστη γλώσσα σεναρίων αντικειμένων, και έκτοτε έχει γίνει το αποδεκτό πρότυπο για σεναριογράφηση από την πλευρά του πελάτη στους φυλλομετρητές ιστού. Η JavaScript διαθέτει ένα σύστημα ασθενών τύπων που επιτρέπει ευέλικτες μετατροπές τύπων, με ένα μοντέλο αντικειμένων που βασίζεται σε πρωτότυπα αντί για κλασική κληρονομικότητα. Διευκολύνει συναρτήσεις πρώτης κατηγορίας, επιτρέποντας λειτουργικά στυλ προγραμματισμού, και προσφέρει χαρακτηριστικά όπως προγραμματισμό με βάση τα γεγονότα, ασύγχρονες εργασίες και χειρισμό του Μοντέλου Αντικειμένου Εγγράφου (Document Object Model - DOM) για την ανάπτυξη εξελιγμένων εφαρμογών ιστού [35]. Αν και η JavaScript είναι ισχυρή, τα δυναμικά χαρακτηριστικά της δημιουργούν δυσκολίες στην ανάπτυξη, γεγονός που καθιστά τη στατική ανάλυση και τα εργαλεία ελέγχου τύπων απαραίτητα για την ανίχνευση σφαλμάτων και την ενίσχυση της αξιοπιστίας του κώδικα.

Η JavaScript διαθέτει αρκετά κρίσιμα χαρακτηριστικά που ενισχύουν την προσαρμοστικότητα και την ελκυστικότητά της ως γλώσσα προγραμματισμού. Είναι δυναμικά τυποποιημένη, δηλαδή οι μεταβλητές δεν απαιτούν ρητές δηλώσεις τύπου, και επιτρέπει την έμμεση μετατροπή τύπου, η οποία μπορεί μερικές φορές να οδηγήσει σε απροσδόκητες συμπεριφορές. Η JavaScript δίνει έμφαση στα πρωτότυπα, χρησιμοποιώντας αντικείμενα πρωτοτύπων για την κληρονομικότητα αντί των συμβατικών πλαισίων που βασίζονται σε κλάσεις. Θεωρεί τις συναρτήσεις ως αντικείμενα πρώτης κατηγορίας, επιτρέποντάς τους να ανατίθενται σε μεταβλητές, να περνούν ως παράμετροι και να επιστρέφονται από διαφορετικές συναρτήσεις. Επιπλέον, η JavaScript προσφέρει δυνατότητες

προγραμματισμού με βάση τα γεγονότα και ασύγχρονου προγραμματισμού, διευκολύνοντας τις δυναμικές εφαρμογές ιστού μέσω λειτουργιών όπως το AJAX (Asynchronous JavaScript και XML) και οι Promises. Το ευέλικτο πλαίσιο αντικειμένων του επιτρέπει τη δυναμική προσθήκη ή αφαίρεση ιδιοτήτων από αντικείμενα και περιλαμβάνει ενσωματωμένες μεθόδους για το χειρισμό απροσδιόριστων τιμών και εξαιρέσεων [35]. Παρόλα αυτά, αυτά τα δυναμικά χαρακτηριστικά δημιουργούν προκλήσεις στην ανάλυση προγραμμάτων, απαιτώντας εξειδικευμένα εργαλεία για να διασφαλιστεί η αξιοπιστία του κώδικα και να αποφευχθούν τα συνήθη λάθη.

Η JavaScript λειτουργεί σε συνδυασμό με την HTML και το CSS για τη δημιουργία δυναμικών, διαδραστικών ιστότοπων μέσω του χειρισμού του DOM. Με τη JavaScript, οι προγραμματιστές μπορούν να τροποποιούν στοιχεία HTML, να ενημερώνουν στυλ CSS και να ανανεώνουν περιεχόμενο χωρίς να απαιτείται πλήρης επαναφόρτωση της σελίδας. Αυτό επιτυγχάνεται μέσω του προγραμματισμού με βάση τα συμβάντα, όπου η JavaScript παρατηρεί τις δραστηριότητες του χρήστη, όπως τα κλικ, τις υποβολές φόρμας ή τις πληκτρολογήσεις, και ανταποκρίνεται αμέσως. Επιπλέον, τεχνολογίες όπως η AJAX και το Fetch API επιτρέπουν στη JavaScript να επικοινωνεί με έναν διακομιστή, να αντλεί ή να στέλνει δεδομένα παρασκηνιακά, ενώ παράλληλα ανανεώνει γρήγορα συγκεκριμένα τμήματα μιας ιστοσελίδας. Αυτή η λειτουργικότητα είναι ζωτικής σημασίας για εφαρμογές μίας σελίδας (Single Page Applications - SPA), όπου πλαίσια JavaScript όπως τα React, Angular και Vue φορτώνουν δυναμικά το περιεχόμενο, παρέχοντας μια απρόσκοπτη εμπειρία χρήστη [35]. Το άρθρο υπογραμμίζει ότι η προσαρμόσιμη αρχιτεκτονική της JavaScript, η οποία ορίζεται από την αδύναμη τυποποίηση, τη δομή που βασίζεται σε πρωτότυπα και ένα πλαίσιο δυναμικής εκτέλεσης, παρουσιάζει τόσο ευελιξία όσο και προκλήσεις στην αξιολόγηση τύπων και την αξιοπιστία του προγράμματος, καθιστώντας αναγκαία την ύπαρξη προηγμένων εργαλείων στατικής ανάλυσης για τον εντοπισμό πιθανών προβλημάτων.

3.3.3 CSS

Τα Cascading Style Sheets (CSS) είναι μια τυποποιημένη γλώσσα που χρησιμοποιείται για τον καθορισμό της οπτικής απεικόνισης δομημένων εγγράφων, κυρίως HTML και XML. Επιτρέπει στους προγραμματιστές ιστοσελίδων να διαχειρίζονται τη διάταξη, τα χρώματα, τις γραμματοσειρές και τη συνολική εμφάνιση των ιστοσελίδων, παρέχοντας συνέπεια και βελτιωμένη εμπειρία χρήσης σε διάφορες συσκευές και μεγέθη οθόνης [36]. Η CSS λειτουργεί με την εφαρμογή κανόνων στυλ σε στοιχεία HTML μέσω επιλογέων, επιτρέποντας τη διάκριση μεταξύ περιεχομένου και σχεδιασμού για βελτιωμένη συντηρησιμότητα [34]. Η CSS προσφέρει ευελιξία στην αποτελεσματική διαμόρφωση στοιχείων ιστού μέσω χαρακτηριστικών όπως η κλιμακούμενη προτεραιότητα, η κληρονομικότητα και η εξειδίκευση. Φιλοξενεί διάφορες μεθόδους εφαρμογής, όπως ενσωματωμένα (inline), εσωτερικά (internal) και εξωτερικά (external) φύλλα στυλ, επιτρέποντας στους προγραμματιστές να επαναχρησιμοποιούν τα στυλ σε πολλές σελίδες μειώνοντας παράλληλα τον πλεονασμό. Επιπλέον, οι βελτιώσεις στους προεπεξεργαστές CSS, όπως οι SASS (Syntactically Awesome Style Sheets) και LESS (Leaner CSS), έφεραν χαρακτηριστικά που μοιάζουν με τον προγραμματισμό, όπως οι μεταβλητές, η ένθεση και τα mixins, ενισχύοντας την επεκτασιμότητα και την αποδοτικότητα εκτεταμένων διαδικτυακών έργων.

Τα κύρια χαρακτηριστικά που κατέχει η CSS είναι τα ακόλουθα:

- **Διαχωρισμός περιεχομένου και παρουσίασης:** Η CSS επιτρέπει τη δημιουργία στυλ ανεξάρτητα από το πλαίσιο HTML, απλοποιώντας τη συντήρηση και την ενημέρωση των ιστοτόπων.

- **Κλιμακούμενη ιεραρχία:** Τα στυλ εκχωρούνται σύμφωνα με την προτεραιότητα από διάφορες πηγές, όπως inline, internal και external φύλλα στυλ.
- **Διαμόρφωση βασισμένη σε επιλογείς:** Οι προγραμματιστές μπορούν να στοχεύουν σε συγκεκριμένα στοιχεία χρησιμοποιώντας κλάσεις, αναγνωριστικά, χαρακτηριστικά και ψευδο-επιλογείς.
- **Κληρονομικότητα:** Συγκεκριμένα χαρακτηριστικά, όπως το χρώμα και η γραμματοσειρά του κειμένου, μεταφέρονται αυτόματα από τα γονικά στοιχεία στους απογόνους τους, ελαχιστοποιώντας την επανάληψη.
- **Προσαρμοστικός σχεδιασμός:** Η CSS επιτρέπει στις διατάξεις να προσαρμόζονται σε διάφορες διαστάσεις οθόνης και συσκευές μέσω ερωτημάτων πολυμέσων.
- **Προεπεξεργαστές (SASS & LESS):** Προσθέτει χαρακτηριστικά που μοιάζουν με τον προγραμματισμό, όπως μεταβλητές, φωλιασμό και mixins, ενισχύοντας την επεκτασιμότητα και την αποδοτικότητα της CSS για πολύπλοκα έργα.

Η CSS λειτουργεί μέσω μιας συλλογής κανόνων στυλ που καθορίζουν τον τρόπο παρουσίασης των στοιχείων HTML. Κάθε κανόνας περιλαμβάνει έναν επιλογέα που προσδιορίζει συγκεκριμένα στοιχεία και ένα μπλοκ δήλωσης που περιέχει ένα ή περισσότερα ζεύγη ιδιοτήτων-τιμών που περιγράφουν λεπτομερώς την εμφάνιση αυτών των επιλεγμένων στοιχείων [36]. Τα CSS μπορούν να χρησιμοποιηθούν μέσω inline styles (τοποθετημένα απευθείας μέσα σε ένα στοιχείο HTML), εσωτερικών stylesheets (που βρίσκονται μέσα σε μια ετικέτα `<style>` στο έγγραφο HTML) ή εξωτερικών stylesheets (ένα ξεχωριστό αρχείο `.css` συνδεδεμένο με την HTML) [34]. Η γλώσσα προσφέρει πολλαπλούς επιλογείς, όπως class (`.`), ID (`#`), attribute (`[ ]`) και ψευδο-επιλογείς (`:hover`, `:nth-child`), επιτρέποντας στους προγραμματιστές να διαμορφώνουν δυναμικά τα στοιχεία ανάλογα με τις ιδιότητές τους ή τις αλληλεπιδράσεις του χρήστη. Επιπλέον, τα CSS ακολουθούν μια δομή κλιμάκωσης και κληρονομικότητας, όπου οι κανόνες επιβάλλονται σύμφωνα με την ιδιαιτερότητα, την ακολουθία και την ιεραρχία, εξασφαλίζοντας ότι τα στυλ εφαρμόζονται αποτελεσματικά και ομοιόμορφα σε όλη την ιστοσελίδα.

Στον Πίνακα 3.7 παρακάτω, απεικονίζεται ένα παράδειγμα σύνταξης κώδικα CSS.

```
#saveModelBtn {
    margin-left: 10px;
}

/* Arrow of each section of API Documents */
.accordion-button::after {
    color: #fff; /* Makes the arrow white */
    filter: brightness(100%);
}

/* Style for the API path text */
.api-path {
    font-family: 'Courier New', monospace;
    font-size: 14px;
    word-wrap: break-word;
```

```
16      white-space: pre-wrap;
17      overflow-wrap: break-word;
18      line-height: 1.5;
19  }
```

Table 3-6 Παράδειγμα κώδικα CSS

3.3.4 Bootstrap

Το Bootstrap είναι ένα δωρεάν πλαίσιο front-end που δημιουργήθηκε από το Twitter το 2011 για να βοηθήσει στην ανάπτυξη ευέλικτων και ευέλικτων εφαρμογών ιστού. Δημιουργήθηκε για να αντιμετωπίσει το ζήτημα της βελτιστοποίησης των ιστότοπων για διαφορετικά μεγέθη οθόνης, διευκολύνοντας την ομαλή πλοήγηση τόσο σε επιτραπέζιες όσο και σε κινητές πλατφόρμες. Το Bootstrap συνδυάζει HTML, CSS και JavaScript, προσφέροντας στους προγραμματιστές ένα σύνολο έτοιμων στοιχείων UI, όπως κουμπιά, μενού πλοήγησης, φόρμες και πλέγματα. Μέσω της χρήσης των CSS Media Queries, το Bootstrap επιτρέπει στις εφαρμογές ιστού να προσαρμόζουν τις διατάξεις τους δυναμικά ανάλογα με το μέγεθος της οθόνης της συσκευής, βελτιώνοντας την εμπειρία του χρήστη και την προσβασιμότητα [37]. Αυτό το πλαίσιο έχει κερδίσει δημοτικότητα λόγω της απλής εφαρμογής του, της συμβατότητας μεταξύ των φυλλομετρητών και της ικανότητάς του να απλοποιεί την ανάπτυξη ιστοσελίδων χωρίς την ανάγκη για εκτεταμένες γνώσεις κωδικοποίησης.

Το Bootstrap είναι ένα εξαιρετικά ευπροσάρμοστο front-end πλαίσιο που αναγνωρίζεται για την ανταπόκρισή του, επιτρέποντας στις εφαρμογές ιστού να προσαρμόζονται αβίαστα σε διάφορα μεγέθη οθόνης μέσω του συστήματος πλέγματος 12 στηλών. Υιοθετεί μια στρατηγική «πρώτα για κινητά», δίνοντας προτεραιότητα στη βέλτιστη απόδοση των ιστότοπων σε smartphones και tablets πριν επεκταθεί σε μεγαλύτερες οθόνες. Το πλαίσιο παρέχει μια ποικιλία από προκατασκευασμένα στοιχεία UI, όπως κουμπιά, φόρμες και μπάρες πλοήγησης, απλοποιώντας την ανάπτυξη ιστοσελίδων και μειώνοντας την ανάγκη για εκτενή κωδικοποίηση. Επιπλέον, το Bootstrap διασφαλίζει τη συμβατότητα σε διαφορετικά προγράμματα περιήγησης, διατηρώντας μια συνεπή εμφάνιση και λειτουργικότητα σε όλα τα σύγχρονα προγράμματα περιήγησης ιστού [37]. Οι προγραμματιστές μπορούν επίσης να επωφεληθούν από την εύκολη προσαρμογή του μέσω προεπεξεργαστών CSS όπως το SASS και το LESS, επιτρέποντάς τους να τροποποιούν αποτελεσματικά τα στυλ για να ικανοποιούν συγκεκριμένες σχεδιαστικές ανάγκες.

Το Bootstrap λειτουργεί ενσωματώνοντας HTML, CSS και JavaScript για την ανάπτυξη ευέλικτων και ελκυστικών εφαρμογών ιστού. Χρησιμοποιεί ένα πλαίσιο πλέγματος 12 στηλών, επιτρέποντας στους προγραμματιστές να οργανώνουν το περιεχόμενο με ευελιξία και εγγυάται την αυτόματη προσαρμογή στις διάφορες διαστάσεις της οθόνης. Το πλαίσιο χρησιμοποιεί CSS Media Queries για να τροποποιεί τα στυλ σε πραγματικό χρόνο σύμφωνα με τις αναλύσεις των συσκευών, παρέχοντας μια απρόσκοπτη εμπειρία χρήστη σε επιτραπέζιους υπολογιστές, tablet κι έξυπνων κινητών (smartphones) [37]. Επιπλέον, το Bootstrap προσφέρει έτοιμα στοιχεία και πρόσθετα JavaScript, όπως modals, dropdowns και carousels, διευκολύνοντας τα διαδραστικά και φιλικά προς το χρήστη σχέδια με ελάχιστη κωδικοποίηση. Οι προγραμματιστές έχουν τη δυνατότητα να ενσωματώσουν το Bootstrap μέσω ενός CDN (Content Delivery Network) ή κατεβάζοντας και συνδέοντας τα αρχεία τοπικά για εργασία χωρίς σύνδεση.

Στον Πίνακα 3.8, απεικονίζεται ένα παράδειγμα δήλωσης της Bootstrap για την CSS και JavaScript μέσα σε ένα αρχείο HTML.

```
<!DOCTYPE html>
<html lang="en">
    <head>
        <meta charset="UTF-8">
        <meta name="viewport" content="width=device-width, initial-scale=1.0">
        <title>Home | AutoKNN</title>
        <!-- logo of web app -->
        <link rel="icon" href="./source/image/logo.ico">
        <!-- CSS -->
        <link rel="stylesheet" href="./source/css/style.css">
        <!-- Bootstrap CSS -->
        <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/css/bootstrap.min.css" rel="stylesheet">
    </head>
    </body>
        <!-- Some code -->

        <!-- jQuery, Popper, Bootstrap JS, JavaScript -->
        <script src="https://code.jquery.com/jquery-3.7.1.min.js"></script>
                <script  src="https://cdn.jsdelivr.net/npm/@popperjs/core@2.9.2/dist/umd/popper.min.js"  integrity="sha384-IQsoLXl5PILFhosVNubq5LC7Qb9DXgDA9i+tQ8Zj3iwWAwPtgFTxbJ8NT4GN1R8p" crossorigin="anonymous"></script>
        <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/js/bootstrap.bundle.min.js"></script>
        <script src="./source/js/script.js"></script>

        <!-- Rest of code -->
    </body>
</html>
```

Table 3-7 Παράδειγμα κώδικα HTML ορισμού και χρήσης Bootstrap

3.3.5 jQuery

Η jQuery είναι μια γρήγορη και ελαφριά βιβλιοθήκη JavaScript που απλοποιεί τη διάσχιση εγγράφων HTML, το χειρισμό συμβάντων, τις κινούμενες εικόνες και τις αλληλεπιδράσεις AJAX, βελτιώνοντας την αποτελεσματικότητα της ανάπτυξης ιστοσελίδων. Έχει σχεδιαστεί για να κάνει τη συγγραφή JavaScript ευκολότερη, προσφέροντας μια σύνταξη που είναι πιο ευανάγνωστη και διαχειρίσιμη. Η jQuery, με τις ισχυρές ενσωματωμένες λειτουργίες της, επιτρέπει στους προγραμματιστές να χειρίζονται το DOM, να αντιδρούν στις αλληλεπιδράσεις του χρήστη και να εκτελούν ασύγχρονες αιτήσεις με ελάχιστο κώδικα [38]. Επιπλέον, λειτουργεί καλά με πολλαπλά προγράμματα περιήγησης, εξασφαλίζοντας μια συνεπή εμπειρία σε διαφορετικές πλατφόρμες ιστού. Αυτά τα χαρακτηριστικά καθιστούν την jQuery ένα απαραίτητο εργαλείο για τη σύγχρονη ανάπτυξη ιστοσελίδων, ιδίως για εφαρμογές HTML5.

Η jQuery αποτελείται από τα εξής κύρια χαρακτηριστικά:

- **Γρήγορο και ελαφρύ:** Η jQuery έχει σχεδιαστεί για αποδοτικότητα με μικρό μέγεθος αρχείου, εξασφαλίζοντας γρήγορη φόρτωση στην πλευρά του πελάτη.
- **Διευκολύνει την πλοήγηση και την τροποποίηση εγγράφων HTML:** Διευκολύνει την απλή επιλογή και τροποποίηση στοιχείων μέσω φιλικών προς το χρήστη επιλογέων.

- **Αποτελεσματική διαχείριση εκδηλώσεων:** Προσφέρει ένα ομαλό πλαίσιο για το χειρισμό των αλληλεπιδράσεων του χρήστη σε διάφορα προγράμματα περιήγησης.
- **Ενσωματωμένα χαρακτηριστικά animation:** Επιτρέπει την ανάπτυξη ζωντανών διεπαφών χρήστη με ευκολία.
- **Ισχυρή συμβατότητα AJAX:** Επιτρέπει την απρόσκοπτη μεταφορά δεδομένων μεταξύ του πελάτη και του διακομιστή χωρίς την ανάγκη πλήρους ανανέωσης της σελίδας.
- **Συμβατότητα μεταξύ διαφορετικών φυλλομετρητών (Cross-Browser):** Εγγυάται ότι τα σενάρια λειτουργούν ομοιόμορφα σε διάφορες πλατφόρμες ιστού, εξαλείφοντας τις τυπικές ασυνέπειες της JavaScript.

Η jQuery λειτουργεί χρησιμοποιώντας επιλογείς τύπου CSS για τον εντοπισμό στοιχείων HTML και την εκτέλεση διαφορετικών μεθόδων σε αυτά. Εξορθολογίζει τη διαχείριση συμβάντων μέσω λειτουργιών όπως `.click()` και `.hover()`, διευκολύνοντας τη γρήγορη απόκριση στις ενέργειες του χρήστη. Οι προγραμματιστές έχουν τη δυνατότητα να τροποποιούν το DOM μέσω μεθόδων όπως οι `.html()`, `.append()` και `.remove()`, επιτρέποντας δυναμικές ενημερώσεις του περιεχομένου. Επιπλέον, η jQuery διευκολύνει τις κλήσεις Ajax μέσω μεθόδων όπως `$.ajax()`, `$.get()` και `$.post()`, επιτρέποντας την ομαλή αλληλεπίδραση με τον διακομιστή χωρίς ανανέωση της σελίδας [38]. Η ικανότητά της για αλυσιδωτή σύνδεση μεθόδων επιτρέπει την εκτέλεση πολλών λειτουργιών σε μία μόνο γραμμή κώδικα, βελτιώνοντας τόσο την αναγνωσιμότητα όσο και την αποδοτικότητα.

Στον Πίνακα 3.9 απεικονίζεται ένα παράδειγμα σύνταξης κώδικα γλώσσας JavaScript με τη χρήση της βιβλιοθήκης jQuery.

```
$(document).ready(function() {
    // Check if the user is logged in by looking for the token
    var token = localStorage.getItem('token');

    // If no token is found, redirect to the login page
    if (!token) {
        window.location.href = './login.html';
        return; // Prevent further execution of the script
    }

    // Fetch user data from session storage
    var fname = localStorage.getItem('fname');
    var lname = localStorage.getItem('lname');

    // User is logged in
    $('#loginBtn').hide();
    $('#registerBtn').hide();
    $('#profileNav').show();
    $('#username').text(fname + ' ' + lname);

    // Logout functionality
```

```
22      $('#logoutBtn').on('click', function() {
23          localStorage.removeItem('token');
24          localStorage.removeItem('fname');
25          localStorage.removeItem('lname');
26          localStorage.removeItem('allowPublic');
27          window.location.href = '../index.html'; // Redirect to home page
28      });
29
30      // Rest of code
31  });
```

Table 3-8 Παράδειγμα κώδικα JavaScript όπου χρησιμοποιεί τη jQuery

Κεφάλαιο 4ο: Σχεδίαση και Υλοποίηση του AutoKNN

4.1 Λειτουργικές απαιτήσεις

Οι λειτουργικές απαιτήσεις συμβάλλουν στον καθορισμό βασικών κριτήρια για τη λειτουργικότητα ενός συστήματος λογισμικού, και τον χειρισμό ποικίλων απαιτήσεων των χρηστών. Ο ορθός καθορισμός τους βοηθάει στη βελτιστοποίηση της εμπειρίας του χρήστη, και ταυτόχρονα μειώνει τα προβλήματα απόδοσης.

Οι λειτουργικές απαιτήσεις της διαδικτυακής εφαρμογής AutoKNN, είναι οι ακόλουθες:

- **Εγγραφή του χρήστη:** Κάθε χρήστης έχει τη δυνατότητα να δημιουργήσει ένα νέο λογαριασμό στη διαδικτυακή εφαρμογή. Στη σελίδα όπου πραγματοποιείται η εγγραφή, υπάρχει μια φόρμα όπου κάθε χρήστης απαιτείται να συμπληρώσει τα προσωπικά του στοιχεία, τα οποία είναι τα εξής: Όνομα (First Name), Επώνυμο (Last Name), Ηλεκτρονική Διεύθυνση (Email Address), και το Κωδικό Πρόσβασης (Password). Για την ολοκλήρωση της δημιουργίας λογαριασμού, είναι αναγκαία η επαλήθευση των στοιχείων του, μέσω ηλεκτρονικού μηνύματος που θα έχει αποσταλλεί στο Email που συμπλήρωσε στη φόρμα εγγραφής.
- **Είσοδος του χρήστη:** Εφόσον ο χρήστης επαλήθευσε τα στοιχεία του, θα μπορεί να πραγματοποιήση την είδοσο του στην εφαρμογή. Στη σελίδα εισόδου, ο χρήστης καλείται να συμπληρώσει στη φόρμα το Email και το password που χρησιμοποίησε στην εγγραφή. Είναι σημαντικό ο χρήστης να κάνει είσοδο στην εφαρμογή, για να μπορεί να αξιοποιήσει την εφαρμογή στη πληρότητά της, καθώς όταν δεν κάνει είσοδο, οι δυνατότητές του στην εφαρμογή είναι πολύ περιορισμένες.
- **Επαναφορά του κωδικού πρόσβασης:** Ο κάθε χρήστης που έχει κάνει εγγραφή, έχει τη δυνατότητα να επαναφέρει το λογαριασμό, στη περίπτωση που ξέχασε το κωδικό πρόσβασης. Η σελίδα επαναφοράς του κωδικού αποτελείται από μια φόρμα όπου ο χρήστης καλείται να συμπληρώσει το Email του. Για την ορθή επαναφορά, πρέπει ο χρήστης να ακολουθήσει τις οδηγίες επαναφοράς που θα αποσταλλούν μέσω ηλεκτρονικού μηνύματος στο Email εγγραφής του.
- **Επεξεργασία στοιχείων του χρήστη:** Όποιος χρήστης επιθυμεί να τροποποιήσει τα προσωπικά του στοιχεία, έχει τη δυνατότητα να το κάνει μέσω της σελίδας τροποποίησης. Στη περίπτωση αλλαγής Ονόματος ή/και Κωδικού Πρόσβασης, εμφανίζεται μια φόρμα όπου ο χρήστης καλείται να επιλέξει αν θα αλλάξει είτε το όνομα, είτε το κωδικό είτε και τα δύο, και καλείται να συμπληρώσει εκ νέου, ανάλογα τι επέλεξε, και με το που συμπληρώσει νέα στοιχεία, θα γίνεται κατευθείαν αλλαγή. Ωστόσο, αν επιθυμεί να αλλάξει το Email του, εμφανίζεται μια φόρμα όπου πρέπει να συμπληρώση νέο Email, και για την ολοκλήρωση αλλαγής Email, πρέπει να γίνει εκ νέου επαλήθευση μέσω ηλεκτρονικού μηνύματος στο καινούργιο Email.
- **Διαγραφή λογαριασμού του χρήστη:** Αν επιθυμεί να διαγράψει το λογαριασμό του από την εφαρμογή, έχει τη δυνατότητα να το κάνει μέσω σελίδας διαγραφής λογαριασμού. Η σελίδα αυτή αποτελείται από μια φόρμα, όπου πρέπει ο χρήστης να συμπληρώσει την Ηλεκτρονική Διεύθυνσή του και τον Κωδικό Πρόσβασης του.
- **Σελίδες πληροφόρησης:** Για τη πληροφόρηση σχετικά με τις βασικές λειτουργίες που περιέχει η εφαρμογή, θα υπάρχει μια σελίδα με όνομα Αρχική Σελίδα (Home). Για τη πληροφόρηση σχετικά με γενικές πληροφορίες που αφορούν την εργασία, θα υπάρχει μια σελίδα με όνομα Σχετικά (About).
- **Σελίδες εκτέλεσης αλγορίθμου:** Για την εκτέλεση του αλγορίθμου κατηγοριοποίησης k-NN, θα υπάρξουν δύο σελίδες, οι οποίες θα ομαδοποιηθούν κάτω από τη κατηγορία Υπηρεσίες (Services), και θα είναι οι εξής: Η πρώτη θα ονομάζεται Δημιουργία ενός μοντέλου (Create a model), όπου θα εκτελεί εργασίες δημιουργίας νέων μοντέλων κατηγοριοποίησης, και η δεύτερη θα ονομάζεται Προ-εκπαιδευμένα Μοντέλα (Prertained Models), όπου θα εκτελεί

εργασίες σχετικές με προ-εκπαιδευμένα μοντέλα. Αυτές οι σελίδες θα είναι προσβάσιμες μόνο μέσω χρηστών που έχουν κάνει εγγραφή και είσοδο στην εφαρμογή.

- **Μεταφόρτωση αρχείου:** Κάθε χρήστης θα μπορεί να ανεβάζει αρχεία τα οποία αντιπροσωπεύουν σύνολα δεδομένων (έγκυρα μόνο αρχείου τύπου .csv). Τα συγκεκριμένα αρχεία θα μπορούν να αξιοποιηθούν είτε για να πραγματοποιείται εκπαίδευση μοντέλου κατηγοριοποίησης, είτε για κατηγοριοποίηση των στιγμιοτύπων τους. Επιπλέον, θα δίνεται η επιλογή είτε να ανήκουν σε Δημόσιο (Public) φάκελο, όπου θα είναι διαθέσιμο για κάθε χρήστη, εφόσον ο διαχειριστής του παρέχει άδεια να ανεβάζει δημόσια, είτε σε Ιδιωτικό (Private) φάκελο, όπου μόνο ο χρήστης που το έκανε μεταμόρφωση θα έχει πρόσβαση.
- **Προεπισκόπηση αρχείου:** Κάθε φορά που ο χρήστης επιλέγει ένα από τα διαθέσιμα σύνολα δεδομένων, θα πραγματοποιείται προεπισκόπηση ενός τμήματος του επιλεγμένου αρχείου σε μορφή πίνακα (συγκεκριμένα είναι ορατές οι πρώτες 10 γραμμές). Αυτό συμβάλλει στη καλύτερη κατανόηση της δομής των δεδομένων που επρόκειτο να χρησιμοποιήσει πριν την εκτέλεση του k-NN.
- **Διαγραφή αρχείου:** Εφόσον επιλέξει ο χρήστης ένα αρχείο συνόλου δεδομένων, έχει τη δυνατότητα να το διαγράψει αν επιθυμεί. Αν ανήκει βέβαια στο Δημόσιο φάκελο, θα είναι επιτρεπτό μόνο αν ο διαχειριστής του παραχωρήσει σχετική άδεια.
- **Δημιουργία μοντέλου κατηγοριοποίησης:** Κάθε χρήστης έχει τη δυνατότητα να δημιουργήσει νέα μοντέλα k-NN, βάσει το σύνολο δεδομένων εκπαίδευσης που επέλεξε. Θα έχει επίσης τη δυνατότητα να μπορεί να επιλέγει όποια διαθέσιμα χαρακτηριστικά (features) επιθυμεί από σχετική λίστα πεδίων συνόλου, καθώς και το διαθέσιμο πεδίο κλάσης (class). Επιπρόσθετα, θα μπορεί να ορίσει την παράμετρο k, και τη μετρική απόστασης. Για διευκόλυνση του χρήστη, θα μπορεί τόσο για το k όσο και για τη μετρική απόσταση να ορίσει προκαθορισμένες τιμές. Αν το σύνολο δεδομένων έχει τουλάχιστον 1.000 δεδομένα, του δίνεται η επιλογή να πραγματοποιήσει στρωματοποιημένη δειγματοληψία.
- **Αξιολόγηση απόδοσης του μοντέλου:** Για την αξιολόγηση της απόδοσης του μοντέλου, θα εκτελείται η μέθοδος train test split, όπου το σύνολο δεδομένων θα χωρίζεται σε 70% σύνολο εκπαίδευσης και 30% σύνολο δοκιμής. Αν ο χρήστης επιλέξει να εφαρμόσει στρωματοποιημένη δειγματοληψία, τότε εφαρμόζεται στο σύνολο εκπαίδευσης με μέγεθος εκπαίδευσης (train size) 1.000 δείγματα. Εφόσον εκτελεστεί ο k-NN, θα είναι δυνατή η προβολή των μετρικών απόδοσης και για κάθε κλάση ξεχωριστά (label), αλλά και για το μοντέλο συνολικά.
- **Αποθήκευση του νέου μοντέλου:** Ύστερα την εκτέλεση του αλγορίθμου, ο χρήστης αν επιθυμεί μπορεί να αποθηκεύσει το νέο μοντέλο, έτσι ώστε να μπορεί να το αξιοποιήσει για μελλοντικές εργασίες. Για την αποθήκευση του μοντέλου, ζητείται η συμπλήρωση ονόματος που θα αντιπροσωπεύει το μοντέλο.
- **Προεπισκόπηση του μοντέλου:** Κάθε φορά που ο χρήστης επιλέγει ένα μοντέλο, εφόσον έχει δημιουργήσει κι αποθηκεύσει τουλάχιστον ένα, θα πραγματοποιείται προεπισκόπηση των features και το class του επιλεγμένου μοντέλου.
- **Διαγραφή του μοντέλου:** Εφόσον επιλέξει ένα μοντέλο, ο χρήστης έχει τη δυνατότητα να διαγράψει το επιλεγμένο μοντέλο, εφόσον επιθυμεί.
- **Κατηγοριοποίηση νέων στιγμιοτύπων:** Εφόσον επιλέξει ο χρήστης ένα προ-εκπαιδευμένο μοντέλο, έχει τη δυνατότητα να κατηγοριοποιήσει νέα στιγμιότυπα, τα οποία θα παρέχονται από ένα σχετικό σύνολο δεδομένων, όπου ο χρήστης θα κάνει μεταμόρφωση και θα επιλέξει (ίδια λογική με τη προαναφερόμενη μεταμόρφωση αρχείου). Εφόσον επιλέξει ένα σχετικό σύνολο δεδομένων, πραγματοποιείται η προεπισκόπησή του σε μορφή πίνακα (ίδια λογική με τη προαναφερόμενη προεπισκόπηση αρχείου), κι αφού ολοκληρωθεί η κατηγοριοποίηση των νέων στιγμιοτύπων, θα προβάλλονται τα αποτελέσματα της κατηγοριοποίησης. Αν το σχετικό σύνολο δεδομένων περιέχει τη στήλη class, πέρα από το πίνακα των εγγραφών, θα μπορεί ο χρήστης να δει τις μετρικές, όπου πραγματοποιούν εκτίμηση της ποιότητας των αποτελεσμάτων. Θα μπορεί επίσης ο χρήστης να εξάγει (download) το σύνολο δεδομένων σε μορφή .csv.
- **Προεπισκόπηση δημόσιου Web API:** Όποιος χρήστης κάνει είσοδο στην εφαρμογή κι ενδιαφέρεται να μάθει το προγραμματιστικό κομμάτι των λειτουργιών, θα μπορεί να έχει

πρόσβαση στις προαναφερόμενες λειτουργίες του AutoKNN, μέσω δημόσιου Web API. Αυτή η σελίδα, θα παρέχει αναλυτικές πληροφορίες καθώς και σχετικά παραδείγματα για τη χρήση των endpoints που αντιστοιχούν σε κάθε μια από τις προαναφερόμενες λειτουργικές απαιτήσεις.

4.2 Αρχιτεκτονική AutoKNN

Η διαδικτυακή εφαρμογή του AutoKNN αποσκοπεί στη παροχή ενός λογισμικού περιβάλλοντος, όπου θα έχει τη δυνατότητα να εφαρμόζει κατηγοριοποίηση δεδομένων με τη χρήση του αλγόριθμου κατηγοριοποίησης k-NN. Συγκεκριμένα, εφόσον ο χρήστης ολοκληρώσει τη διαδικασία της αυθεντικοποίησης και πραγματοποιήσει είσοδο στην εφαρμογή, θα μπορεί να επιλέξει οποιοδήποτε επιθυμητό και διαθέσιμο σύνολο δεδομένων εκπαίδευσης για κατηγοριοποίηση. Εφόσον επιλέξει ένα σύνολο δεδομένων, ακολουθεί η διαδικασία καθορισμού τόσο των χαρακτηριστικών (features) όσο και της κλάσης (class) του επιλεγμένου συνόλου δεδομένων εκπαίδευσης. Παράλληλα με το καθορισμό των πεδίων features και class του συνόλου, πρέπει ο χρήστης να καθορίσει και τις παραμέτρους του κατηγοριοποιητή k-NN, συγκεκριμένα το k, τη μετρική απόσταση (metric distance) και το p, εφόσον η απόσταση είναι η 'Minkowski'. Αν το σύνολο δεδομένων είναι μεγάλο (δηλαδή περιέχει τουλάχιστον 1.000 δεδομένα), τότε δίνεται η δυνατότητα στο χρήστη να επιλέξει αν θέλει να εφαρμόσει στρωματοποιημένη δειγματοληψία (stratified sampling). Αφού ο χρήστης καθορίσει τα πεδία και τις παραμέτρους, θα εκτελεστεί ο αλγόριθμος κατηγοριοποίησης k-NN για να δημιουργηθεί το μοντέλο και η αξιολόγηση της αποτελεσματικότητας θα πραγματοποιηθεί μέσω της μεθόδου train test split, όπου θα παρέχει και τα αποτελέσματα των μετρικών.

Το AutoKNN θα δίνει επίσης τη δυνατότητα στο χρήστη να αποθηκεύει το μοντέλο, εφόσον το επιθυμεί, το οποίο μοντέλο θα μπορέσει να το χρησιμοποιήσει στο μέλλον για να μπορέσει να προβλέψει μη κατηγοριοποιημένα δεδομένα. Για να μπορέσει να προβλέψει ορθά ο χρήστης, θα πρέπει να επιλέξει το επιθυμητό διαθέσιμο μοντέλο (το οποίο είναι της μορφής '.pkl'), όπως θα πρέπει να επιλέξει και το μη κατηγοριοποιημένο σύνολο δεδομένων, το οποίο θα πρέπει να ταιριάζει με το μοντέλο (δηλαδή να έχει κοινά features και class). Στη συνέχεια, θα εφαρμόζεται κατηγοριοποίηση στο επιλεγμένο σύνολο κι ύστερα θα εμφανίζονται τα αποτελέσματα της διαδικασίας, όπου ο χρήστης θα έχει την ικανότητα να τα εξάγει σε μορφή '.csv'. Αν το επιλεγμένο σύνολο δεδομένων δεν είναι κατηγοριοποιημένο, δηλαδή περιέχει το πεδίο της κλάσης, πέρα από την εμφάνιση των αποτελεσμάτων της κατηγοριοποίησης, θα γίνεται προβολή και των μετρικών, για την εκτίμηση της ποιότητας των αποτελεσμάτων.

Οι προαναφερόμενες λειτουργίες θα είναι προσβάσιμες τόσο μέσω της γραφικής διεπαφής χρήστη (GUI), όσο και μέσω διαδικτυακού (Web) API. Ανάμεσα στο GUI και το Web API, οι κύριες διαφορές τους σχετίζονται με τη ταυτοποίηση του χρήστη, καθώς και στο τρόπο εμφάνισης των αποτελεσμάτων. Συγκεκριμένα, για τη ταυτοποίηση του χρήστη μέσω GUI, θα πρέπει να εισάγει τα στοιχεία του, τα οποία είναι το email του και το κωδικό πρόσβασης που χρησιμοποίησε για να κάνει εγγραφή στο AutoKNN. Ωστόσο, για τη ταυτοποίηση μέσω Web API, είναι αναγκαίο ο χρήστης να εισάγει το δικό του API token που δημιουργήθηκε κατά τη διαδικασία της εγγραφής. Σχετικά με την εμφάνιση των αποτελεσμάτων, μέσω του Web API, εμφανίζονται με τη μορφή JSON.

Στο Σχήμα 4.1 παρακάτω, απεικονίζεται το διάγραμμα ροής του AutoKNN, όπου εστιάζεται στη λειτουργικότητά του όπως προαναφέρθηκε.

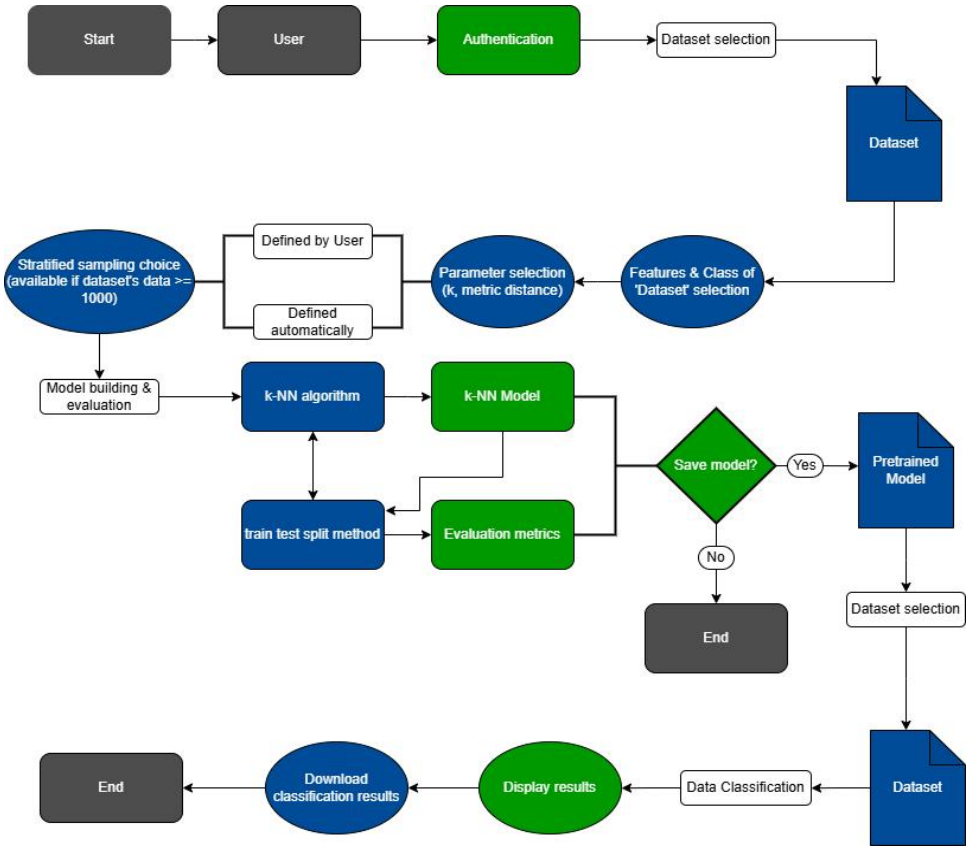


Figure 4-1 Διάγραμμα ροής του AutoKNN

Όσον αφορά την αρχιτεκτονική του AutoKNN, αποτελείται από 5 κύρια τμήματα:

- **Web API:** Αναπτύχθηκε με τη γλώσσα προγραμματισμού PHP κι αναγνωρίζεται ως πηγή συνεχής κίνησης της εφαρμογής. Καθορίζει δυνατή την επικοινωνία με το Back-end και τη δρομολόγηση της εκτέλεσης πολλών διεργασιών. Μπορεί επιπρόσθετα, όταν είναι ανάγκη, να συνεργάζεται και με τα υπόλοιπα τμήματα της εφαρμογής.
- **Βάση δεδομένων (Database):** Χρησιμοποιήθηκε η MySQL. Πραγματοποιείται η χρήση της σε εργασίες όπως είναι η εγγραφή, η αυθεντικοποίηση κι η διαχείριση των χρηστών της εφαρμογής. Σε συγκεκριμένες περιστάσεις, αποτελεί υποστήριξη για την ολοκλήρωση αναγκαίων διεργασιών αποθήκευσης ενός προ-εκπαιδευμένου μοντέλου.
- **Σύστημα αρχείων του Server (File system):** Βρίσκεται σε επικοινωνία με το Web API για την αποθήκευση των αρχείων του χρήστη, όπως τα σύνολα δεδομένων (εκπαίδευσης και μη κατηγοριοποιημένα) και μοντέλα.
- **Εργασίες Μηχανικής Μάθησης (M.L. modules):** Τα modules είναι προγραμματισμένα με τη γλώσσα Python και με τη βιβλιοθήκη Scikit-learn. Γίνεται η κλήση τους από το Web API, για να εκτελέσουν τον αλγόριθμο κατηγοριοποίησης k-NN και διάφορες εργασίες Μηχανικής Μάθησης.
- **Γραφική Διεπαφή:** Η σχεδίαση κι η ανάπτυξή του πραγματοποιήθηκε μέσω των HTML, CSS, Bootstrap, JS και iQuery. Επικοινωνεί με το Web API, αποσκοπώντας στους χρήστες να είναι ικανοί να προσπελάσουν τις διαθέσιμες λειτουργίες της εφαρμογής, μέσω ενός φιλικού προς τον χρήστη περιβάλλον.

Στο Σχήμα 4.2 παρακάτω, παρουσιάζεται η αρχιτεκτονική του AutoKNN, όπως προαναφέρθηκε.

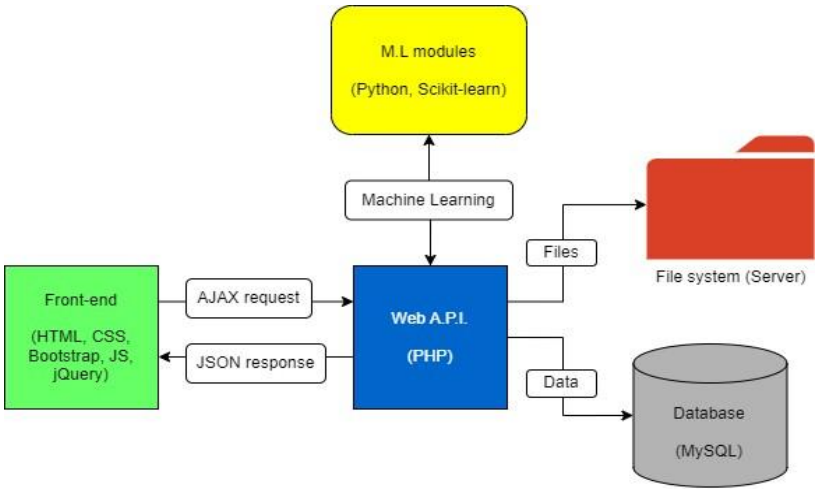


Figure 4-2 Αρχιτεκτονική του AutoKNN

4.3 Χρήστες, δημόσια και ιδιωτικά σύνολα δεδομένων

Στην εφαρμογή AutoKNN, όσον αφορά τα σύνολα δεδομένων, ομαδοποιούνται ανάλογα με τη πρόσβαση που έχουν οι χρήστες. Οι τύποι συνόλων είναι οι εξής:

- **Ιδιωτικά σύνολα (Private):** Σε αυτά, ο κάθε χρήστης που έχει κάνει εγγραφή στην εφαρμογή, μπορεί να μεταφορτώσει όποια επιθυμεί στο δικό του λογαριασμό και να τα αξιοποιήσει. Ωστόσο, εφόσον τα μεταφόρτωσε ιδιωτικά, δεν μπορεί να τα διαμοιράσει με άλλους εγγεγραμμένους χρήστες της εφαρμογής.
- **Δημόσια σύνολα (Public):** Σε αυτά, όπως και στα ιδιωτικά σύνολα, μπορεί κάθε εγγεγραμμένος χρήστης να μεταφορτώσει τα επιθυμητά αρχεία και να τα χρησιμοποιήσει. Η κύρια διαφορά όμως με τα ιδιωτικά σύνολα. είναι ότι τα δημόσια μεταφορτωμένα αρχεία, είναι διαθέσιμα, προσβάσιμα κι αξιοποιήσιμα από όλους τους εγγεγραμμένους χρήστες. Τα δημόσια σύνολα έχουν ως πρωταρχικό στόχο τη διευκόλυνση των χρηστών και στη δημιουργία μοντέλων με τον k-NN, αλλά και σε περιστάσεις όπως για παράδειγμα ο διαμοιρασμός συνόλων ανάμεσα σε ορισμένες ομάδες ανθρώπων.

Η επιλογή ανάμεσα σε ιδιωτικά και δημόσια σύνολα πραγματοποιείται μόνο στη περίπτωση της εκπαίδευσης νέων μοντέλων, καθώς στη περίσταση της κατηγοριοποίησης στιγμιοτύπων αποθηκεύονται τα επιθυμητά σύνολα δεδομένων στο προσωπικό λογαριασμό, με αποτέλεσμα να έχει πρόσβαση μόνο ο χρήστης που τα μεταφόρτωσε.

Ένας χρήστης για να έχει τη δυνατότητα να ανεβάσει (επίσης και να διαγράψει) ένα σύνολο δεδομένων, πρέπει να του παραχωρηθεί σχετική άδεια από τον διαχειριστή της εφαρμογής. Για λόγους ασφαλείας, και βάσει δικαιωμάτων του χρήστη, οι τύποι χρηστών της εφαρμογής είναι οι εξής:

- **Μη εγγεγραμμένοι (Not registered):** Οι χρήστες που δεν έχουν κάνει εγγραφή στην εφαρμογή, κι εν τέλει δεν κατέχουν δικαιώματα.
- **Απλοί (Ordinary):** Οι χρήστες που έχουν κάνει εγγραφή, οπότε έχουν πλήρη πρόσβαση στην εφαρμογή. Παρ' όλα αυτά, δεν έχουν το δικαίωμα να ανεβάζουν και να διαγράφουν δημόσια σύνολα δεδομένων.
- **Προχωρημένοι (Advanced):** Οι χρήστες που έχουν κάνει εγγραφή, οπότε έχουν πλήρη πρόσβαση στην εφαρμογή. Επιπλέον, ύστερα από την εγγραφή τους, παραχωρήθηκαν δικαιώματα μεταφόρτωσης και διαγραφής δημοσίων συνόλων δεδομένων από τον διαχειριστή της εφαρμογής.
- **Διαχειριστής (Administrator):** Αυτός ο χρήστης, διαχειρίζεται όλη την εφαρμογή, οπότε κατέχει και δικαιώματα διαχειρισμού δημοσίων συνόλων δεδομένων. Επιπρόσθετα, μέσω της

Βάσης Δεδομένων (για λεπτομέριες δείτε την ενότητα 4.4 παρακάτω), επιλέγει ποιοι εγγεγραμμένοι χρήστες θα κατέχουν επίσης δικαιώματα χειρισμού δημοσίων συνόλων δεδομένων.

4.4 Βάση Δεδομένων

Η Βάση Δεδομένων MySQL του AutoKNN, όπως προαναφέραμε στην ενότητα 4.2, χρησιμοποιείται για ενέργειες σχετικές με το χρήστη και τα προ-εκπαιδευμένα μοντέλα. Συγκεκριμένα, όσον αφορά το χρήστη, διαχειρίζεται την εγγραφή, αυθεντικοποίηση και γενικά τη διαχείριση του, και σχετικά με τα προ-εκπαιδευμένα μοντέλα, συμβάλλει στην ορθή αποθήκευσή τους.

Η Βάση Δεδομένων της Web εφαρμογής αποτελείται από 4 βασικούς πίνακες, οι οποίοι είναι οι ακόλουθοι: 'users', 'verify_account', 'dataset_execution' και 'models'.

Στο Σχήμα 4.3 μπορούμε να παρατηρήσουμε το πίνακα του 'users', όπου αξιοποιείται για τη ταυτοποίηση του χρήστη. Περιέχει 8 πεδία, τα οποία αναλυτικά είναι τα εξής:

- **'id':** Αναγνωριστικό του χρήστη, το οποίο είναι μοναδικό για κάθε έναν, δημιουργείται κατα την εγγραφή του κι εισάγεται αυτόματα μέσω της εντολής 'AUTO_INCREMENT'. Επίσης, αποτελεί το κύριο κλειδί (primary key) του πίνακα.
- **'fname':** Όνομα του χρήστη.
- **'lname':** Επώνυμο του χρήστη.
- **'email':** Email του χρήστη.
- **'pass':** Κωδικός πρόσβασης του χρήστη που δημιούργησε κατα τη διαδικασία της εγγραφής. Για λόγους ασφαλείας, ο κωδικός μετατρέπεται σε μορφή hash πριν την εισαγωγή το στη Βάση.
- **'token':** Κωδικός του χρήστη, με τον οποίο μπορεί να αποκτήσει πρόσβαση στο Web API του AutoKNN και να εκτελέσει διάφορα αιτήματα προς αυτό. Δημιουργείται κι εισάγεται ως εξής: μέσω της συνάρτησης 'random_bytes()' παράγονται μια σειρά από 50 κρυπτογραφημένα τυχαία bytes, και μέσω της συνάρτησης 'bin2hex()' τα δυαδικά 50 bytes μετατρέπονται σε δεκαεξαδικό αριθμό, συνολικού μήκους 100 χαρακτήρων (1 byte ισούται με 2 δεκαεξαδικούς χαρακτήρες).
- **'email_verification':** Υπόδειξη επιβεβαίωσης του email που ο χρήστης δήλωσε στην εγγραφή. Εξ' ορισμού η προκαθορισμένη τιμή είναι 0, ωστόσο αν ο χρήστης επιβεβαιώση τη διεύθυνση με επιτυχία, τότε η τιμή είναι '1'.
- **'allowPublic':** Υπόδειξη δικαιώματος του χρήστη σχετικά με τη διαχείριση δημόσιων συνόλων δεδομένων εκπαίδευσης. Εξ' ορισμού η προκαθορισμένη τιμή είναι 0, ωστόσο αν ο διαχειριστής παραχωρήσει στον χρήστη σχετικό δικαίωμα, τότε η τιμή είναι '1'.

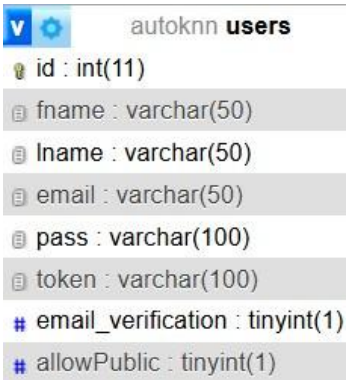


Figure 4-3 Πίνακας 'users' της Βάσης Δεδομένων

Στο Σχήμα 4.4 μπορούμε να παρατηρήσουμε το πίνακα του 'verify_account', όπου αξιοποιείται για την επιβεβαίωση του χρήστη. Ειδικά όσον αφορά τη διαδικασία της εγγραφής, αλλαγής email κι ανάκτησης λογαριασμού λόγω απώλειας κωδικού πρόσβασης του χρήστη. Περιέχει 3 πεδία, τα οποία αναλυτικά είναι τα εξής:

- **'id_of_user':** Αναγνωριστικό του χρήστη, το οποίο είναι ξένο κλειδί (foreign key) του πίνακα 'users'. Ταυτόχρονα είναι κύριο κλειδί του πίνακα.
- **'verification_key':** Κωδικός επιβεβαίωσης του λογαριασμού του χρήστη, ο οποίος περιέχει μια τυχαία συμβολοσειρά (string) αρχικά μήκους 16 bytes με την συνάρτηση 'random_bytes()', κι ύστερα με τον αλγόριθμο MD5, όπου τελικό μήκος είναι ίσο με 32.
- **'creation_time':** Χρονικό διάστημα (ημερομηνία κι ώρα) δημιουργίας κωδικού επιβεβαίωσης για μελλοντικό έλεγχο λήξης του.

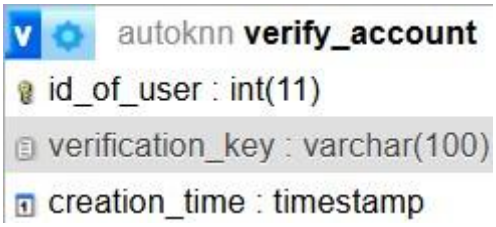


Figure 4-4 Πίνακας 'verify_account' της Βάσης Δεδομένων

Στο Σχήμα 4.5 μπορούμε να παρατηρήσουμε το πίνακα του 'dataset_execution', όπου αξιοποιείται για τη διαχείριση του συνόλου δεδομένων που επέλεξε ο χρήστης να χρησιμοποιήσει στον αλγόριθμο k-NN. Συγκεκριμένα, αξιοποιείται για τον έλεγχο ύπαρξης ενός ολόιδιου εκτελέσιμου συνόλου (δηλαδή ενός συνόλου δεδομένων με ίδιες παραμέτρους, επιλεγμένα features και class), έτσι ώστε αντί να εκτελεστεί ξανά ο αλγόριθμος, απλά να τραβήξει και να προβάλλει τα αποτελέσματα για το συγκεκριμένο σύνολο. Περιέχει 8 πεδία, τα οποία αναλυτικά είναι τα εξής:

- **'id':** Αναγνωριστικό του εκτελέσιμου συνόλου δεδομένων, το οποίο είναι μοναδικό για κάθε ένα, δημιουργείται κατα την εκτέλεση του k-NN κι εισάγεται αυτόματα μέσω της εντολής 'AUTO_INCREMENT'. Επίσης, αποτελεί το κύριο κλειδί του πίνακα.
- **'id_of_user':** Αναγνωριστικό του χρήστη, το οποίο είναι ξένο κλειδί του πίνακα 'users'.
- **'name_of_dataset':** Όνομα του συνόλου δεδομένων που χρησιμοποιήθηκε για την εκτέλεση του k-NN.
- **'parameters':** Οι παράμετροι που χρησιμοποιήθηκαν για την εκτέλεση του k-NN. Οι παράμετροι είναι οι εξής: features, class, k, metric_distance, p και stratified_sampling, όπου και μετατρέπονται (κωδικοποιούνται) σε μορφή JSON. Στη Βάση Δεδομένων εμφανίζονται με αυτόν το τρόπο: {"features": "sepal.length,sepal.width,petal.length,petal.width", "target": "variety", "k": "1,3,5,7,9,11,13,15,17,19,21,23,25,27,29,31,33,35,37,39,41,43,45,47,49", "distance": "euclidean", "p": "", "stratified_sampling": "false"}, όπου το target είναι το επιλεγμένο class του συνόλου δεδομένων και distance είναι η επιλεγμένη μετρική απόσταση.
- **'status':** Η κατάσταση που βρίσκεται η διαδικασία εκτέλεσης του κατηγοριοποιητή k-NN. Αποτελείται από 4 καταστάσεις: 'Not Started' (προεπιλεγμένη/αρχική τιμή), 'In Progress' (σε διαδικασία εκτέλεσης), 'Completed' (επιτυχής ολοκλήρωση εκτέλεσης), 'Failed' (αποτυχής ολοκλήρωση εκτέλεσης).
- **'results_path':** Η σχετική διαδρομή όπου τα αποτελέσματα εκτελεσμένων συνόλων δεδομένων βρίσκονται. Έχει 2 τύπου μονοπάτια: το 1ο είναι αν χρησιμοποιηθεί ένα δημόσιο σύνολο και το 2ο είναι αν χρησιμοποιηθεί ένα ιδιωτικό σύνολο.
- **'creation_time':** Το χρονικό διάστημα όπου ο k-NN για το επιλεγμένο σύνολο ενός χρήστη ξεκίνησε να εκτελείται.

- **'update_time':** Το ενημερωμένο χρονικό διάστημα όπου ο k-NN για το επιλεγμένο σύνολο ενός χρήστη σταμάτησε να εκτελείται, είτε επιτυχημένα (status = 'Completed') είτε αποτυχημένα (status = 'Failed').

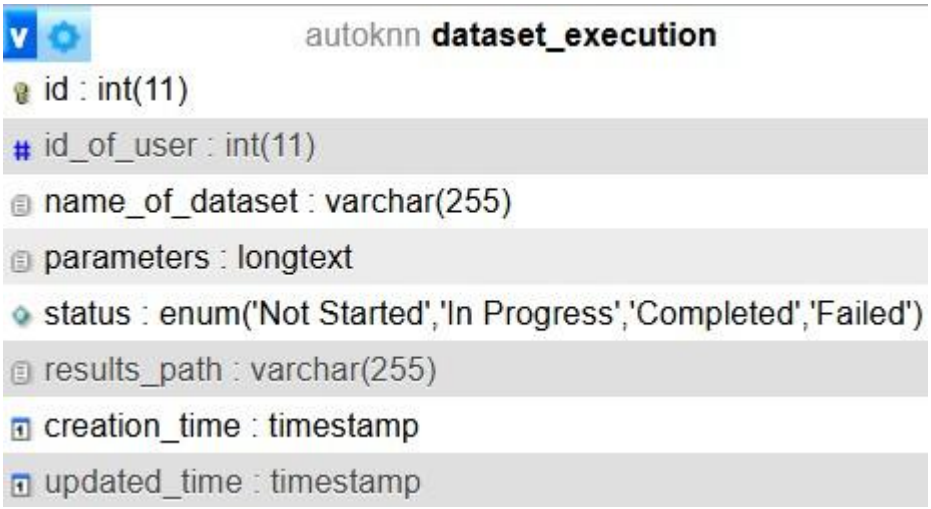


Figure 4-5 Πίνακας 'dataset_execution' της Βάσης Δεδομένων

Στο Σχήμα 4.6 μπορούμε να παρατηρήσουμε το πίνακα του 'models', όπου αξιοποιείται για τη διαχείριση του προ-εκπαιδευμένου μοντέλου του χρήστη. Συγκεκριμένα, ασχολείται με διεργασίες όπως η αποθήκευση του μοντέλου, κι η προβολή των features και του class. Περιέχει 9 πεδία, τα οποία αναλυτικά είναι τα εξής:

- **'id':** Αναγνωριστικό του μοντέλου, το οποίο είναι μοναδικό για κάθε ένα, δημιουργείται κατα την αποθήκευσή του κι εισάγεται αυτόματα μέσω της εντολής 'AUTO_INCREMENT'. Επίσης, αποτελεί το κύριο κλειδί του πίνακα.
- **'id_of_executed_dataset':** Αναγνωριστικό του εκτελέσιμου συνόλου δεδομένων, το οποίο είναι ξένο κλειδί του πίνακα 'dataset_execution'.
- **'name_of_model':** Το όνομα του μοντέλου που χρησιμοποίησε ο χρήστης για να το αποθηκεύσει.
- **'features':** Τα γνωρίσματα του μοντέλου που χρησιμοποιήθηκαν για τη δημιουργία του. Χρησιμοποιούνται επίσης για να γίνει η προβολή τους στη σελίδα Προ-εκπαιδευμένα Μοντέλα.
- **'name_of_class':** Το όνομα της κλάσης του μοντέλου που χρησιμοποιήθηκε για τη δημιουργία του. Χρησιμοποιείται επίσης για να γίνει η προβολή του στη σελίδα Προ-εκπαιδευμένα Μοντέλα.
- **'k':** Η παράμετρος k που χρησιμοποιήθηκε για τη δημιουργία του.
- **'metric_distance':** Η μετρική απόσταση που χρησιμοποιήθηκε για τη δημιουργία του.
- **'p'**: Η παράμετρος p που χρησιμοποιήθηκε για τη δημιουργία του. Περιέχει 2 τύπους τιμών: είτε NULL είτε ακέραιος (συγκεκριμένα 3 ή 4).
- **'stratified_sampling':** Η παράμετρος στρωματοποιημένης δειγματοληψίας που χρησιμοποιήθηκε για τη δημιουργία του. Αν χρησιμοποιήθηκε, τότε η τιμή είναι 1, αλλιώς είναι 0.

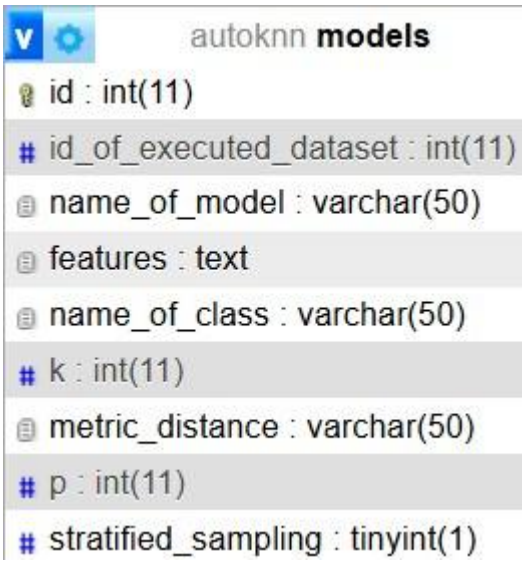


Figure 4-6 Πίνακας 'models' της Βάσης Δεδομένων

Στο Σχήμα 4.7, απεικονίζεται το διάγραμμα οντοτήτων-συσχετίσεων (Entity-Relationship diagram - ER diagram) της Βάσης Δεδομένων του AutoKNN.

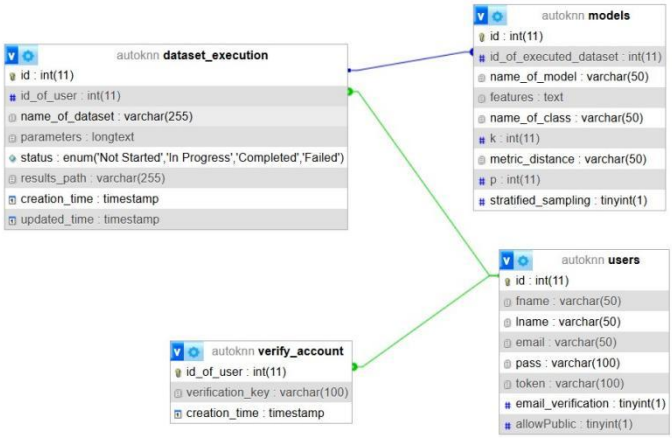


Figure 4-7 Διάγραμμα ER Βάσης Δεδομένων του AutoKNN

4.5 Web API

Όπως προαναφέρθηκε, το Web API είναι πηγή συνεχής κίνησης της εφαρμογής, διότι εκτελούνται πολλές διεργασίες μέσω αυτού κι επικοινωνεί επιτυχημένα με το Back-end. Ο χρήστης για να μπορέσει να έχει πρόσβαση στο Web API της εφαρμογής AutoKNN, πρέπει να καλέσει το αντίστοιχο endpoint, βάσει επιθυμητής λειτουργίας. Όταν καλεστεί το endpoint από τον χρήστη, το ίδιο πραγματοποιεί κλήση στο αντίστοιχο αρχείο PHP, κι όταν ολοκληρωθεί η εκτέλεση του, επιστρέφει το endpoint το αποτέλεσμα της εκτέλεσης με τη μορφή JSON. Για κάθε endpoint της εφαρμογής, για την ορθή κλήση του, είναι αναγκαία η δήλωση σχετικών παραμέτρων βάσει του endpoint που καλείται. Ο τρόπος δήλωσης παραμέτρων διαφέρει ανάλογα με το είδος μεθόδου της HTTP (HyperText Transfer Protocol) που αξιοποιεί το συγκεκριμένο endpoint, καθώς για τα POST και DELETE οι παράμετροι εισάγονται στο σώμα του αιτήματος, ενώ για το GET μπορεί και να εισαχθούν στο σώμα, αλλά και να είναι ενσωματωμένα στο URL (Uniform Resource Locator) του

αιτήματος endpoint, μετά τον χαρακτήρα '?'. Όπως είναι ορατό στο Πίνακα 4.1, εμφανίζονται όλα τα endpoint του Web API που χρησιμοποιεί το AutoKNN, όπου κάθε ένα από αυτά τα αρχεία PHP βρίσκονται στο φάκελο: https://kclusterhub.iee.ihu.gr/autoknn/server/php/api κι είναι συνολικά 25 endpoints. Αν ένας χρήστης για παράδειγμα επιθυμεί να κάνει λήψη το σύνολο δεδομένων που έχει επιλέξει, η σύνταξη του endpoint είναι η εξής: https://kclusterhub.iee.ihu.gr/autoknn/server/php/api/download_dataset.php?token=token\&file=file. Στην αντίθετη περίπτωση, όπου επιθυμεί να διαγράψει το επιλεγμένο σύνολο δεδομένων, το URL του σχετικού endpoint είναι το εξής: https://kclusterhub.iee.ihu.gr/autoknn/server/php/api/delete_dataset.php, και το σώμα του αιτήματος θα διαμορφωθεί σε μορφή JSON ως εξής: {"token": "token", "file": "file", "folder": folderType"}.

Number	HTTP Method	Endpoint	Description
1	POST	register.php	Εγγραφή νέου χρήστη
2	POST	login.php	Είσοδος του χρήστη
3	POST	edit_name_password.php	Τροποποίηση ονόματος ή/και κωδικού πρόσβασης του χρήστη
4	POST	edit_email.php	Τροποποίηση email του χρήστη
5	DELETE	delete_account.php	Διαγραφή του χρήστη
6	GET	list_datasets.php	Απαρίθμηση διαθέσιμων συνόλων δεδομένων
7	POST	upload_dataset.php	Μεταφόρτωση νέου συνόλου δεδομένων από τον χρήστη
8	GET	load_dataset.php	Φόρτωση επιλεγμένου συνόλου δεδομένων
9	GET	download_dataset.php? token={token}&file={file}	Λήψη επιλεγμένου συνόλου δεδομένων
10	DELETE	delete_dataset.php	Διαγραφή επιλεγμένου συνόλου δεδομένων
11	GET	list_unclassified_datasets.php	Απαρίθμηση διαθέσιμων μη κατηγοριοποιημένων συνόλων δεδομένων
12	POST	upload_unclassified_dataset.php	Μεταφόρτωση νέου μη κατηγοριοποιημένου συνόλου δεδομένων από τον χρήστη
13	GET	load_unclassified_dataset.php	Φόρτωση επιλεγμένου μη κατηγοριοποιημένου συνόλου δεδομένων
14	GET	download_unclassified_dataset.php? token={token}&file={file}	Λήψη επιλεγμένου μη κατηγοριοποιημένου συνόλου δεδομένων
15	DELETE	delete_unclassified_dataset.php	Διαγραφή επιλεγμένου μη κατηγοριοποιημένου συνόλου δεδομένων
16	GET	load_classified_dataset.php	Φόρτωση κατηγοριοποιημένου συνόλου δεδομένων
17	GET	download_classified_dataset.php	Λήψη κατηγοριοποιημένου συνόλου δεδομένων

18	GET	list_models.php	Απαρίθμηση διαθέσιμων κατασκευασμένων μοντέλων
19	GET	get_model_content.php	Ανάκτηση περιεχομένου μοντέλου (επιλεγμένα features και class)
20	GET	download_model.php? token={token}&file={file}	Λήψη επιλεγμένου κατασκευασμένου μοντέλου
21	DELETE	delete_model.php	Διαγραφή επιλεγμένου κατασκευασμένου μοντέλου
22	POST	post_knn_train_test.php	Εκτέλεση κατηγοριοποιητή k-NN
23	GET	get_knn_train_test.php	Ανάκτηση αποτελεσμάτων εκτέλεσης του k-NN (είτε πριν, αν υπάρχουν ήδη, είτε μετά την εκτέλεση)
24	POST	save_model.php	Αποθήκευση κατασκευασμένου μοντέλου
25	POST	classify_data.php	Κατηγοριοποίηση του επιλεγμένου μη κατηγοριοποιημένου συνόλου δεδομένων, βάσει του επιλεγμένου κατασκευασμένου μοντέλου

Table 4-1 Endpoints του Web API στον AutoKNN

Ακολουθούνται μερικά παραδείγματα κώδικα PHP για τη καλύτερη κατανόηση της λειτουργικότητας του Web API. Στη αρχή της κλήσης αρχείου κώδικα PHP, πραγματοποιείται έλεγχος των παραμέτρων που εισήγαγε ο χρήστης. Όπως απεικονίζεται στον Πίνακα 4.2, γίνεται έλεγχος για τον ορθό ορισμό μεθόδου HTTP (εδώ για την POST), ύστερα για τις παραμέτρους (όνομα, επώνυμο, email και κωδικός πρόσβασης). Στη περίπτωση σφάλματος είτε στη μέθοδο είτε στις παράμετρους, επιστρέφει ως αποτέλεσμα σχετικό header και μήνυμα σφάλματος κι ολοκληρώνεται η εκτέλεση του κώδικα.

```php
<?php
    // Part of register.php

    require_once "../db_connection.php";
    require_once "../functions.php";
    require_once "../phpmailer.php";

    header('Content-Type: application/json');

    // Ensure the request method is POST
    if ($_SERVER['REQUEST_METHOD'] !== 'POST') {
        header("HTTP/1.1 400 Bad Request");
        echo json_encode(["message" => "Only POST requests are allowed"]);
        exit;
    }

    // Retrieve the POST data
    $data = json_decode(file_get_contents("php://input"), true);

    // Validate required parameters
    if (!isset($data["fname"]) || !isset($data["lname"]) || !isset($data["email"]) || !isset($data["password"]) || !isset($data["confirmPassword"])) {
        header("HTTP/1.1 400 Bad Request");
        echo json_encode(["status" => "danger", "message" => "Missing required parameters"]);
        exit;
```

```
25     }
26
27     // Rest of code
28 ?>
```

Table 4-2 Κώδικας PHP για έλεγχο μεθόδου HTTP κι ορθότητας παραμέτρων

Σε πολλές περιπτώσεις, για τον πιο ορθό έλεγχο ορισμένων συνθηκών, είναι αναγκαία η εμπλοκή της Βάσης Δεδομένων. Στον παρακάτω Πίνακα 4.3 απεικονίζεται ένα παράδειγμα σύνταξης κώδικα SQL μέσα στο αρχείο PHP με τη βιβλιοθήκη MySQLi, όπου πραγματοποιείται έλεγχος ύπαρξης του email που ο χρήστης εισήγαγε στη φόρμα εγγραφής του AutoKNN.

```
1  <?php
2      // Part of register.php
3
4      // Some code
5
6      // Check if email already exists
7      $sql = "SELECT id FROM users WHERE email = ?";
8      $stmt = $mysqli->prepare($sql);
9      $stmt->bind_param("s", $email);
10     $stmt->execute();
11     $result = $stmt->get_result();
12
13     if ($result->num_rows > 0) {
14         echo json_encode(["status" => "warning", "message" => "Email already exists."]);
15         exit;
16     }
17
18     $stmt->close();
19
20     // Rest of code
21 ?>
```

Table 4-3 Κώδικας PHP για έλεγχο ύπαρξης email στον πίνακα ‘users’

Όπως προαναφέραμε στην προηγούμενη ενότητα 4.4, ο κωδικός πρόσβασης που ο χρήστης χρησιμοποίησε για την εγγραφή του, μετατρέπεται πρώτα σε μορφή hash, όπως και το token με τον αλγόριθμο MD5, κι ύστερα εισάγονται στη Βάση Δεδομένων. Στον Πίνακα 4.4, απεικονίζεται ο κώδικας όπου μετατρέπει το κωδικό και το token σε μορφή hash, καθώς και την εισαγωγή των στοιχείων εγγραφής του χρήστη στη Βάση Δεδομένων.

```
1  <?php
2      // Part of register.php
3
4      // Some code
5
6      // Hash the password and generate a random token
7      $hashedPassword = password_hash($password, PASSWORD_DEFAULT);
```

```
8       $token = bin2hex(random_bytes(50));
9
10      // Insert the user into the database
11      $sql = "INSERT INTO users (fname, lname, email, pass, token) VALUES (?, ?, ?, ?, ?)";
12      $stmt = $mysqli->prepare($sql);
13      $stmt->bind_param("sssss", $fname, $lname, $email, $hashedPassword, $token);
14      $stmt->execute();
15      $stmt->close();
16
17      // Rest of code
18  ?>
```

Table 4-4 Κώδικας PHP για μετατροπή του token σε hash μορφή κι εισαγωγή στοιχείων εγγραφής χρήστη στη Βάση Δεδομένων

Το token είναι ένα σημαντικό χαρακτηριστικό που ταυτοποιεί τον χρήστη, το οποίο είναι απαραίτητο για να εκτελεστούν οι περισσότερες διεργασίες του Web API. Όπως απεικονίζεται στον Πίνακα 4.5, γίνεται ένας έλεγχος της παραμέτρου token και ταυτόχρονα γίνεται κι έλεγχος για το αν υπάρχει στη Βάση Δεδομένων. Αν υπάρχει, επιστρέφεται το email του χρήστη, αλλιώς επιστρέφει αντίστοιχο μήνυμα σφάλματος.

```
1   <?php
2       // Part of list_datasets.php
3
4       // Some code
5
6       if (!isset($_GET['token'])) {
7           http_response_code(400);
8           echo json_encode(["status" => "danger", "message" => "Missing required parameters"]);
9           exit;
10      }
11
12      // Fetch user email based on the session token
13      $token = $_GET['token'];
14      $sql = "SELECT email FROM users WHERE token = ?";
15      $stmt = $mysqli->prepare($sql);
16      $stmt->bind_param('s', $token);
17      $stmt->execute();
18      $result = $stmt->get_result();
19
20      if ($result->num_rows === 0) {
21          http_response_code(401);
22          echo json_encode(["status" => "danger", "message" => "Invalid token"]);
23          exit;
24      }
25
```

```
26      // Rest of code
27  ?>
```

Table 4-5 Κώδικας PHP για έλεγχο ύπαρξης του token του χρήστη

Σχετικά με τη μεταφόρτωση ενός νέου συνόλου δεδομένων, για να πραγματοποιηθεί ορθή μεταφόρτωσης του συνόλου από τον χρήστη, θα πρέπει να γίνει απαραίτητος έλεγχος όσον αφορά το τύπο και το μέγεθος του αρχείου. Στον Πίνακα 4.6, πραγματοποιούνται σχετικοί έλεγχοι, ειδικότερα για τύπου αρχείου '.csv' και για μέγεθος μεγαλύτερο από 10 MB (10 * 1024 bytes * 1024 bytes = 10.485.760 bytes).

```
1   <?php
2       // Part of upload_dataset.php
3
4       // Some code
5
6       // Validate the file type (must be a .csv)
7       $allowedTypes = ['text/csv', 'application/vnd.ms-excel'];
8       if (!in_array($file['type'], $allowedTypes)) {
9           http_response_code(400);
10          echo json_encode(["status" => "danger", "message" => "Invalid file type. Only CSV files are allowed."]);
11          exit;
12      }
13
14      // Validate the file size (must be less than 10MB)
15      if ($file['size'] > 10 * 1024 * 1024) {
16          http_response_code(400);
17          echo json_encode(["status" => "danger", "message" => "File size exceeds 10MB limit."]);
18          exit;
19      }
20
21      // Rest of code
22  ?>
```

Table 4-6 Κώδικας PHP για έλεγχο τύπου και μεγέθους μεταφορτωμένου αρχείου

Επιπρόσθετα, είναι σημαντικός ο έλεγχος μεταφόρτωσης σε δημόσιο ή ιδιωτικό φάκελο. Στη περίπτωση που ο χρήστης επέλεξε να ανεβάσει δημόσια το νέο σύνολο, πρέπε να γίνει έλεγχος από τη Βάση Δεδομένων για το αν έχει δικαίωμα ο χρήστης να εκτελέσει αυτή την ενέργεια, και γενικά να διαχειρίζεται δημόσια σύνολα. Είτε επιλέξει μεταφόρτωση δημόσια είτε ιδιωτικά, γίνεται επιπλέον έλεγχος για την ύπαρξη αυτού του αρχείου στο φάκελο που επέλεξε ο χρήστης να κάνει μεταφόρτωση. Στον Πίνακα 4.7 απεικονίζονται οι έλεγχοι σχετικά με τον τύπο φακέλου που επέλεξε ο χρήστης, στη περίπτωση που ο χρήστης επέλεξε δημόσια μεταφόρτωση ελέγχει αν έχει δικαίωμα διαχειρισμού δημόσιων αρχείων (allowPublic), και αν το αρχείο υπάρχει ήδη στον επιλεγμένο φάκελο.

```
1   <?php
2       // Part of upload_dataset.php
3
4       // Some code
5
6       // Query to find the user based on the token
```

```
7       $query = "SELECT id, email, allowPublic FROM users WHERE token = ?";
8       $stmt = $mysqli->prepare($query);
9       $stmt->bind_param("s", $token);
10      $stmt->execute();
11      $stmt->bind_result($userId, $email, $allowPublicFromDb);
12      $stmt->fetch();
13      $stmt->close();
14
15      if (!$userId) {
16          http_response_code(401);
17          echo json_encode(["status" => "danger", "message" => "Invalid token"]);
18          exit;
19      }
20
21      // Generate the user's hash using the md5 function on their email
22      $userHash = md5($email);
23
24      // Determine the upload path based on the user's selection
25      $uploadDir = '../../python/';
26      if ($folder === 'public') {
27          if ($allowPublicFromDb == 0) {
28              http_response_code(403);
29              echo json_encode(["status" => "danger", "message" => "You are not allowed to upload to the public folder."]);
30              exit;
31          }
32          $uploadPath = $uploadDir . 'public/datasets';
33      } else {
34          $uploadPath = $uploadDir . 'private/' . $userHash . '/datasets';
35      }
36
37      // Ensure the upload path exists
38      if (!is_dir($uploadPath)) {
39          if (!mkdir($uploadPath, 0755, true)) {
40              http_response_code(500);
41              echo json_encode(["status" => "danger", "message" => "Failed to create directories."]);
42              exit;
43          }
44      }
45
46      // Some code
47
48      // Generate a unique filename to avoid overwriting
49      $filename = basename($file['name']);
50      $targetFile = $uploadPath . DIRECTORY_SEPARATOR . $filename;
51
52      // Check if the file already exists in the target directory
53      if (file_exists($targetFile)) {
54          http_response_code(409);
55          echo json_encode(["status" => "danger", "message" => "This file already exists in the folder"]);
56          exit;
```

```
    }

    // Move the uploaded file to the target directory
    if (move_uploaded_file($file['tmp_name'], $targetFile)) {
        http_response_code(200);
        echo json_encode(["status" => "success", "message" => "File uploaded successfully.", "file" => $filename]);
    } else {
        http_response_code(500);
        echo json_encode(["status" => "danger", "message" => "Failed to upload the file."]);
    }
?>
```

Table 4-7 Κώδικας PHP για έλεγχο τύπου φακέλου μεταφόρτωσης αρχείου

Η ανάκτηση του περιεχομένου του επιλεγμένου συνόλου δεδομένων εκπαίδευσης, εφαρμόζεται με τη JavaScript, όπως απεικονίζεται στον Πίνακα 4.8. Είναι αναγκαία η συγκεκριμένη ανάκτηση, καθώς ο χρήστης πρέπει να γνωρίζει ποια πεδία είναι εφικτά να χρησιμοποιηθούν ως features και ποια τα γνωρίσματα του class. Ο k-NN κατά κύριο λόγο διαχειρίζεται αριθμητικά (numerical) δεδομένα, καθώς βασίζεται σε υπολογισμό μετρικών αποστάσεων, οπότε, ελέγχουμε ποια features περιέχουν αριθμητικά δεδομένα.

```
$(document).ready(function() {
    // Part of new_model.js

    // Some code

    // Event listener for dataset selection
    $('#selectDataset').on('change', function() {
        // Some code

        // API call for loading the chosen dataset's contents
        $.ajax({
            url: '../server/php/api/load_dataset.php',
            method: 'GET',
            data: {
                token: token,
                file: file,
                folder: folderType
            },
            success: function(response) {
                // Hide loading button
                $('#loadDatasetBtn').hide();

                if (response.message) {
                    // Show alert if there's a message
                    showAlert('danger', response.message, '#alertPreview');
                    return;
                }

                dataCounter = response.counter;

                // Show or hide stratified sampling based on dataCounter
                if (dataCounter < 1000) {
                    $('#stratify').hide();
```

```
                    $('#checkStratifiedSampling').prop('checked', false);
                } else {
                    $('#stratify').show();
                    $('#checkStratifiedSampling').prop('checked', true);
                }

                dt = {
                    header: response.header,
                    data: response.data
                };

                // Filter and display numerical features with no missing values, and display classes
                filterAndDisplayNumericalFeatures();
                populateClassesSelect();

                // Rest of code
            },
            // Rest of code
        });
    // Rest of code
    });

    // Function to filter and display numerical features with no missing values
    function filterAndDisplayNumericalFeatures() {
        if (!dt.header.length || !dt.data.length) {
            return; // No data to filter
        }

        const header = dt.header;
        const data = dt.data;

        // Determine which columns are numerical and have no missing values
        let numericalFeatures = header.filter((feature, index) => {
            // Check if the column is numerical and has no missing values
            const columnValues = data.map(row => row[index]);
            const allValuesPresent = columnValues.every(value => value !== '');
            const isNumerical = columnValues.every(value => !isNaN(value) && value !== '');

            return isNumerical && allValuesPresent;
        });

        // Generate form HTML for numerical features with no missing values
        let formHtml = '';
        numericalFeatures.forEach(feature => {
            formHtml += `
                <div class="form-check form-check-inline">
                    <input class="form-check-input feature-checkbox" type="checkbox" value="${feature}" id="${feature}" checked>
                    <label class="form-check-label" for="${feature}">
                        ${feature}
                    </label>
                </div>
            `;
        });

        $('#individualFeatures').html(formHtml);
```

```
89      }
90
91      // Function to populate the class options in the #selectClass dropdown
92      function populateClassesSelect() {
93          if (!dt.header.length || !dt.data.length) {
94              return;
95          }
96
97          const header = dt.header;
98          const data = dt.data;
99
100         // Determine which columns have no missing values
101         let featuresWithNoMissingValues = header.filter((feature, index) => {
102             const columnValues = data.map(row => row[index]);
103             return columnValues.every(value => value !== ''); // All values are present
104         });
105
106         // Basic heuristic to determine the class column (assuming categorical and non-numeric values are classes)
107         let potentialClassFeatures = featuresWithNoMissingValues.filter(feature => {
108             const columnValues = data.map(row => row[header.indexOf(feature)]);
109             // Check if the column values are categorical (string values)
110             return columnValues.some(value => isNaN(value));
111         });
112
113         // If no clear class feature is found, use the first one from the non-missing features
114         if (potentialClassFeatures.length === 0) {
115             potentialClassFeatures = featuresWithNoMissingValues;
116         }
117
118         // Populate #selectClass with these potential class features
119         let classOptions = '<option value="default" selected>Select a class</option>';
120         potentialClassFeatures.forEach(feature => {
121             classOptions += `<option value="${feature}">${feature}</option>`;
122         });
123
124         $('#selectClass').html(classOptions);
125     }
126 });
```

Table 4-8 Κώδικας JavaScript για ανάκτηση περιεχομένου του επιλεγμένου συνόλου δεδομένων εκπαίδευσης

Είναι απαραίτητο να τονιστεί η μέθοδος κλήσης των modules εκτέλεσης του k-NN και διαφόρων εργασιών Μηχανικής Μάθησης, όπου εφαρμόζεται ως εξής: τροποποιούνται οι απαραίτητες παραμέτρους έτσι ώστε να είναι κατάλληλες για την εντολή της Python, ορίζεται η εντολή Python ('$pythonCmd') και χρησιμοποιείται η μέθοδος της PHP με ονομασία 'exec()' για να καλεστεί το σχετικό αρχείο Python. Ο τρόπος κλήσης ενός module απεικονίζεται και στον Πίνακα 4.9, μαζί και με έναν τυπικό έλεγχο, σε περίπτωση που προέκυψε κάποιο σφάλμα στην εκτέλεση του αρχείου Python.

```
1   <?php
2       // Part of post_knn_train_test.php
3
4       // Some code
5
6       // Escape and format parameters for the Python command
```

```
7       $escapedFilePath = escapeshellarg($filePath);
8       $escapedFeatures = escapeshellarg($features);
9       $escapedTarget = escapeshellarg($target);
10      $escapedKValue = escapeshellarg($k_value);
11      $escapedDistanceValue = escapeshellarg($distance_value);
12      $escapedPValue = escapeshellarg($p_value);
13      $escapedStratify = escapeshellarg($stratify);
14      $escapedResultsFilePath = escapeshellarg($resultsFilePath);
15
16      // Create the Python command to execute the kNN model
17      $pythonCmd = "python3 ../../python/knn_train_test.py $escapedFilePath $escapedFeatures $escapedTarget $escapedKValue
$escapedDistanceValue $escapedPValue $escapedStratify $escapedResultsFilePath 2>&1";
18
19      // Execute the Python command and capture the output
20      $output = [];
21      $return = 0;
22      exec($pythonCmd, $output, $return);
23
24      // Some code
25
26      // Return response based on the execution result
27      if ($return === 0) {
28          echo json_encode([
29              "message" => "Completed execution of algorithm",
30              "dataset_id" => $dataset_id,
31              "folder" => $folder,
32              "status" => $status
33          ]);
34      } else {
35          http_response_code(500);
36          echo json_encode([
37              "message" => "Failed to execute algorithm",
38              "status" => $status
39          ]);
40      }
41  ?>
```

Table 4-9 Κώδικας PHP για κλήσης αρχείου Python υπεύθυνο για εργασία εκτέλεσης κατηγοριοποιητή k-NN

4.6 Δημιουργία μοντέλου

Μια σημαντική λειτουργία της εφαρμογής AutoKNN είναι η χρήση του κατηγοριοποιητή k-NN για να δημιουργήσει μοντέλα. Ο χρήστης για να μπορέσει να φτιάξει ένα καινούργιο μοντέλο, καλείται αρχικά να επιλέξει ένα σύνολο δεδομένων εκπαίδευσης, όπου ύστερα καλείται να επιλέξει ποια διαθέσιμα πεδία θα επιλέξει ως features και class και πρέπει επίσης να ορίσει τιμές για τις παραμέτρους του κατηγοριοποιητή k-NN ('k', 'metric distance') και της μεθόδου train test split ('stratified sampling', αν το σύνολο δεδομένων έχει τουλάχιστον 1.000 δεδομένα). Μέσω JavaScript πραγματοποιείται ο έλεγχος των παραμέτρων, ύστερα εκτελείται κώδικας PHP για να ελέγξει αν έχει ήδη εκτελεστεί στο παρελθόν ο κατηγοριοποιητής με το ίδιο σύνολο, πεδία και παραμέτρους. Αν όχι, προωθεί στο αντίστοιχο Python module οι τιμές, όπου εκτελούνται ο train test split και ο k-NN, και ταυτόχρονα πραγματοποιείται μια εκτίμηση της απόδοσης του νέου μοντέλου. Στη περίπτωση ωστόσο

που στο παρελθόν έχει δημιουργήσει μοντέλο με το επιλεγμένο σύνολο δεδομένων και με τα επιλεγμένα πεδία και παραμέτρους, η PHP επιστρέφει τα αποτελέσματα σε μορφή JSON με τα αποτελέσματα και προβάλλονται στον χρήστη.

Στον Πίνακα 4.10, απεικονίζονται οι μέθοδος 'train_test_split' κι ο κατηγοριοποιητής 'KNeighborsClassifier', όπου εκτελούνται βάσει τιμών που καθόρισε ο χρήστης. Το σύνολο διαχωρίζεται σε 70% εκπαίδευσης και 30% δοκιμής, κι αν ο χρήστης επέλεξε να πραγματοποιήσει stratified sampling, γίνεται στο 70% του συνόλου δοκιμής με μέγεθος εκπαίδευσης 1.000 δείγματα. Ύστερα καλείται ο κατηγοριοποιητής k φορές.

```
# Part of knn_train_test.py

# Some code

def evaluate_knn(X_train, y_train, X_test, y_test, k, distance, p=None):
    clf = KNeighborsClassifier(n_neighbors=k, metric=distance, p=p)
    clf.fit(X_train, y_train)
    y_pred = clf.predict(X_test)

    # Overall metrics
    accuracy = accuracy_score(y_test, y_pred)
    precision = precision_score(y_test, y_pred, average='weighted', zero_division=0)
    recall = recall_score(y_test, y_pred, average='weighted', zero_division=0)
    f1 = f1_score(y_test, y_pred, average='weighted', zero_division=0)

    # Class-wise metrics
    class_report = classification_report(y_test, y_pred, output_dict=True)

    return accuracy, precision, recall, f1, class_report

# Some code

# Load dataset
dataset = pd.read_csv(file)

# Split the features and target variables
X = dataset[features].values
y = dataset[target].values

# Perform train test split (70% train, 30% test)
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3, random_state=42)

# Check if stratified sampling is needed
if stratified_sampling:
    # Perform stratified sampling on the training set
    X_sampled, _, y_sampled, _ = train_test_split(X_train, y_train, train_size=1000, random_state=42, stratify=y_train)
else:
    X_sampled, y_sampled = X_train, y_train

# Rest of code
```

Table 4-10 Κώδικας Python για εκτέλεση κατηγοριοποιητή k-NN και μεθόδου Train Test Split βάσει προκαθορισμένων παραμέτρων του χρήστη

Για κάθε επανάληψη που πραγματοποιείται, πέρα από τη κλήση της συνάρτησης που εφαρμόζει τον k-NN, υπολογίζονται για το μοντέλο στο σύνολο του τις μετρικές απόδοσης (δηλαδή τα Accuracy, Precision, Recall και F-Score) και οι μέσοι όροι για Precision, Recall και F-Score, και για κάθε γνώρισμα κλάσης (label) τις μετρικές απόδοσης κατά μέσο όρο (Precision, Recall και F-Score). Επιπρόσθετα, σε κάθε επανάληψη γίνεται εύρεση καλύτερων παραμέτρων και μετρικών απόδοσης, βάσει Accuracy. Επιστρέφονται στο χρήστη τα αποτελέσματα των υπολογισμών, όπου και προβάλλονται σε μορφή πινάκων, για να τον βοηθήσει στην απόφαση της αποθήκευσης ή μη του νέου μοντέλου. Στον Πίνακα 4.11 παρουσιάζονται οι υπολογισμοί και τα αποτελέσματα που επιστρέφονται με μορφή JSON.

```
# Part of knn_train_test.py

# Some code

# Store per-label metrics
label_metrics = {label: {'precision': 0, 'recall': 0, 'f1': 0, 'count': 0} for label in set(y_test)}

# Evaluate different combinations
for k in k_values:
    for distance in distance_values:
        if distance == 'minkowski':
            for p in p_value:
                accuracy, precision, recall, f1, class_report = evaluate_knn(X_sampled, y_sampled, X_test, y_test, k, distance, p)
        else:
            accuracy, precision, recall, f1, class_report = evaluate_knn(X_sampled, y_sampled, X_test, y_test, k, distance)

        if accuracy > max_accuracy:
            max_accuracy = accuracy
            best_k = k
            best_distance = distance
            best_p = p_value if distance == 'minkowski' else None
            best_precision = precision
            best_recall = recall
            best_f1 = f1
            best_class_metrics = class_report

        # Accumulate overall metrics
        average_accuracy += accuracy
        total_evaluations += 1

        # Accumulate per-label metrics
        for label, metrics in class_report.items():
            if label not in ['accuracy', 'macro avg', 'weighted avg']:
                total_precision_sum += metrics['precision']
                total_recall_sum += metrics['recall']
                total_f1_sum += metrics['f1-score']
                total_classes_count += 1

                # Update per-label metrics for final report
                label_metrics[label]['precision'] += metrics['precision']
                label_metrics[label]['recall'] += metrics['recall']
```

```
                label_metrics[label]['f1'] += metrics['f1-score']
                label_metrics[label]['count'] += 1

# Finalize averages
average_accuracy /= total_evaluations

# Calculate the averages based on all classes
average_precision = total_precision_sum / total_classes_count if total_classes_count > 0 else 0
average_recall = total_recall_sum / total_classes_count if total_classes_count > 0 else 0
average_f1 = total_f1_sum / total_classes_count if total_classes_count > 0 else 0

# Calculate average per-label metrics
label_metrics_final = []
for label, metrics in label_metrics.items():
    if metrics['count'] > 0:
        label_metrics_final.append({
            'class': label,
            'precision': metrics['precision'] / metrics['count'],
            'recall': metrics['recall'] / metrics['count'],
            'f1': metrics['f1'] / metrics['count']
        })

# Output the best results and the parameters the user selected as JSON
results = {
    'dataset': file,
    'features': features,
    'class': target,
    'k_values': k_values,
    'distance_values': distance_values,
    'p_value': p_value,
    'stratified_sampling': stratified_sampling,
    'best_k': best_k,
    'best_distance': best_distance,
    'best_p': best_p,
    'max_accuracy': max_accuracy,
    'best_precision': best_precision,
    'best_recall': best_recall,
    'best_f1': best_f1,
    'average_accuracy': average_accuracy,
    'average_precision': average_precision,
    'average_recall': average_recall,
    'average_f1': average_f1,
    'class_metrics': label_metrics_final
}

# Save results to a file
with open(results_path, 'w') as f:
    json.dump(results, f)

print(json.dumps(results))
```

Table 4-11 Κώδικας Python για υπολογισμό μετρικών απόδοσης μοντέλου, μέσοι όροι και μετρικές απόδοσης κατά μέσο όρο για κάθε γνώρισμα, εύρεση καλύρετων παραμέτρων κι επιστροφή αποτελεσμάτων σε JSON

Στη περίπτωση που ο χρήστης επιθυμεί να αποθηκεύσει το νέο μοντέλο, πρέπει να του δώσει ένα όνομα. Με το που ονοματίσει το μοντέλο και πατήσει το κουμπί αποθήκευσης, καλείται ένα κατάλληλο endpoint του Web API, όπου μαζί με το νέο όνομα, λαμβάνει και τις παραμέτρους που χρησιμοποιήθηκαν για τη δημιουργία του μοντέλου (δηλαδή οι σχετικές του συνόλου και του κατηγοριοποιητή). Ύστερα προωθούνται σε αντίστοιχο Python module που αποσκοπεί στην αποθήκευση του μοντέλου. Στον Πίνακα 4.12, παρουσιάζεται η διαδικασία αποθήκευσης, όπου η KNeighborsClassifier αρχικοποιείται βάσει των καλύτερων παραμέτρων ('k', 'metric distance', 'p') που βρέθηκαν στη διαδικασία δημιουργίας του μοντέλου κι εκπαιδεύεται το μοντέλο, στηριζόμενο στα features και class που ορίστηκαν από τον χρήστη. Μετέπειτα, χρησιμοποιείται η βιβλιοθήκη της Python με όνομα 'joblib' για την αποθήκευση του μοντέλου, όπου μετατρέπεται σε μορφή '.pkl'.

```
# Part of save_model.py

# Some code for declaring libraries

def save_knn_model(file_path, features, target_class, k_value, distance_value, p_value, saved_model_file):
    # Load the dataset
    data = pd.read_csv(file_path)
    X = data[features]
    y = data[target_class]

    # Instantiate and configure the kNN model
    model = KNeighborsClassifier(n_neighbors=k_value, metric=distance_value, p=p_value)
    model.fit(X, y)

    # Save the model to a file
    try:
        joblib.dump(model, saved_model_file)
        print(f"Model saved successfully at {saved_model_file}")
    except Exception as e:
        print(f"Error saving the model file: {e}")
        sys.exit(1)


if len(sys.argv) != 8:
    print("Usage: python save_model.py <file_path>, <features>, <target_class>, <k_value>, <distance_value>, <p_value>, <saved_model_file>")
    print(f"Received arguments: {sys.argv}")
    sys.exit(1)

file_path = sys.argv[1]
features = sys.argv[2].split(",")
target_class = sys.argv[3]
k_value = int(sys.argv[4])
distance_value = sys.argv[5]

p_value = sys.argv[6] if sys.argv[6] and sys.argv[6] != 'null' else None
if p_value:
    p_value = int(p_value)  # Convert to integer if it's not None

saved_model_file = sys.argv[7]

# Call the function to save the model
save_knn_model(file_path, features, target_class, k_value, distance_value, p_value, saved_model_file)
```

Table 4-12 Κώδικας Python για αποθήκευση του νέου μοντέλου

Εφόσον ολοκληρωθεί η εκτέλεση του Python module αποθήκευσης κι επιστραφούν τα αποτελέσματα στο αρχείο της PHP που έχει κληθεί, αποθηκεύονται το όνομα και τα πεδία δημιουργίας του μοντέλου στη Βάση Δεδομένων. Ο κώδικας PHP σχετικός με την αποθήκευση του μοντέλου και στη Βάση, απεικονίζεται στον Πίνακα 4.13.

```
<?php
    // Part of save_model.php

    // Some code

    // Now save the model details in models table
    $name_of_class = $target;
      $sql = "INSERT INTO models (id_of_executed_dataset, name_of_model, features, name_of_class, k, metric_distance, p, stratified_sampling) VALUES (?, ?, ?, ?, ?, ?, ?, ?)";
    $stmt = $mysqli->prepare($sql);
       $stmt->bind_param("isssisii", $dataset_id, $model_name, $features, $name_of_class, $best_k_value, $best_distance_value, $best_p_value, $stratify);
    $stmt->execute();
    $stmt->close();

    echo json_encode([
        "message" => "Model saved successfully.",
    ]);
?>
```

Table 4-13 Κώδικας PHP για αποθήκευση μοντέλου στη Βάση Δεδομένων

4.7 Χρήση προ-εκπαιδευμένου μοντέλου

Μαζί με τη δημιουργία ενός μοντέλου βάσει επιλεγμένου συνόλου δεδομένων εκπαίδευσης, μια εξίσου σημαντική λειτουργία της διαδικτυακής εφαρμογής AutoKNN, είναι η αξιοποίησή τους. Πριν την οποιαδήποτε χρήση του προ-εκπαιδευμένου μοντέλου, είναι σημαντικό να μπορεί ο χρήστης να δει το περιεχόμενό του, συγκεκριμένα τα features και το class που κατέχει. Η προβολή της δομής του επιλεγμένου μοντέλου είναι σημαντική για τη καλύτερη κατανόηση των πληροφοριών που περιέχει. Από τη στιγμή που θα επιλέξει ο χρήστης το μοντέλο που επιθυμεί, καλείται ένα αντίστοιχο endpoint του Web API, όπου ελέγχεται το όνομα του μοντέλου αν υπάρχει στη Βάση Δεδομένων. Με βάσει το όνομα του μοντέλου, ανακτούνται τα features και το class του μοντέλο και τα αποθηκεύει σε μεταβλητές PHP, όπου κι επιστρέφονται στον χρήστη με μορφή JSON. Η διαδικασία ανάκτησης των features Και class βάσει ονόματος επιλεγμένου μοντέλου, απεικονίζονται στον Πίνακα 4.14.

```
<?php
    // Part of get_model_content.php

    // Some code

    // Define models' file path
    $modelFilePath = '../../python/private/' . $hash_user . '/' . 'models_saved/' . $model;

    // Check if the file exists
    if (!file_exists($modelFilePath)) {
        echo json_encode(['error' => 'Model file not found.']);
        exit;
```

```
13      }
14
15      // Extract the base name of the model file without the extension
16      $baseFileName = pathinfo($model, PATHINFO_FILENAME);
17
18      // Extract the class of the chosen model
19      $sql = "SELECT name_of_class, features FROM models WHERE name_of_model = ?";
20      $stmt = $mysqli->prepare($sql);
21      $stmt->bind_param('s', $baseFileName);  // Use base file name
22      $stmt->execute();
23      $result = $stmt->get_result();
24
25      if ($result->num_rows === 0) {
26          echo json_encode(["status" => "danger", "message" => "Class not found for the specified model.", "model_name" =>
$baseFileName]);
27          exit;
28      }
29
30      $results = $result->fetch_assoc();
31      $class = $results['name_of_class'];
32
33      // Use features from the databasev and convert features to an array
34      $features = $results['features'];
35      $featuresArray = explode(",", $features);
36      $stmt->close();
37
38      // Return both features and class to the frontend
39      echo json_encode([
40          "status" => "success",
41          "features" => $featuresArray ?? [],  // Ensure features are present
42          "class" => $class ?? '' // Use the class retrieved from the database
43      ]);
44  ?>
```

Table 4-14 Κώδικας PHP για ανάκτησης features και class του μοντέλου από Βάση Δεδομένων

Όταν γίνεται χρήση ενός προ-εκπαιδευμένου μοντέλου k-NN, ο χρήστης μπορεί να προβλέψει μη κατηγοριοποιημένα στιγμιότυπα. Εφόσον ο χρήστης επιλέξει το μοντέλο που επιθυμεί, καλείται να επιλέξει ένα κατάλληλο σύνολο δεδομένων, όπου αποτελείται από μη κατηγοριοποιημένα δεδομένα. Για την εκτέλεση της κατηγοριοποίησης, το σύνολο δεδομένων, συγκεκριμένα η δομή του, είναι αναγκαίο να είναι ίδια με του επιλεγμένου μοντέλου, δηλαδή να έχουν κοινά features. Ο έλεγχος των features ανάμεσα σε επιλεγμένο σύνολο δεδομένων και σε επιλεγμένο μοντέλο, πραγματοποιείται με κλήση σχετικού endpoint. Από τη στιγμή που έχουν κοινή δομή, προωθούνται οι παράμετροι για τη διαδικασία της κατηγοριοποίησης προωθούνται στο κατάλληλο Python module κατηγοριοποίησης δεδομένων. Όσο αφορά τη κατηγοριοποίηση, αρχικά γίνεται ανάκληση του επιλεγμένου μοντέλου με την αξιοποίηση της βιβλιοθήκης 'joblib' και μετέπειτα για τις προβλέψεις χρησιμοποιείται η μέθοδος 'predicted'. Με την ολοκλήρωση της κατηγοριοποίησης και πρόβλεψης, επιστρέφεται στον χρήστη το σύνολο δεδομένων που επέλεξε μαζί με μία επιπλέον στήλη που αντιπροσωπεύει τα αποτελέσματα των προβλέψεων. Για να μπορέσει ο χρήστης να κάνει λήψη το νέο σύνολο δεδομένων σε μια μορφή που μπορεί να διαβαστεί, επιστρέφεται από το Python module και σε μορφή CSV. Στον παρακάτω

Πίνακα 4.15, απεικονίζονται η διαδικασία πρόβλεψης και μετατροπής σε μορφής CSV του συνόλου δεδομένων.

```
# Part of classify_data.py

# Some code for declaring libraries

if len(sys.argv) != 6:
print("Usage: python classify_data.py <file_path> <model_path> <features> <class_column> <saved_file_path>")
    sys.exit(1)

# Extract arguments passed from PHP
file_path = sys.argv[1]
model_path = sys.argv[2]
features = sys.argv[3].split(',')
class_name = sys.argv[4]
saved_file_path = sys.argv[5]

# Load the dataset
dataset = pd.read_csv(file_path)
attributes = dataset[features]

# Load the model and make predictions
model = joblib.load(model_path)
predicted_values = model.predict(attributes)

# Initialize results
results = {}

# Some code

else:
    # Add predictions to the dataset
    dataset["predicted"] = predicted_values

    # Prepare output data
    columns = dataset[features + ["predicted"]].columns.to_list()
    rows = dataset[features + ["predicted"]].values.tolist()
    data = [columns] + rows

    # Construct the results dictionary
    results = {
        "dataset": data,
        "labels": list(predicted_values),  # Predicted labels as a list
    }

# Save the classified dataset
dataset.to_csv(saved_file_path, index=False, encoding='utf-8')

print(json.dumps(results))
```

Table 4-15 Κώδικας Python για πρόβλεψη μη κατηγοριοποιημένων στιγμιοτύπων

Είναι πιθανό ο χρήστης να επιλέξει ένα σύνολο δεδομένων το οποίο να περιέχει μέσα τη στήλη της κλάσης. Στη περίπτωση αυτή, πέρα από πρόβλεψη, πραγματοποιείται κι υπολογισμός μετρικών απόδοσης για να εκτιμηθεί η ποιότητα των αποτελεσμάτων. Πριν τη κλήση του αντίστοιχου Python module, πραγματοποιείται, όπως και στα μη κατηγοριοποιημένα σύνολα, έλεγχος τόσο στα features, όσο τώρα και στο class. Κατά την εκτέλεση του Python module κατηγοριοποίησης, υπολογίζονται οι μετρικές απόδοσης, δηλαδή τα Accuracy (μόνο για το μοντέλο συνολικά), Precision, Recall και F-Score, όπου και στρογγυλοποιούνται σε 2 δεκαδικά ψηφία. Οι μετρικές απόδοσης που υπολογίζονται, αντιπροσωπεύουν τόσο το μοντέλο στο σύνολό του, όσο και για κάθε label ξεχωριστά. Στον Πίνακα 4.16, παρουσιάζεται ο υπολογισμός των μετρικών απόδοσης στη περίπτωση που υπάρχει η στήλη class στο επιλεγμένο σύνολο δεδομένων.

```
# Part of classify_data.py

# Some code

# Check if the class column is provided and valid
if class_name != 'None' and class_name in dataset.columns:
    # Calculate metrics
    class_label = dataset[class_name]
    labels = class_label.unique().tolist()  # Get unique labels

    # Calculate accuracy
    accuracy = round(metrics.accuracy_score(class_label, predicted_values), 2)

    # Calculate precision, recall, and f1-score per label
    precision_per_label, recall_per_label, fscore_per_label, _ = metrics.precision_recall_fscore_support(
        class_label, predicted_values, average=None, labels=labels, zero_division=0
    )
    precision_per_label = [round(p, 2) for p in precision_per_label]
    recall_per_label = [round(r, 2) for r in recall_per_label]
    fscore_per_label = [round(f, 2) for f in fscore_per_label]

    # Calculate average precision, recall, and f1-score
    average_precision, average_recall, average_fscore, _ = metrics.precision_recall_fscore_support(
        class_label, predicted_values, average='macro', zero_division=0
    )
    average_precision, average_recall, average_fscore = (
        round(average_precision, 2),
        round(average_recall, 2),
        round(average_fscore, 2),
    )

    # Add predictions to the dataset
    dataset["predicted"] = predicted_values

    # Prepare output data
    columns = dataset[features + ["predicted"]].columns.to_list()
    rows = dataset[features + ["predicted"]].values.tolist()
```

```
38      data = [columns] + rows
39
40      # Construct the results dictionary
41      results = {
42          "dataset": data,
43          "accuracy": accuracy,
44          "average_precision": average_precision,
45          "average_recall": average_recall,
46          "average_f1_score": average_fscore,
47          "precision_per_label": precision_per_label,
48          "recall_per_label": recall_per_label,
49          "f1_score_per_label": fscore_per_label,
50          "labels": labels,
51      }
52
53  # Rest of code
```

Table 4-16 Κώδικας Python για πρόβλεψη κατηγοριοποιημένων στιγμιοτύπων, όπου επιπλέον υπολογίζονται οι μετρικές απόδοσης

4.8 Υλοποίηση του Front-end

Το Fron-end είναι ένα σημαντικό κομμάτι του AutoKNN, καθώς με την συνεχή επικοινωνία με το Web API, γίνεται δυνατή η αξιοποίηση όλων των δυνατοτήτων που έχει να προσφέρει. Πραγματοποιείται αυτή η χρήση μέσω ενός φιλικού στο χρήστη GUI και χωρίς προγραμματιστικά εργαλεία. Η ανάπτυξη του Front-end έγινε εφικτή μέσω των γλωσσών HTML και CSS σε συνεργασία με το framework εν ονόματι Bootstrap, και παράλληλα αξιοποιήθηκε η γλώσσα προγραμματισμού JavaScript μαζί με τη βιβλιοθήκη jQuery.

Συγκεκριμένα, με τη χρήση της HTML έγινε εφικτή η δημιουργία κι η οργάνωση των περιεχομένων κάθε σελίδας της εφαρμογής AutoKNN. Όπως φαίνεται στο παρακάτω Σχήμα 4.8, οι σελίδες της εφαρμογής είναι συνολικά 13, εκ των οποίων οι 12 βρίσκονται κάτω από το φάκελο "web_pages", ενώ το 'index.html' είναι εκτός του φακέλου και μέσα στο αρχικό φάκελο, καθώς αντιπροσωπεύει την Αρχική σελίδα.

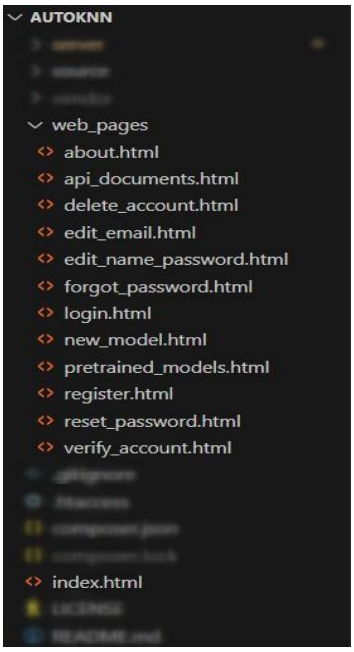


Figure 4-8 Αρχεία HTML του AutoKNN

Με την CSS καθορίζεται το πως θα προβληθούν τα περιεχόμενα των σελιδών HTML. Ο ορισμός τρόπου προβολής γίνεται με την εφαρμογή σχετικών κανόνων στα στοιχεία των σελιδών της HTML. Όπως παρατηρείται στο Σχήμα 4.9, τα αρχεία CSS είναι συνολικά 9, επειδή είναι εφικτή η αξιοποίησή τους από περισσότερες από μία σελίδες, και βρίσκονται στο φάκελο "/sources/css".

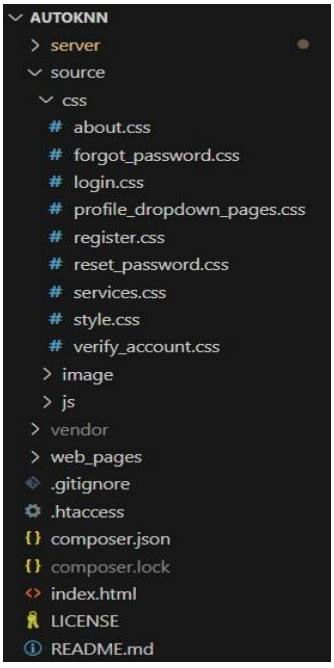


Figure 4-9 Αρχεία CSS του AutoKNN

Πέρα από την CSS, για τον καθορισμό προβολής των σελιδών χρησιμοποιήθηκε και η Bootstrap. Σε κάποια από τα στοιχεία της HTML χρησιμοποιήθηκαν εργαλεία που προσφέρει η ίδια η Bootstrap, τα οποία είναι κλάσεις έτοιμες για χρήση, όπου βοηθούν στο γρηγορότερο κι αποτελεσματικότερο σχεδιασμό συστατικών και λειτοργιών. Επίσης, πέρα από τις κλάσεις, είναι εξαιρετικά βοηθητική στη παροχή ενός ευέλικτου σχεδιασμού ιστοσελίδων (responsive web design), συμβάλλοντας σε ένα GUI

το οποίο είναι πιο εύκολα προσβάσιμο από περισσότερα είδη συσκευών του χρήστη. Στον Πίνακα 4.17, απεικονίζεται ένα παράδειγμα αξιοποίησης των κλάσεων της Bootstrap, όπου η κλάση ".card" δημιουργεί ένα στυλιζαρισμένο δοχείο με γέμισμα και σκιά, και η κλάση ".img-fluid" κάνει τις εικόνες να κλιμακώνονται ανάλογα, εξασφαλίζοντας ότι χωράνε στο δοχείο τους χωρίς να ξεχειλίζουν.

```
<!DOCTYPE html>
<!-- Part of index.html -->
<html>
    <head>
        <!-- Some code -->
    </head>
    <body>
        <!-- Some code -->

        <!-- Create New Model Section Card -->
        <div class="card mb-4">
            <div class="card-body">
                <div class="row">
                    <div class="col-md-12">
                        <h2>Create New Model</h2>
                         <p>In this section, users can select an existing dataset or upload their own. After uploading, they can choose parameters, and execute the k-NN algorithm. Once executed, an evaluation appears, displaying metrics per class, max accuracy and average metrics, and the best parameters. To save the model, enter its name and click 'Save model'.</p>
                        <p>The picture on top (below this text), shows the half top of the evaluation window, in which includes the 'Metrics per Class' and the 'Accuracy and Average Metrics' tables.</p>
                    </div>
                    <div class="col-md-12">
                        <img src="./source/image/model_evaluation1.png" alt="model evaluation" class="img-fluid">
                    </div>
                    <div class="col-md-12">
                          <p>The picture below the top, is the other half of the evaluation window, in which shows the 'Best parameters' table, the text input for the model's name, and the 'Save model' button.</p>
                    </div>
                    <div class="col-md-12">
                        <img src="./source/image/model_evaluation2.png" alt="model evaluation" class="img-fluid">
                    </div>
                </div>
            </div>
        </div>
    </body>
</html>
```

Table 4-17 Κώδικας HTML όπου πραγματοποιεί χρήση κλάσεων Bootstrap (‘.card’ και ‘.img-fluid’)

Με την JavaScript, όπου συνεργάζεται στενά με τη βιβλιοθήκη jQuery, πέρα από τον πιο αποδοτικό τρόπο εμφάνισης ιστοσελίδων, τις μετατρέπει επίσης σε διαδραστικές, δηλαδή το περιεχόμενο των σελίδων μεταμορφώνεται ζωντανά, χωρίς ανάγκη για επαναφόρτωση της σελίδας, βάσει των ενεργειών του χρήστη. Ένα σχετικό παράδειγμα αποτελεί η εκτέλεση αιτημάτων στο Web API, όπου τα αποτελέσματα των αιτημάτων εμφανίζονται στο χρήστη μέσω GUI. Για να καλεστούν τα endpoints που κατέχει το Web API, αξιοποιείται η τεχνική AJAX, όπου στην εφαρμογή εφαρμόζεται αυτή η τεχνική με τη μέθοδο της jQuery "$.ajax()". Στο Σχήμα 4.10, απεικονίζονται τα αρχεία της JavaScript, τα οποία συνολικά είναι 14, εκ των οποίων το 1 από αυτά είναι της jQuery με όνομα 'jquery-3.7.1.min.js', και βρίσκονται στο φάκελο '/source/js'.

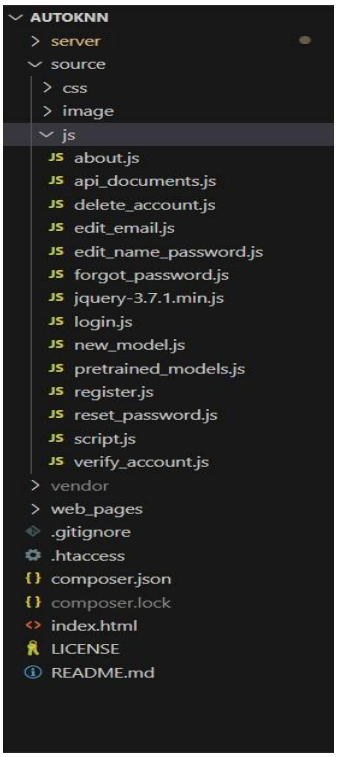


Figure 4-10 Αρχεία JavaScript του AutoKNN

Επιπρόσθετα, στον Πίνακα 4.18, καλείται το endpoint που χρησιμοποιείται για τη διαγραφή του χρήστη από την εφαρμογή. Εισάγεται σχετικό URL στο 'url' κομμάτι, τη μέθοδο DELETE της HTTP στο 'type' κομμάτι, κι οι παράμετροι σε μορφή JSON απαραίτητοι για τη διαγραφή: email και κωδικός πρόσβασης που χρησιμοποίησε στη φόρμα της σελίδας διαγραφής λογαριασμού του AutoKNN. Ύστερα, η jQuery μέσω των 'success' και 'error', καθορίζει κώδικα χειρισμού μετά την εκτέλεση του endpoint διαγραφής χρήστη, ανάλογα με το αν ολοκληρώθηκε η εκτέλεση του με επιτυχία ή όχι. Αν εκτελέστηκε με επιτυχία και το status είναι 'success', εμφανίζεται αντίστοιχο μήνυμα επιτυχίας και διαγράφεται ο χρήστης (ειδικότερα τα στοιχεία του) από το περιηγητή, και μετά από 2 δευτερόλεπτα, ο χρήστης πηγαίνει πίσω στην Αρχική σελίδα. Αν παρ' όλο που η εκτέλεση ήταν επιτυχής και προέκυψε κάποιο άλλο σφάλμα, δηλαδή το status δεν είναι 'success', εμφανίζεται σχετικό μήνυμα και δε διαγράφεται ο χρήστης. Αν δεν είναι επιτυχής η εκτέλεση του endpoint, δε διαγράφεται ο χρήστης κι εμφανίζεται αντίστοιχο μήνυμα σφάλματος. Είναι αναγκαία η σημείωση του γεγονότος ότι τα στοιχεία του χρήστη αποθηκεύονται όταν ο χρήστης συνδέεται στην εφαρμογή, έτσι ώστε να έχει ελεύθερη πρόσβαση σε σελίδες όπου δεν θα είχε αν δεν είχε συνδεθεί, καθώς και στα διαθέσιμα endpoints του Web API. Η διαχείριση των αποθηκευμένων προσωπικών πληροφοριών του χρήστη πραγματοποιείται μέσω της ιδιότητας της JavaScript με ονομασία 'localStorage'.

```
1   $(document).ready(function() {
2       // Part of delete_account.js
3
4       // Some code
5
6       // Handle account deletion confirmation
7       $('#confirmDeleteBtn').on('click', function() {
8           var email = $('#email').val().trim();
9           var password = $('#password').val().trim();
10          var confirmPassword = $('#confirmPassword').val().trim();
```

```
11
12          // API call for deleting the account
13          $.ajax({
14              url: '../server/php/api/delete_account.php',
15              type: 'DELETE',
16              contentType: 'application/json',
17              data: JSON.stringify({
18                  email: email,
19                  password: password,
20                  confirmPassword: confirmPassword,
21                  token: token,
22              }),
23              success: function(response) {
24                  if (response.status === 'success') {
25                      showAlert('success', 'Account deleted successfully. You will be redirected shortly.', '#alertDelete');
26                      setTimeout(function() {
27                          $('#loadDeleteBtn').hide();
28                          $('#confirmBtn').show();
29                          $('#deleteModal').modal('hide');
30                          localStorage.removeItem('token');
31                          localStorage.removeItem('fname');
32                          localStorage.removeItem('lname');
33                          window.location.href = '../index.html'; // Redirect to home page
34                      }, 2000); // Delay for user to see the alert
35                  } else {
36                      showAlert('danger', response.message, '#alertDelete');
37                      $('#loadDeleteBtn').hide();
38                      $('#confirmBtn').show();
39                      $('#deleteModal').modal('hide');
40                  }
41              },
42              error: function(xhr, status, error) {
43                  // Handle error during the kNN execution
44                  console.log('Error:', error);
45                  console.log('XHR object:', xhr);
46                  console.log('Status:', status);
47
48                  // Display specific error message from the server response
49                  const response = xhr.responseJSON;
50                  const message = response && response.message ? response.message : 'An unexpected error occurred.';
51
52                  showAlert('danger', message, '#alertDelete');
53                  $('#loadDeleteBtn').hide();
54                  $('#confirmBtn').show();
55                  $('#deleteModal').modal('hide');
56              }
57          });
58      });
59  });
```

Table 4-18 Κώδικας JavaScript για κλήση API Endpoint για διαγραφής του χρήστη

4.9 GitHub Repository

Ο κώδικας της Web εφαρμογής AutoKNN, όπου χρησιμοποιείται για την υλοποίηση της ίδιας της εφαρμογής, είναι διαθέσιμος και προσβάσιμος από όλους μέσω της διαδικτυακής πλατφόρμας GitHub.

Ο σύνδεσμος του GitHub για το AutoKNN είναι ο ακόλουθος:

https://github.com/KostasKyriakosBatsios/AutoKNN

Κεφάλαιο 5ο: Παρουσίαση του AutoKNN

5.1 Αρχική Σελίδα

Ο χρήστης όταν ακολουθήσει τον σύνδεσμο https://kclusterhub.iee.ihu.gr/autoknn, θα εισέρθει στην Αρχική Σελίδα της διαδικτυακής εφαρμογής AutoKNN. Όπως απεικονίζεται στο Σχήμα 5.1, εμφανίζεται αρχικά ένα μήνυμα το οποίο καλωσορίζει τον χρήστη στην εφαρμογή και τον συνιστά να χρησιμοποιήσει την εφαρμογή στο έπακρο. Εμφανίζεται επάνω το μενού της εφαρμογής, όπου ο χρήστης μπορεί να περιηγηθεί ελεύθερα στις σελίδες που προσφέρει, ενώ ταυτόχρονα μπορεί να κάνει εγγραφή ή είσοδο, εφόσον έχει κάνει εγγραφή με επιτυχία. Επίσης, αναφέρονται πληροφορίες για το πως να έχεις πρόσβαση στις σελίδες αξιοποίησης του αλγορίθμου k-NN (Services), όπου φια να γίνει το κείμενο ορατό, πρέπει ο χρήστης να πατήσει το μπλέ κουμπί που γράφει "How to access the services section?". Επίσης αναφέρει το τι περιέχουν περιληπτικά οι σελίδες Σχετικά (About) και Αξιολόγησε (Rate).

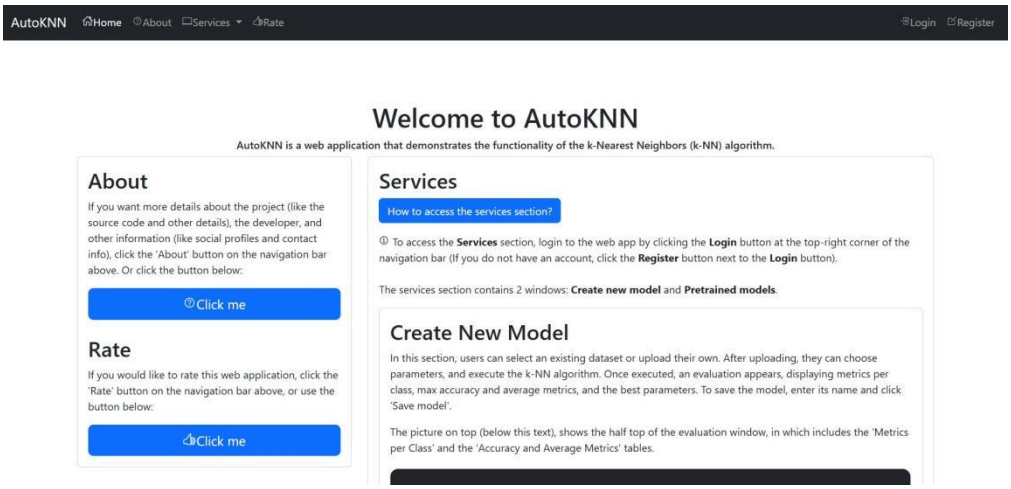


Figure 5-1 Αρχική Σελίδα

Στο Σχήμα 5.2, απεικονίζεται μέσα στο πλαίσιο που αναφέρεται στο μενού Υπηρεσίες (Services), ένα υποπλαίσιο που περιγράφει περιληπτικά τη σελίδα Δημιούργησε ένα μοντέλο (Create a model). Στον Πίνακα 5.1, απεικονίζονται μέσα στο πλαίσιο που αναφέρεται στο μενού Υπηρεσίες (Services), ένα υποπλαίσιο που περιγράφει περιληπτικά τη σελίδα Προ-εκπαιδευμένα Μοντέλα (Pretrained Models).

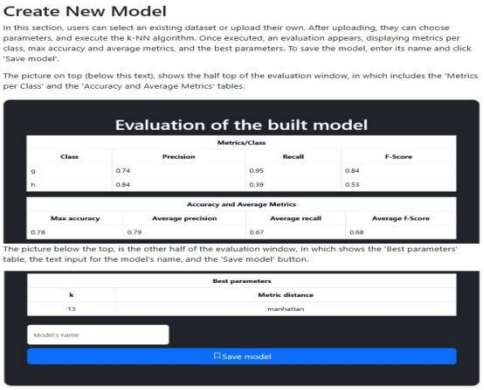


Figure 5-2 Παρουσίαση δυνατοτήτων της σελίδας Create a model

Table 5-1 Παρουσίαση δυνατοτήτων της σελίδας Pretrained Model

5.2 Δημιουργία λογαριασμού και σύνδεση στην εφαρμογή

Πατώντας το κουμπί της Εγγραφής (Register) πάνω δεξιά στο μενού (δείτε το Σχήμα 5.1), εμφανίζεται η σελίδα όπου ο χρήστης μπορεί να δημιουργήσει λογαριασμό. Όπως απεικονίζεται στο Σχήμα 5.3, υπάρχει ένα μαύρο κουμπί πάνω δεξιά που αν ο χρήστης το πατήσει θα κατευθυνθεί στην Αρχική σελίδα, κι από κάτω εμφανίζεται η φόρμα εγγραφής, όπου ο χρήστης καλείται να τη συμπληρώσει εφόσον επιθυμεί να δημιουργήσει λογαριασμό στη σελίδα. Απαιτείται να πληκτρολογήσει το Όνομα (First Name), το Επώνυμο (Last Name), την Ηλεκτρονική Διεύθυνση (Email address) και το Κωδικό πρόσβασης (Password). Αφού τα συμπληρώσει τα συγκεκριμένα πεδία, πρέπει να πατήσει το κουμπί "Register", όπου ύστερα θα του σταλεί ένα ηλεκτρονικό μήνυμα επαλήθευσης λογαριασμού. Εφόσον ο χρήστης ακολουθήσει τις οδηγίες επαλήθευσης του ηλεκτρονικού μηνύματος, θα τον κατευθύνει στη σελίδα Επαλήθευσης, όπως φαίνεται στο Σχήμα 5.4, όπου εμφανίζεται μήνυμα ανάλογα με την επιτυχία της επαλήθευσης και ένα κουμπί "Login". Αφού επαληθεύσει ο χρήστης το λογαριασμό του,

μπορεί να προχωρήσει στην είσοδο της εφαρμογής. Επίσης, αν έχει ήδη λογαριασμό, μπορεί να πατήσει εκεί που λέει "Already signed up? Login!", θα οδηγηθεί στην ιστοσελίδα Εισόδου (Login) του χρήστη, όπως απεικονίζεται στο σχήμα 5.3.

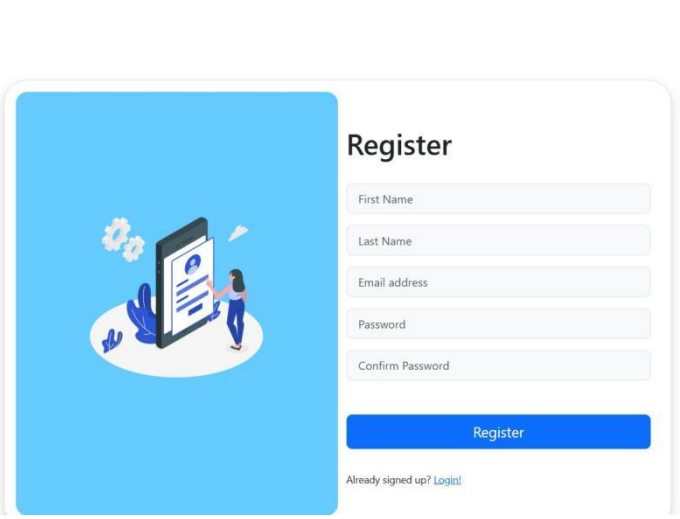


Figure 5-3 Σελίδα Εγγραφής

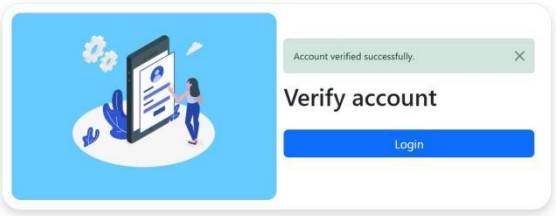


Figure 5-4 Σελίδα Επαλήθευσης

Στο Σχήμα 5.5 παρακάτω, απεικονίζεται η σελίδα Εισόδου του χρήστη, όπου είναι πανομοιότυπη με τη σελίδα Εγγραφής. Απαιτείται ο χρήστης να συμπληρώσει το Email address και το Password που χρησιμοποίησε για να κάνει εγγραφή και για να συνδεθεί πρέπει να πατήσει το κουμπί "Login". Στη περίπτωση που ξέχασε ο χρήστης τον κωδικό του και θέλει να φτιάξει έναν καινούργιο, μπορεί να πατήσει το "Forgot Password?", όπου θα τον ανακατευθύνει στη σελίδα όπου μπορεί ο χρήστης να ανακτήσει το λογαριασμό του. Αν θέλει να κάνει εγγραφή, μπορεί να πατήσει το "New to AutoKNN? Sign up!" για να κατευθυνθεί στη σελίδα Εγγραφής.

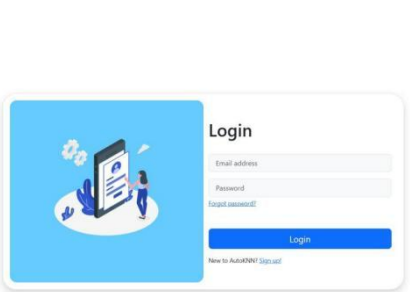


Figure 5-5 Σελίδα Εισοδου

Στο Σχήμα 5.6 παρακάτω, απεικονίζεται η Αρχική σελίδα, στη περίπτωση που είναι συνδεδεμένος στην εφαρμογή. Εμφανίζεται διαφορετικά το μήνυμα καλωσορίσματος (χαιρετάει το χρήστη με το Ονοματεπώνυμο του, όπου είναι το First Name μαζί με το Last Name), και δεν υπάρχουν τα κουμπιά Login και Register πάνω στο μενού, αλλά ένα εικονίδιο που περιέχει μια λίστα διαχείρισης λογαριασμού και οδηγίες που αφορούν το πως να έχει πρόσβαση ο χρήστης στο Services. Επιπρόσθετα, στο Σχήμα 5.7, απεικονίζεται περιληπτικές πληροφορίες και για τις επιλογές της λίστας διαχείρισης λογαριασμού.

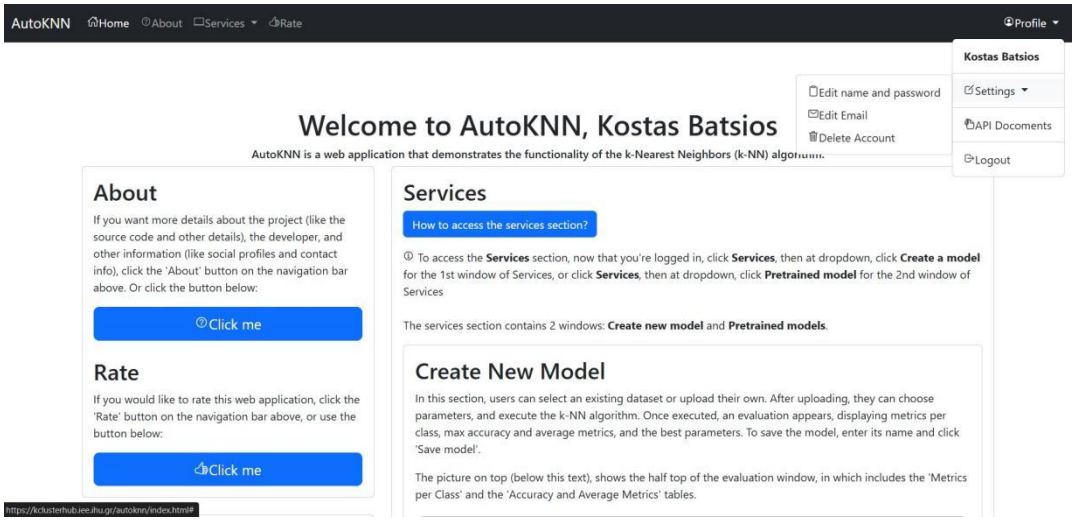


Figure 5-6 Αρχική Σελίδα, ενώ ο χρήστης είναι συνδεδεμένος

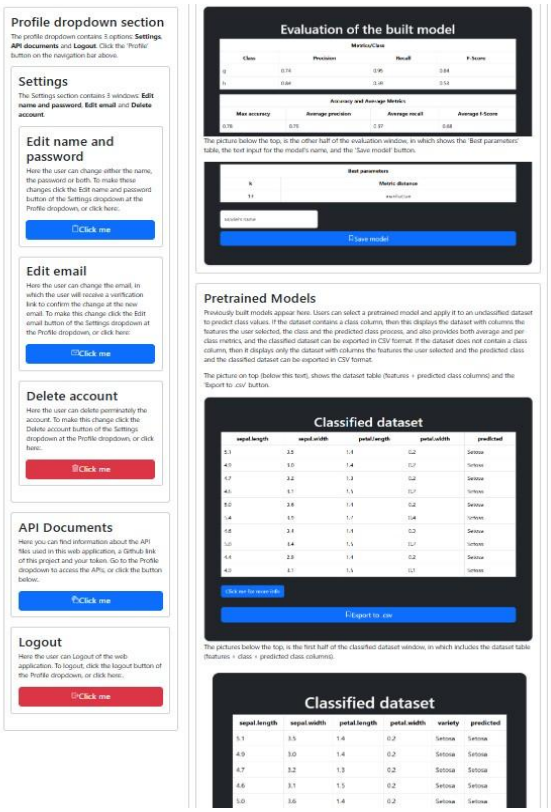


Figure 5-7 Πληροφορίες για τη λίστα διαχείρισης λογαριασμού

5.3 Ανάκτηση και διαχείριση λογαριασμού

Στη περίπτωση όπου υπάρξει απώλεια του κωδικού πρόσβασης, ο χρήστης μπορεί να τον ανακτήσει με το να πατήσει στο "Forgot password?" της σελίδας Login (δείτε Σχήμα 5.5), όπου θα οδηγηθεί στη σελίδα Ανάκτησης κωδικού πρόσβασης. Στο Σχήμα 5.8 παρουσιάζεται η σελίδα επαναφοράς του κωδικού πρόσβασης, όπου απαιτείται στη φόρμα να συμπληρώσει ο χρήστης το Email του. Εφόσον γράψει το Email του και πατήσει το κουμπί "Reset", θα του σταλεί ένα ηλεκτρονικό μήνυμα ανάκτησης, όπου εφόσον ακολουθήσει τις οδηγίες θα τον κατευθύνει στη σελίδα όπου πρέπει να γράψει καινούργιο κωδικό (Σχήμα 5.9). Ωστόσο, ο χρήστης αν δεν επιθυμεί να ανακτήσει τον κωδικό του πρόσβασης, μπορεί να επιστρέψει στη σελίδα Login, με το να πατήσει το "Want to go back? Login!" (Σχήμα 5.8).

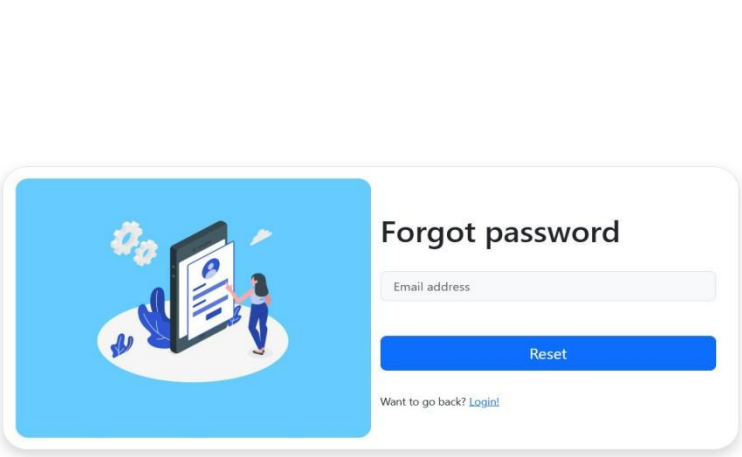


Figure 5-8 Σελίδα Ανάκτησης κωδικού πρόσβασης

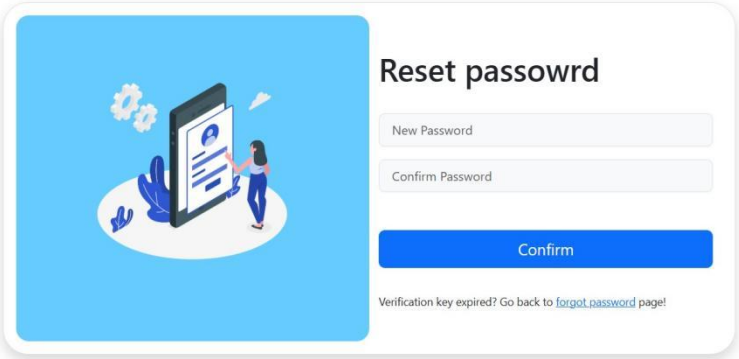


Figure 5-9 Σελίδα ολοκλήρωσης ανάκτησης λογαριασμού

Ο χρήστης μέσω τη λίστα διαχείρισης λογαριασμού (δείτε Σχήμα 5.6), έχει τη δυνατότητα να τροποποιήσει κάποια στοιχεία όσο αφορά το προσωπικό του λογαριασμό. Συγκεκριμένα, μπορεί να τροποποιήσει το Όνομά του (First Name ή/και Last Name), ή/και το Κωδικό Πρόσβασης του. Όπως

απεικονίζεται στο Σχήμα 5.10, ο χρήστης μπορεί να επιλέξει αν θέλει να αλλάξει είτε το όνομα, είτε το κωδικό είτε και τα δύο, κι ανάλογα με το τι επιλέγει ο χρήστης, ενεργοποιούνται τα πεδία της φόρμας που επιθυμεί να τροποποιήσει. Με το που επιλέξει τι επιθυμεί να αλλάξει και συμπληρώσει τα αντίστοιχα πεδία, για να τροποποιηθούν πρέπει να πατήσει το κουμπί "Confirm", κι η αλλαγή γίνεται με το που πατηθεί το κουμπί!

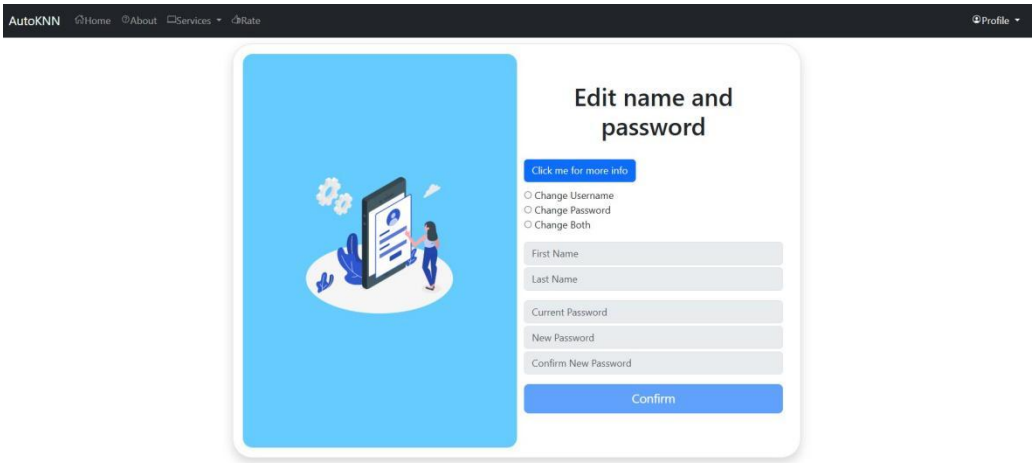


Figure 5-10 Σελίδα Τροποποίησης του Ονόματος ή/και του κωδικού πρόσβασης

Επίσης μέσω της λίστας διαχείρισης λογαριασμού, μπορεί να τροποποιήσει το Email του. Στο Σχήμα 5.11, παρουσιάζεται η φόρμα τροποποίησης του Email. Για να ολοκληρωθεί η αλλαγή του Email, αφού πατήσει το κουμπί "Confirm", θα σταλεί νέο ηλεκτρονικό μήνυμα επαλήθευσης για το νέο Email.

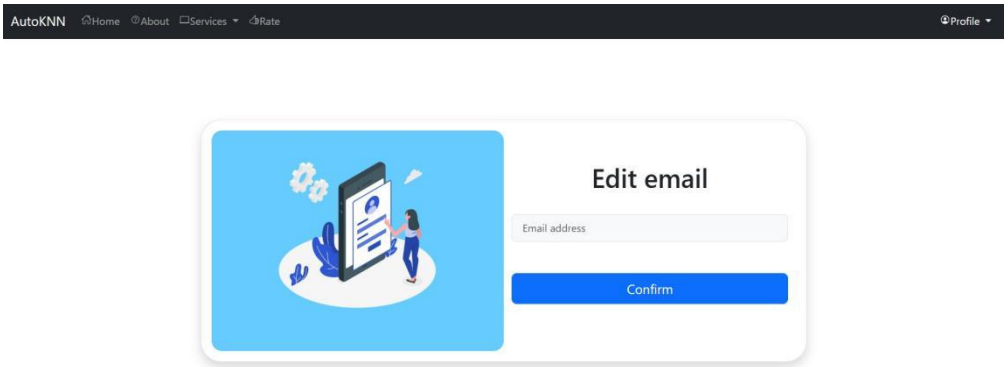


Figure 5-11 Σελίδα Τροποποίησης του Email

Επιπρόσθετα, μπορεί ο χρήστης, εφόσον επιθυμεί, να διαγράψει το λογαριασμό του από την εφαρμογή. Όπως φαίνεται στον Πίνακα 5.2, ο χρήστης πρέπει να συμπληρώσει στη φόρμα το Email του και το Κωδικό πρόσβασης για να ενεργοποιηθεί το κουμπί "Confirm". Εφόσον το πατήσει, θα εμφανιστεί μήνυμα επιβεβαίωσης διαγραφής του λογαριασμού, καθώς αν διαγράψει το λογαριασμό του, δεν μπορεί να το αναιρέσει.

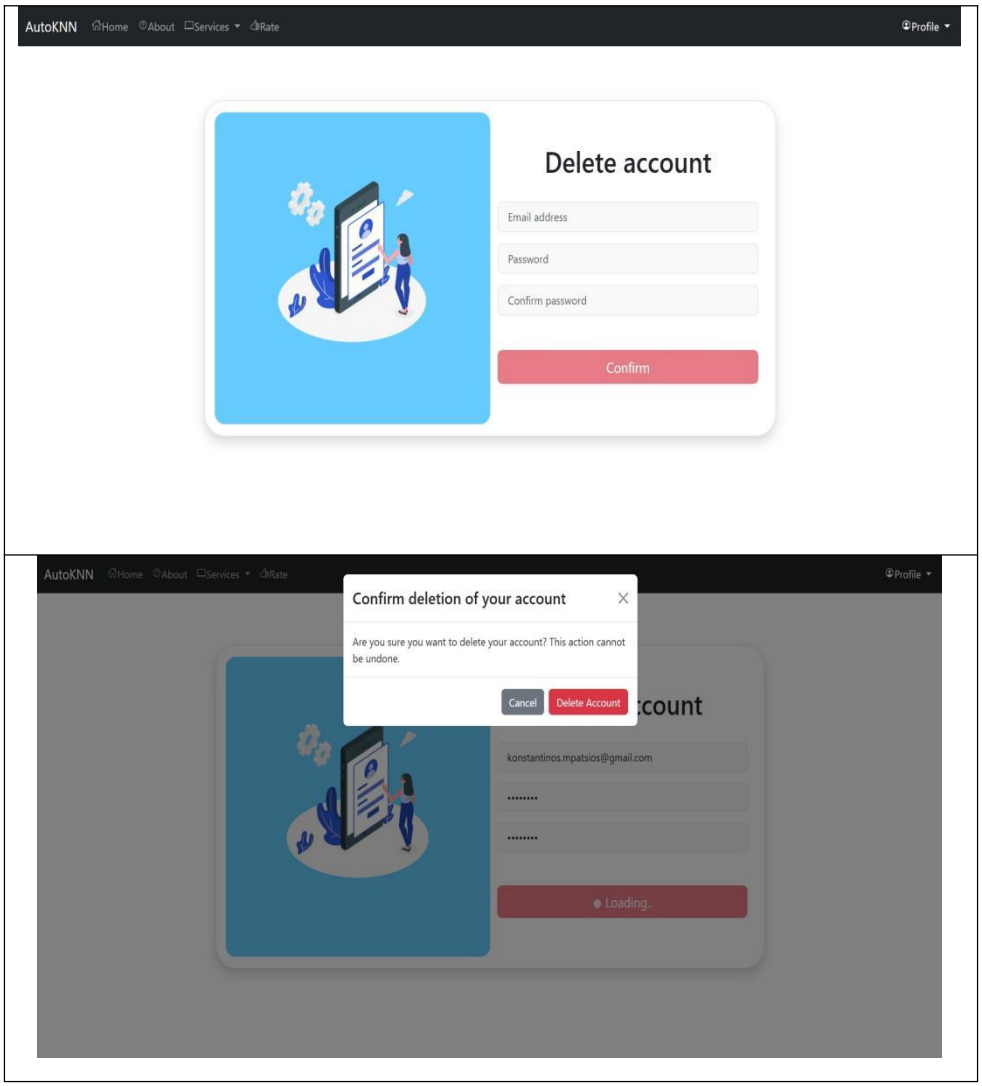


Table 5-2 Σελίδα Διαγραφής Λογαριασμού

5.4 Σελίδα δημιουργίας μοντέλων

Στην επιλογή του μενού που ονομάζεται Υπηρεσίες (Services), υπάρχουν συνολικά 2 επιλογές, οι οποίες είναι οι εξείς: Δημιουργία ενός μοντέλου (Create a model), όπου ο χρήστης έχει τη δυνατότητα να φτιάξει ένα καινούργιο μοντέλου με τη χρήση του k-NN, και Προ-εκπαιδευμένα μοντέλα (Pretrained Models), όπου ο χρήστης μπορεί να αξιοποιήσει το μοντέλο ή μοντέλα που έχει δημιουργήσει. Όπως μπορούμε να παρατηρήσουμε στο Σχήμα 5.12, αυτό είναι το πρώτο παράθυρο που εμφανίζεται όταν ο χρήστης συνδέεται στην εφαρμογή και επιλέγει το παράθυρο της δημιουργίας μοντέλου.

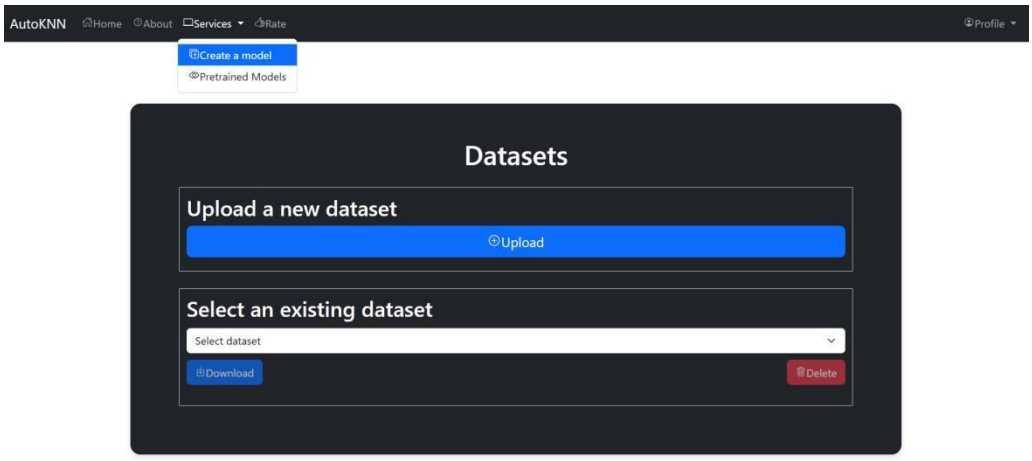


Figure 5-12 1ο παράθυρο της σελίδα Δημιουργίας ενός μοντέλου

Όσον αφορά τα σύνολα δεδομένων, ο χρήστης μπορεί είτε να ανεβάσει ένα καινούργιο σύνολο δεδομένων, είτε μπορεί να χρησιμοποιήσει ένα το οποίο είναι ήδη διαθέσιμο. Στη περίπτωση που ο χρήστης επιλέξει να ανεβάσει ένα νέο σύνολο δεδομένων, πρέπει να πατήσει το μεγάλο μπλέ κουμπί "Upload" που βρίσκεται στο πλαίσιο "Upload a new dataset", κι αφού το πατήσει θα εμφανιστεί ένα παράθυρο για την προώθηση του επιθυμητού συνόλου δεδομένων (Πίνακας 5.3). Όπως απεικονίζεται στο παρακάτω σχήμα, πρέπει το αρχείο να είναι μόνο τύπου CSV (Comma-separated values), μεγέθους μέχρι 10MB, και είναι αναγκαία η επιλογή φακέλου όπου θα αποθηκευτεί το νέο σύνολο δεδομένων. Οι επιλογές φακέλου είναι είτε Δημόσια (Public), όπου ο χρήστης μπορεί να το επιλέξει μόνο εφόσον παρέχεται κατάλληλη άδεια, και Ιδιωτικά (Private).

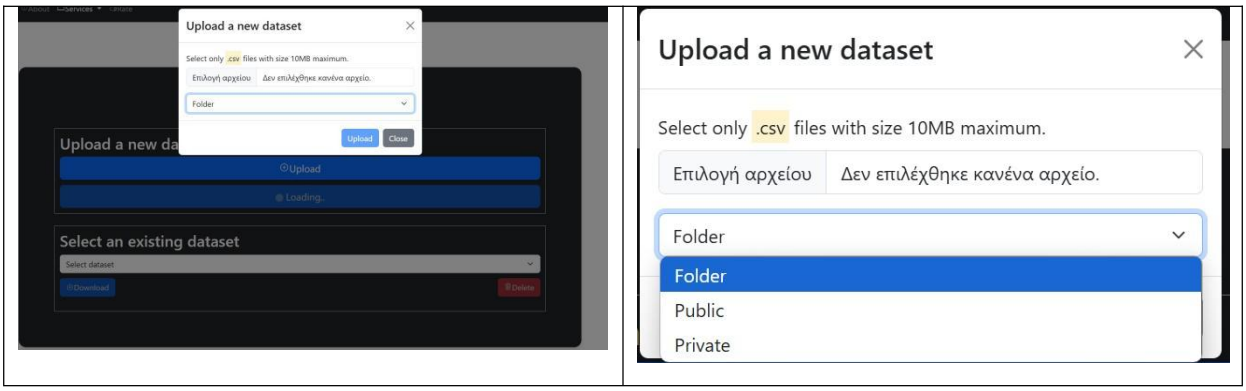


Table 5-3 Παράθυρο μεταφόρτωσης συνόλου δεδομένων εκπαίδευσης

Στο Σχήμα 5.13, απεικονίζεται η περίπτωση όπου ο χρήστης πατήσει πάνω στην άσπρη μπάρα επιλογής "select dataset" του πλαισίου "select an existing dataset", όπου εμφανίζονται όλα τα διαθέσιμα σύνολα δεδομένων σε μορφή λίστας. Από τη στιγμή που ο χρήστης θα επιλέξει ένα επιθυμητό διαθέσιμο σύνολο δεδομένων εκπαίδευσης, μπορεί είτε να το κάνει Λήψη (Download) είτε Διαγραφή (Delete) (για τα κουμπιά δείτε το προηγούμενο Σχήμα 5.12 και Πίνακα 5.3), κάνοντας ο χρήστης κλικ στα αντίστοιχα κουμπιά "Download" και "Delete" που βρίσονται στο ίδιο πλαίσιο με την επιλογή συνόλου δεδομένων. Στη περίπτωση που ο χρήστης επιλέξει διαγραφή, εμφανίζεται ένα μήνυμα προειδοποίησης κι ύστερα αν επιθυμεί να προχωρήσει, θα διαγραφεί. Ταυτόχρονα, όπως φαίνεται στο Σχήμα 5.14, πραγματοποιείται μια προεπισκόπηση των 10 πρώτων σειρών του επιλεγμένου συνόλου δεδομένων εκπαίδευσης.

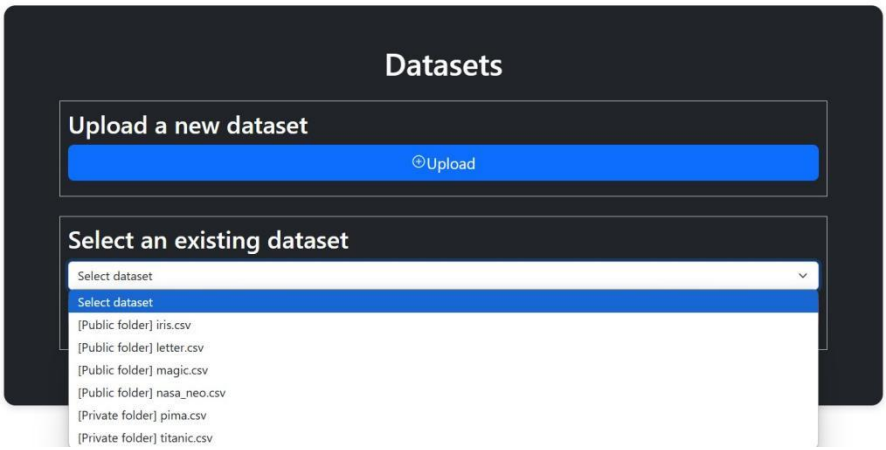


Figure 5-13 Λίστα διαθέσιμων/αποθηκευμένων συνόλων δεδομένων

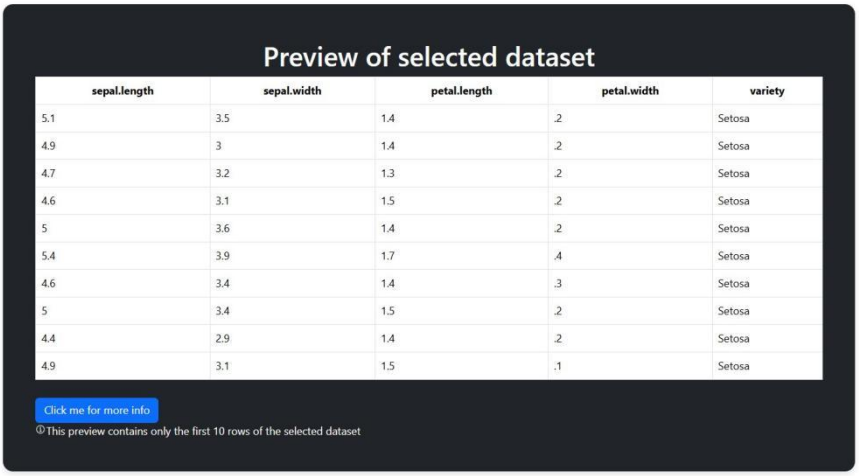


Figure 5-14 Προεπισκόπηση επιλεγμένου συνόλου δεδομένων, 2ο παράθυρο της σελίδα Δημιουργίας ενός μοντέλου

Ύστερα από την επιλογή ενός συνόλου δεδομένων εκπαίδευσης, ακριβώς κάτω από την προεπισκόπηση του συνόλου, εμφανίζεται ένα 3ο παράθυρο όπου ο χρήστης καλείται να ορίσει τις αναγκαίες παραμέτρους για να προχωρήσει στη δημιουργία του μοντέλου με το κατηγοριοποιητή k-NN (δείτε τον παρακάτω Πίνακα 5.4). Οι παράμετροι όπου ο χρήστης καλείται να συμπληρώσει, είναι οι εξής: τα γνωρίσματα (features), που βρίσκονται στο πλαίσιο "Features", η κλάση (class), όπου βρίσκεται στη μπάρα επιλογής του πλαισίου "Class field", το k (δίνεται κι η επιλογή να είναι αυτοματοποιημένο, όπου είναι προκαθορισμένη επιλογή), η μετρική απόστασης (metric distance) (δίνεται κι η επιλογή να είναι αυτοματοποιημένο, όπου είναι προκαθορισμένη επιλογή) και η στρωματοποιημένη δειγματοληψία (stratified sampling) (διαθέσιμη αυτή η παράμετρος μόνο όταν το σύνολο δεδομένων έχει τουλάχιστον 1.000 δεδομένα), όπου και τα 3 βρίσκονται στο πλαίσιο "Parameters' form". Για την καλύτερη κατανόηση των παραμέτρων, μπορεί ο χρήστης αν σύρει το ποντίκι ή πατήσει πάνω στα μπλέ βοηθητικά πλαίσια "Select features for Clasification:", "Select the class field:", "Select values for the parameters of kNN and train test method:", "Stratified Sampling option (recommended):" (διαθέσιμο για μεγάλα σύνολα δεδομένων) και στα λευκά πλαίσια των k και μετρικής απόστασης, να βρει αντίστοιχες περιγραφές για πλήρη κατανόηση.

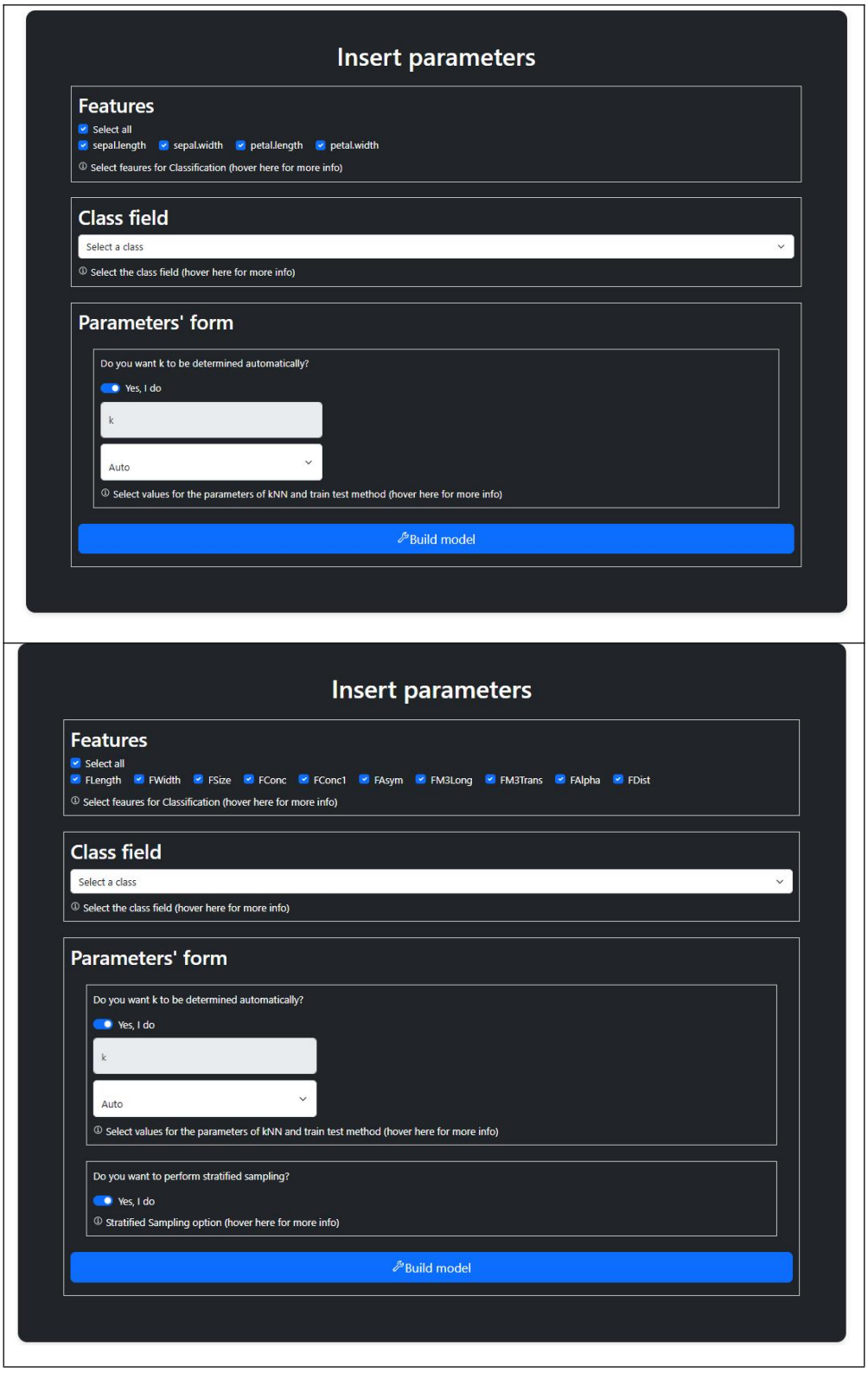


Table 5-4 Καθορισμός παραμέτρων δημιουργίας μοντέλου, 3ο παράθυρο της σελίδα Δημιουργίας ενός μοντέλου

Εφόσον ο χρήστης επιλέξει τις επιθυμητές παραμέτρους και πατήσει το κουμπί "Build model", από κάτω θα εμφανιστεί ένα παράθυρο όπου αφορά τη πρόοδο της διαδικασία δημιουργίας μοντέλου. Δείχνει σε τι κατάσταση βρίσκεται η δημιουργία, κι εμφανίζεται αντίστοιχο μήνυμα. Στον Πίνακα 5.5, εμφανίζονται 2 καταστάσεις: όταν είναι σε εξέλιξη (In Progress) κι όταν έχει ολοκληρωθεί (Completed), όπου ενεργοποιείται και το μπλέ κουμπί "Show Evaluation" για προβολή των

αποτελεσάτων αξιολόγησης. Στη περίπτωση που στο μέλλον ο χρήστης επρόκειτο να ξανατρέξει τις ίδιες παραμέτρους για να φτιάξει ένα μοντέλο με το ίδιο επιθυμητό σύνολο δεδομένων, δεν θα χρειαστεί να περιμένει για τη δημιουργία, καθώς απλά θα τραβήξει τα αποτελέσματα και θα τα εμφανίσει, μαζί και με αντίστοιχο μήνυμα.

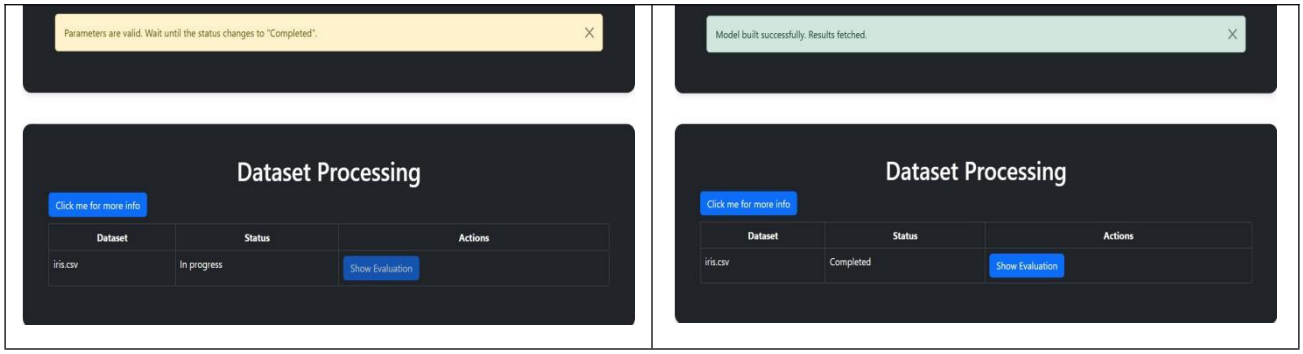


Table 5-5 Πρόοδος δημιουργίας μοντέλου, 4ο παράθυρο της σελίδα Δημιουργίας ενός μοντέλου

Τέλος, όταν ο χρήστης πατήσει το κουμπί "Show Evaluation", το κουμπί μετατρέπεται σε "Hide Evaluation", και γίνεται προβολή του παραθύρου των μετρικών απόδοσης, τόσο για κάθε ετικέτα κλάσης (label), όσο και για το μοντέλο στο σύνολό του. Συγκεκριμένα, όπως απεικονίζεται στο Σχήμα 5.15, εμφανίζετι για κάθε label τα precision, recall και f-score, και για το μοντέλο συνολικά εμφανίζεται μέγιστο accuracy, μέσος όρος των precision, recall και f-score, καθώς και καλύτερες παραμέτρους, δηλαδή καλύτερο k και μετρική απόσταση. Αν το επιθυμεί ο χρήστης, μπορεί να αποθηκεύσει το μοντέλο για μελλοντική χρήση. Για την ορθή αποθήκευση του μοντέλου, ο χρήστης καλείται να συμπληρώσει ένα όνομα για το μοντέλο και να πατήσει το κουμπί "Save model".

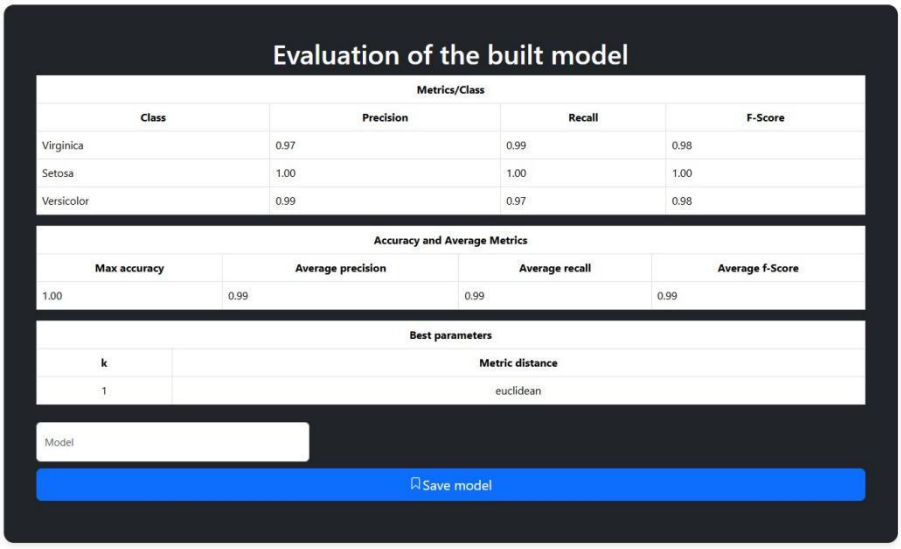


Figure 5-15 Προβολή μετρικών απόδοσης και αποθήκευση μοντέλου, 5ο παράθυρο της σελίδα Δημιουργίας ενός μοντέλου

5.5 Σελίδα χρήσης προεκπαιδευμένων μοντέλων

Η 2η επιλογή από το μενού Υπηρεσίες, όπως προαναφέραμε, ονομάζεται Προ-εκπαιδευμένα Μοντέλα, όπου ο χρήστης μπορεί να αξιοποιήσει τα μοντέλα που δημιούργησε στο παράθυρο Δημιουργία ενός μοντέλου. Όπως απεικονίζεται στο Σχήμα 5.16 παρακάτω, το πρώτο παράθυρο της σελίδας Προ-εκπαιδευμένων Μοντέλων περιέχει μια λίστα στο πλαίσιο "Select a pretrained model" με όλα τα

διαθέσιμα μοντέλα που δημιούργησε ο χρήστης. Όταν ο χρήστης επιλέξει ένα επιθυμητό μοντέλο, μπορεί επίσης να το κάνει λήψη (το μοντέλο είναι σε μορφή '.pkl') και να το διαγράψει πατώντας τα αντίστοιχα κουμπιά "Download" και "Delete" του 1ου παραθύρου. Ταυτόχρονα, εμφανίζεται ένα 2ο παράθυρο (Σχήμα 5.17), όπου προβάλει το περιεχόμενο του μοντέλου, συγκεκριμένα τα featuers στο πλαίσιο "Features" και το class στο πλαίσιο "Class field". Για καλύτερη κατανόηση σχετικά με τα features και το class του μοντέλου, ο χρήστης έχει την ικανότητα να σύρει ή να πατήσει με το ποντίκι πάνω στα αντίστοιχα πλαίσια "Selected features" και "Selected class field" για περαιτέρω πληροφορίες.

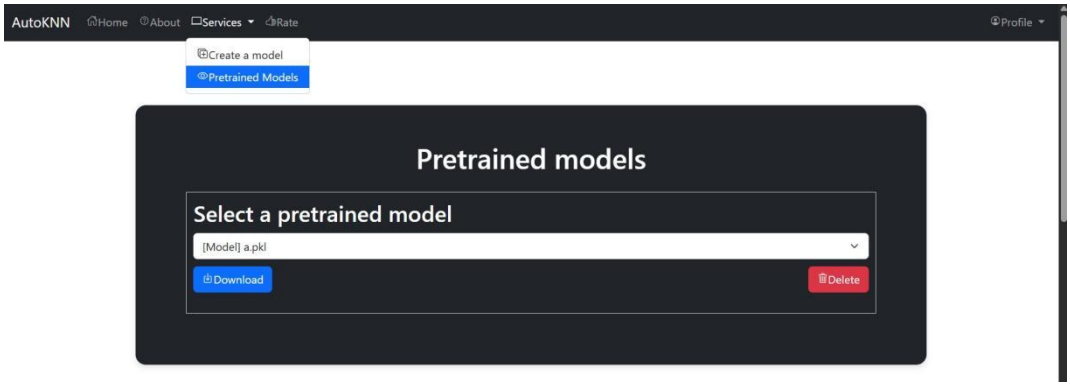


Figure 5-16 Επιλογή διαθέσιμων μοντέλων, 1ο παράθυρο της σελίδα Προ-εκπαιδευμένων Μοντέλων

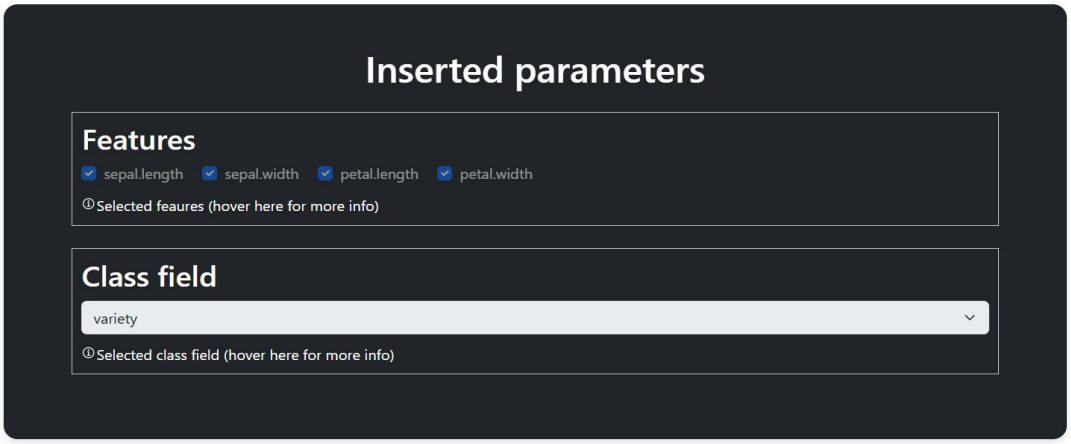


Figure 5-17 Περιεχόμενο επιλεγμένου μοντέλου, 2ο παράθυρο της σελίδα Προ-εκπαιδευμένων Μοντέλων

Όταν ο χρήστης επιλέγει ένα μοντέλο, πέρα από το 2ο παράθυρο που εμφανίζει τα περιεχόμενα του μοντέλου (Σχήμα 5.17), εμφανίζεται από κάτω ένα 3ο παράθυρο, το οποίο αφορά τη μεταφόρτωση και την επιλογή μη κατηγοριοποιημένων συνόλων δεδομένων. Όπως φαίνεται στο Σχήμα 5.18, απεικονίζονται τα πλαίσια μεταφόρτωσης κι επιλογής. Το πλαίσιο μεταφόρτωσης, που έχει τίτλο "Upload an unclassified dataset", είναι παρόμοιο με εκείνου της σελίδας Δημιουργίας μοντέλου (Σχήμα 5.12 και Πίνακας 5.3), με τη κύρια διαφορά να είναι ότι σε εκείνο της σελίδας Προ-εκπαιδευμένων Μοντέλων, δεν υπάρχει επιλογή ανάμεσα σε τύπου φακέλων, καθώς όλα αποθηκεύονται κάτω από το φάκελο "Unclassified dataset". Μπορεί ο χρήστης να ανεβάσει ένα σύνολο δεδομένων το οποίο είναι κατηγοριοποιημένο, δηλαδή περιέχει τη στήλη της κλάσης. Επίσης, με το που ο χρήστης κάνει μεταφόρτωση κι επιλέξει ένα επιθυμητό μη κατηγοριοποιημένο σύνολο δεδομένων, μπορεί να το κάνει λήψη είτε να το διαγράψει.

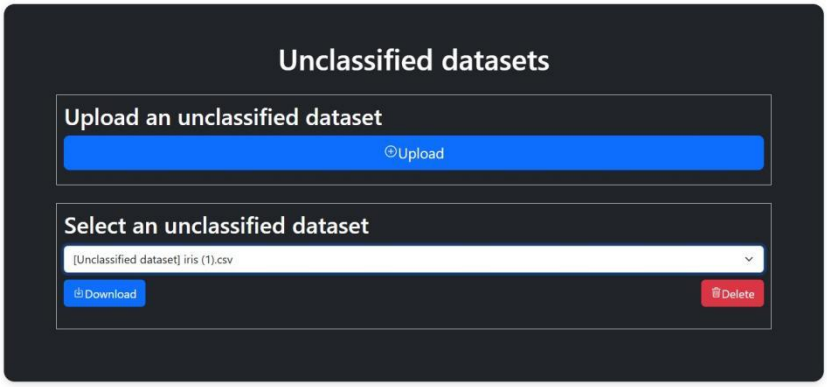
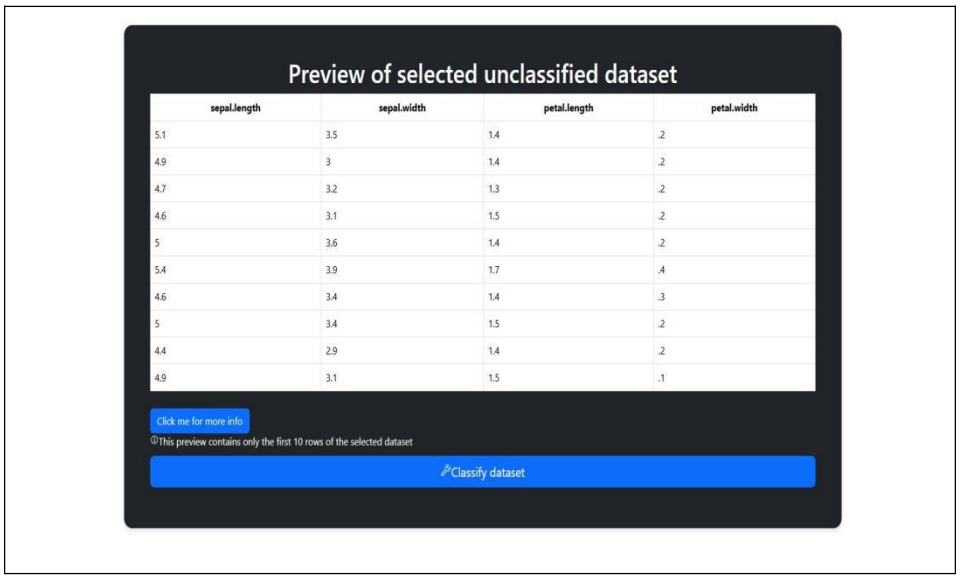


Figure 5-18 Μεταφόρτωση κι επιλογή μη κατηγοριοποιημένων συνόλων δεδομένων, 3ο παράθυρο της σελίδα Προ-εκπαιδευμένων Μοντέλων

Παράλληλα με την επιλογή ενός μη κατηγοριοποιημένου συνόλου δεδομένων, εμφανίζεται ένα 4ο παράθυρο που προβάλλει τις 10 πρώτες σειρές του επιλεγμένου συνόλου. Όπως παρατηρείται στον Πίνακα 5.6, πέρα από τη διαθέσιμη προεπισκόπηση του επιλεγμένου συνόλου δεδομένων, μπορεί να προχωρήσει στη διαδικασία της κατηγοριοποίησης των δεδομένων, πατώντας το κουμπί "Classify dataset". Για να μπορέσει να προχωρήσει ο χρήστης στη διαδικασία της κατηγοριοποίησης, το σύνολο δεδομένων θα πρέπει να ταιριάζει με το επιλεγμένο μοντέλο (δηλαδή να ταιριάζουν τα features και το class).



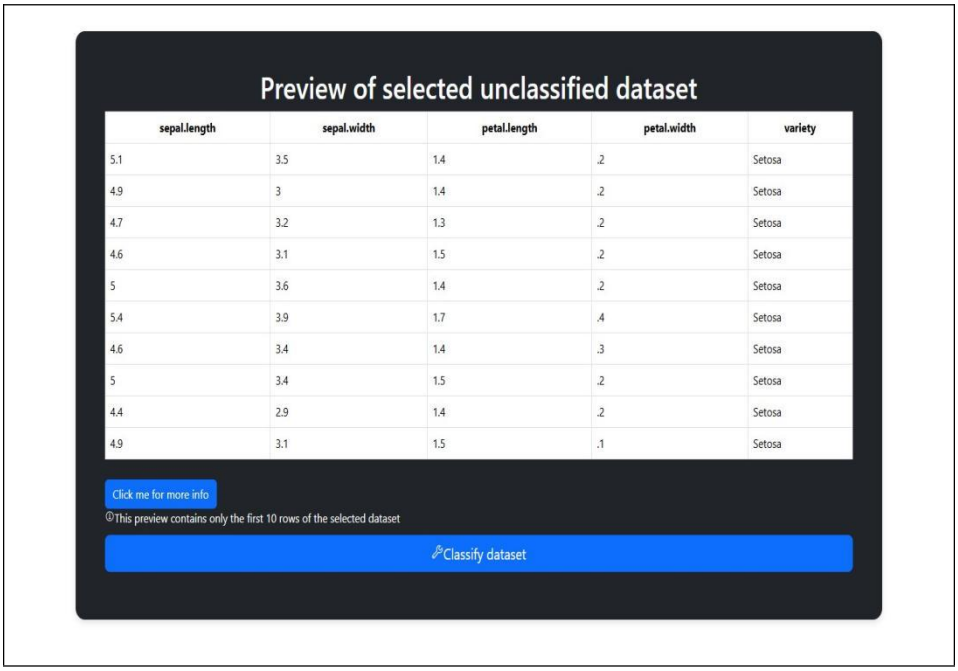


Table 5-6 Προεπισκόπηση επιλεγμένου συνόλου, 4ο παράθυρο της σελίδα Προ-εκπαιδευμένων Μοντέλων (Μη κατηγοριοποιημένο το 1ο πάνω, κατηγοριοποιημένο το 2ο κάτω)

Εφόσον ολοκληρωθεί η κατηγοριοποίηση, εμφανίζεται ένα 5ο παράθυρο, όπου δείχνει τα αποτελέσματα της κατηγοριοποίησης και ένα μπλέ κουμπί κάτω που γράφει "Export to .csv", το οποίο εξάγει πλήρως τα αποτελέσματα αυτής της διαδικασίας σε μορφή '.csv'. Στην περίπτωση που ο χρήστης επέλεξε ένα σύνολο δεδομένων το οποίο είναι όντως μη κατηγοριοποιημένο, στο 5ο παράθυρο εμφανίζεται ο πίνακας του μη κατηγοριοποιημένου συνόλου δεδομένων, ο οποίος έχει ως στήλες τα features, ωστόσο προστίθεται και μια επιπλέον στήλη με το όνομα predicted (δες το Σχήμα 5.19 παρακάτω). Η στήλη predicted δείχνει τα αποτελέσματα της κατηγοριοποίησης, δηλαδή με βάσει τις τιμές των features, σε ποια κλάση κατηγοριοποιήθηκαν.

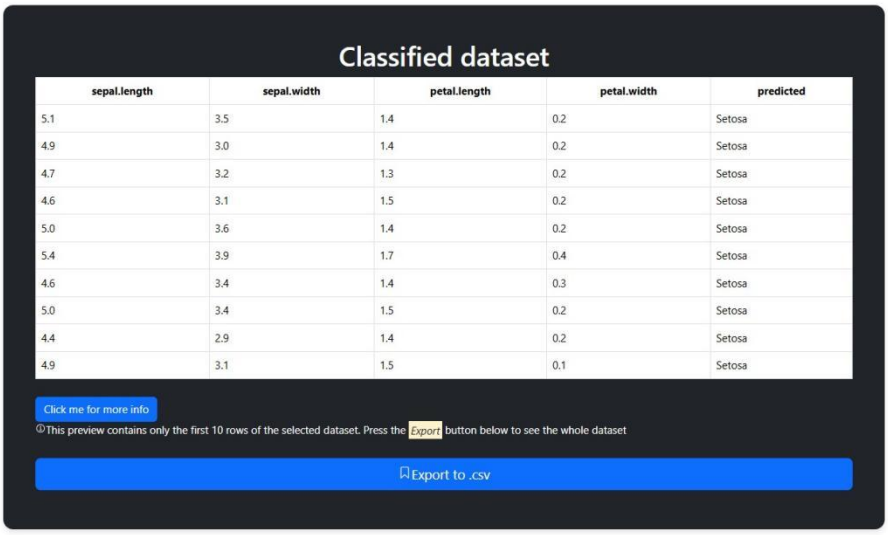


Figure 5-19 Αποτελέσματα κατηγοριοποίησης μη κατηγοριοποιημένου συνόλου δεδομένων, 5ο παράθυρο της σελίδα Προ-εκπαιδευμένων Μοντέλων

Στην αντίθετη περίπτωση που ο χρήστης επέλεξε ένα σύνολο δεδομένων το οποίο είναι κατηγοριοποιημένο (δηλαδή περιέχει τη στήλη class), πέρα από το πίνακα που μαζί με τα features και

class, περιέχει επιπλέον τη στήλη predicted, περιέχει και αποτελέσματα μετρικών ανά κλάση, συγκεκριμένα για κάθε label προβάλλει το precision, το recall και το f-score, και μέσος όρος μετρικών, όπου αποτελείται ο μέσος όρος από τα accuracy, precision, recall και f-score (δες το Σχήμα 5.26 παρακάτω). Για να μπορέσει ο χρήστης να δει τα αποτελέσματα των μετρικών, καλείται να πατήσει το μπλέ κουμπί "Show the metrics/class" για να δει τις μετρικές ανά κλάση, και το μπλέ κουμπί "Show the average metrics" για το μέσο όρο των μετρικών.

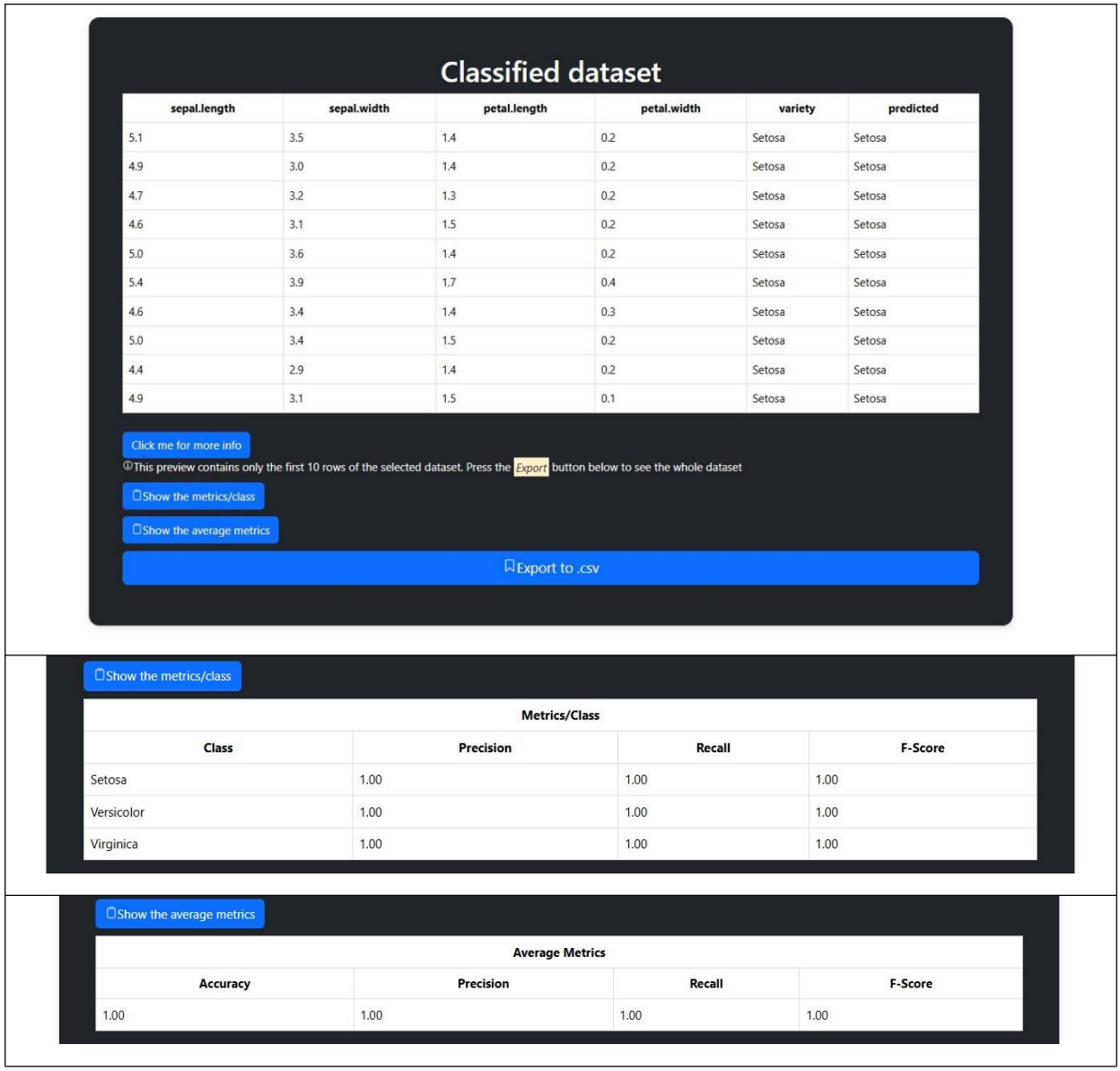


Table 5-7 Αποτελέσματα κατηγοριοποίησης κατηγοριοποιημένου συνόλου δεδομένων, 5ο παράθυρο της σελίδα Προ-εκπαιδευμένων Μοντέλων

5.6 Σελίδα τεκμηρίωσης του Web API

Στη περίπτωση που ο χρήστης ενδιαφέρεται να μάθει πληροφορίες σχετικά με το API αυτής της διαδικτυακής (Web) εφαρμογής, μπορεί να μεταφερθεί στη συγκεκριμένη σελίδα με το να πατήσει πάνω δεξιά στο μενού που λέει "Profile", και να πατήσει εκεί μέσα την επιλογή "API Documents". Όπως φαίνεται στο Σχήμα 5.20, η συγκεκριμένη σελίδα περιέχει 2 πλαίσια: το πρώτο που γράφει "Your personal token", όπου ο χρήστης μπορεί να αντιγράψει και να δει το προσωπικό του token, το οποίο τον επιτρέπει να έχει πρόσβαση στα API Endpoints, και το 2ο πλαίσιο που γράφει "API

Endpoints of AutoKNN", όπου περιέχονται όλα τα API Endpoints της Web εφαρμογής AutoKNN. Στα API Endpoints υπάρχουν 4 είδη λειτουργίας όπου υπάρχουν αυτά τα Endpoints: του Χρήστη (User), των Συνόλων δεδομένω (Datasets), των Μοντέλων (Models) και της Κατηγοριοποίησης (Classification). Στο Σχήμα 5.21 παρακάτω, αν ο χρήστης πατήσει πάνω σε ένα από τα 4 είδη, εμφανίζεται μια λίστα με όλα τα Endpoints που χρησιμοποιούνται για την επιλεγμένη λειτουργία. Ύστερα, κάνοντας κλικ σε ένα από τα Endpoints, εμφανίζεται μια σύντομη περιγραφή της λειτουργίας του, καθώς και παραδείγματα κλήσης κι απόκρισής του.

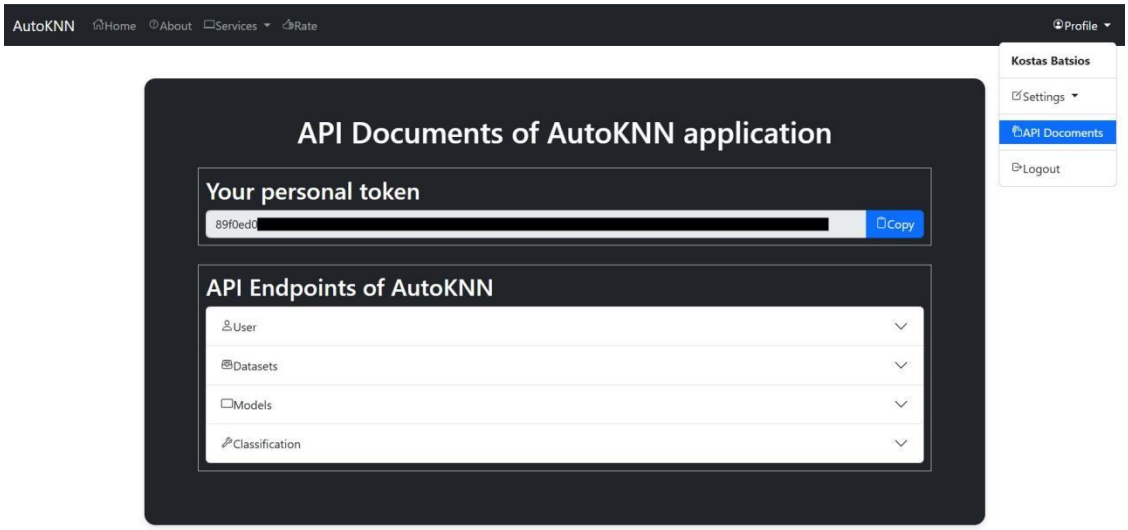


Figure 5-20 Σελίδα τεκμηρίωσης του Web API

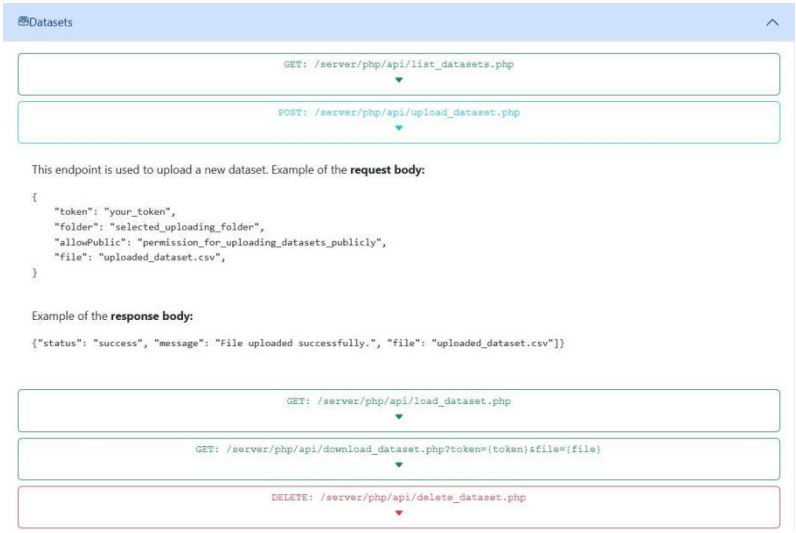


Figure 5-21 Προβολή περιγραφής και παραδειγμάτων κλήσης κι απόκρισης του API Endpoint

Κεφάλαιο 6ο: Αξιολόγηση εφαρμογής

6.1 Αξιολόγηση απόδοσης

Η αξιολόγηση απόδοσης της εφαρμογής εξαρτάται από το χρονικό διάστημα της CPU (Central Processing Unit) που χρειάστηκε για την εκτέλεση του αλγορίθμου κατηγοριοποίησης k-NN μαζί με τη μέθοδο train test split. Ανάλογα με το σύνολο δεδομένων που επιλέγει ο χρήστης, τα πεδία features και class που ορίζει, κι αν έχει επιλέξει στρωματοποιημένη δειγματοληψία (διαθέσιμη για σύνολα δεδομένων με τουλάχιστον 1.000 δεδομένα, κι εφαρμόζεται στο σύνολο εκπαίδευσης με μέγεθος εκπαίδευσης 1.000), απαιτείται κι ανάλογο χρονικό διάστημα εκτέλεσης από τη CPU.

Για τη μέτρηση του χρόνου που πραγματοποιεί η CPU για τους k-NN και train test split, αξιοποιείται η βιβλιοθήκη της Python "psutil}". Όπως φαίνεται στο Πίνακα 6.1 παρακάτω, με αυτό τον τρόπο πραγματοποιείται χρονομέτρηση της CPU.

```
# Part of a test file called knn.py

# Some code

# Execution time (specifically CPU time)
start_time = psutil.Process().cpu_times().user

best_k, best_distance, best_p, max_accuracy, best_precision, best_recall, best_f1 = knn(k, p, distance, strat)

end_time = psutil.Process().cpu_times().user

execution_time = end_time - start_time

print("Train-Test Split Execution Time:", execution_time, "seconds")
```

Table 6-1 Κώδικας Python για χρονομέτρηση της CPU.

Για τη μελέτη του χρόνου εκτέλεσης, χρησιμοποιήθηκαν 2 μεγάλα σύνολα δεδομένων: το magic.csv (19.020 δεδομένα) και το letter.csv (20.000 δεδομένα). Εκτελέστηκαν οι k-NN και train test split 2 φορές για κάθε σύνολο δεδομένων εκπαίδευσης. Μια φορά όπου ο χρήστης έχει επιλέξει να χρησιμοποιήσει στρωματοποιημένη δειγματοληψία (δηλαδή stratified sampling = True), και μια φορά όπου ο χρήστης δεν έχει επιλέξει να αξιοποιήσει στρωματοποιημένη δειγματοληψία (δηλαδή stratified sampling = False). Στα Σχήματα 6.1 και 6.2, απεικονίζονται οι χρόνοι εκτέλεσης για το σύνολο magic.csv, όπου στο 6.1 ο χρήστης επιλέγει στρωματοποιημένη δειγματοληψία, ενώ στο 6.2 δεν επιλέγει στρωματοποιημένη δειγματοληψία.

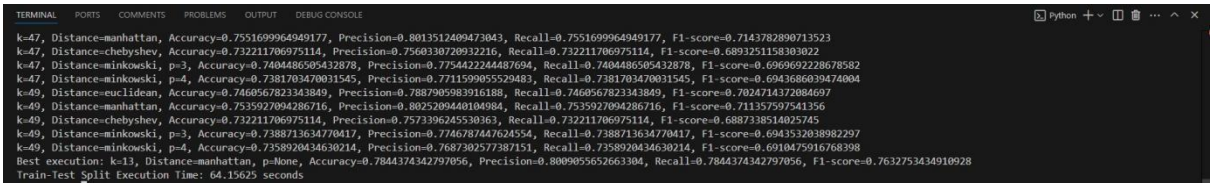


Figure 6-1 Χρόνος εκτέλεσης για magic.csv, όπου stratified sampling = True

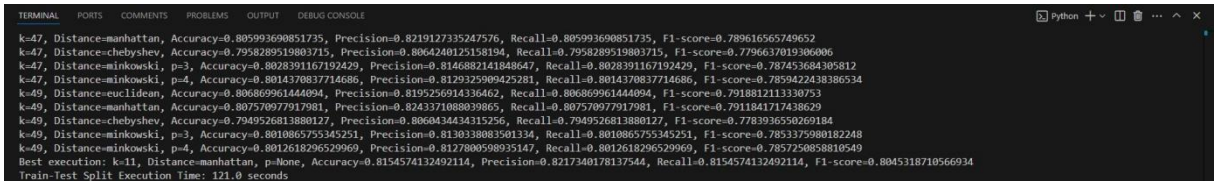


Figure 6-2 Χρόνος εκτέλεσης για magic.csv, όπου stratified sampling = False

Στα Σχήματα 6.3 και 6.4, απεικονίζονται οι χρόνοι εκτέλεσης για το σύνολο letter.csv, όπου στο 6.3 ο χρήστης επιλέγει στρωματοποιημένη δειγματοληψία, ενώ στο 6.4 δεν επιλέγει στρωματοποιημένη δειγματοληψία.

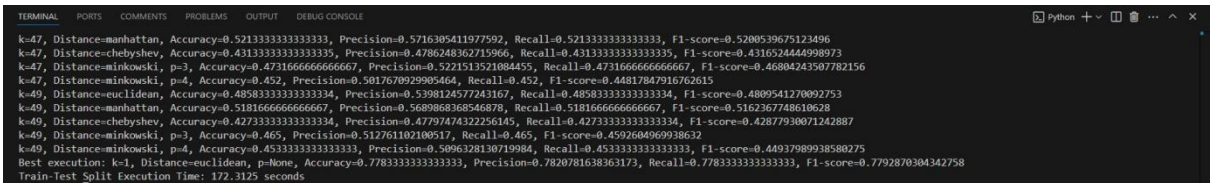


Figure 6-3 Χρόνος εκτέλεσης για letter.csv, όπου stratified sampling = True

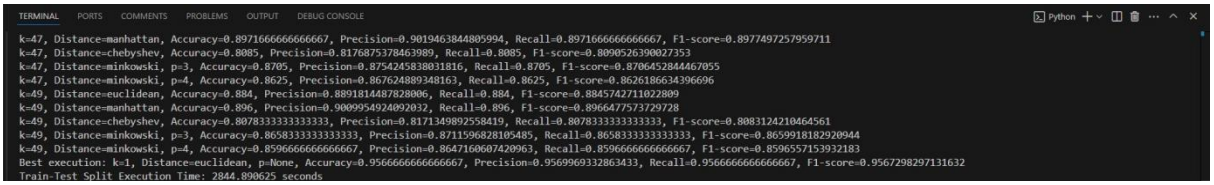


Figure 6-4 Χρόνος εκτέλεσης για letter.csv, όπου stratified sampling = False

6.2 Αξιολόγηση εμπειρίας χρήσης

6.2.1 Εισαγωγή στο SUS

Ένα ακόμα σημαντικό κομμάτι της διαδικτυακής εφαρμογής είναι η εμπειρία του χρήστη. Είναι σημαντικό ο χρήστης να έχει όσο το δυνατόν ευχάριστη και σε γενικές γραμμές καλή εμπειρία κατά τη χρήση του AutoKNN. Για να κατανοήσουμε την εμπειρία που βιώνει ο χρήστης και για τη πιθανή βελτίωση της εφαρμογής, είναι αναγκαία η εφαρμογή μιας αξιολόγησης της.

Για τη διαδικασία της αξιολόγησης, δημιουργήθηκε αντίστοιχο ερωτηματολόγιο με τη πλατφόρμα Google Forms, όπου οι χρήστες του AutoKNN έχουν πρόσβαση μέσω της εφαρμογής, αν πατήσουν πάνω στο μενού το κουμπί "Rate". Η πλειοψηφία των χρηστών είναι φοιτητές του Τμήματος Μηχανικών Πληροφορικής και Ηλεκτρονικών Συστημάτων, όπου οι περισσότεροι έχουν βασικές γνώσεις από το μάθημα Οργάνωση Δεδομένων και Εξόρυξη Πληροφορίας.

Η Κλίμακα Ευχρηστίας Συστήματος (System Usability Scale - SUS) είναι ένα αναγνωρισμένο και αξιόπιστο εργαλείο για την αξιολόγηση της ευχρηστίας ενός συστήματος, το οποίο δημιουργήθηκε αρχικά από τον John Brooke το 1986 [39]. Περιλαμβάνει 10 τυποποιημένες ερωτήσεις που αξιολογούν τη χρηστικότητα, τη λειτουργικότητα και τη συνολική ικανοποίηση από ένα προϊόν ή σύστημα [40]. Οι ερωτήσεις αυτές εναλλάσσονται μεταξύ θετικών και αρνητικών δηλώσεων για να ελαχιστοποιηθεί η προκατάληψη. Οι 10 ερωτήσεις του SUS είναι:

1. Νομίζω ότι θα ήθελα να χρησιμοποιώ αυτό το σύστημα συχνά.
2. Βρήκα το σύστημα περιττά πολύπλοκο.

3. Νομίζω ότι το σύστημα ήταν εύκολο στη χρήση.
4. Νομίζω ότι θα χρειαζόμουν την υποστήριξη ενός τεχνικού για να χρησιμοποιήσω αυτό το σύστημα.
5. Διαπίστωσα ότι οι διάφορες λειτουργίες αυτού του συστήματος ήταν καλά ενσωματωμένες.
6. Νομίζω ότι υπήρχε πολύ μεγάλη ασυνέπεια σε αυτό το σύστημα.
7. Φαντάζομαι ότι οι περισσότεροι άνθρωποι θα μάθουν να χρησιμοποιούν αυτό το σύστημα πολύ γρήγορα.
8. Βρήκα το σύστημα πολύ δύσχρηστο.
9. Αισθάνθηκα πολύ σίγουρος για τη χρήση του συστήματος.
10. Έπρεπε να μάθω πολλά πράγματα πριν μπορέσω να ξεκινήσω με αυτό το σύστημα.

Κάθε μία από αυτές τις ερωτήσεις απαντάται με τη χρήση μιας 5-βάθμιας κλίμακας Likert, όπου οι ερωτώμενοι επιλέγουν ένα από τα ακόλουθα:

- 1 - Διαφωνώ απόλυτα
- 2 - Διαφωνώ
- 3 - Ούτε συμφωνώ ούτε διαφωνώ
- 4 - Συμφωνώ
- 5 - Συμφωνώ απόλυτα

6.2.2 Αποτελέσματα του SUS

Μέχρις στιγμής έχουν απαντήσει 15 χρήστες στο ερωτηματολόγιο, κι αναλυτικά τα αποτελέσματα παρουσιάζονται παρακάτω σε μορφή διαγραμμάτων, όπου ο κάθετος άξονας αντιπροσωπεύει το μέγιστο αριθμό χρηστών που απάντησαν σε μια ερώτηση, κι ο οριζόντιος άξονας αντιπροσωπεύει τις 5 απαντήσεις.

Για το 1ο ερώτημα, όπως απεικονίζεται στο Σχήμα 6.5, οι πλειοψηφία των χρηστών απάντησε "Συμφωνώ Απόλυτα" και "Συμφωνώ", δηλαδή επιθυμούν να χρησιμοποιούν συχνά το AutoKNN.

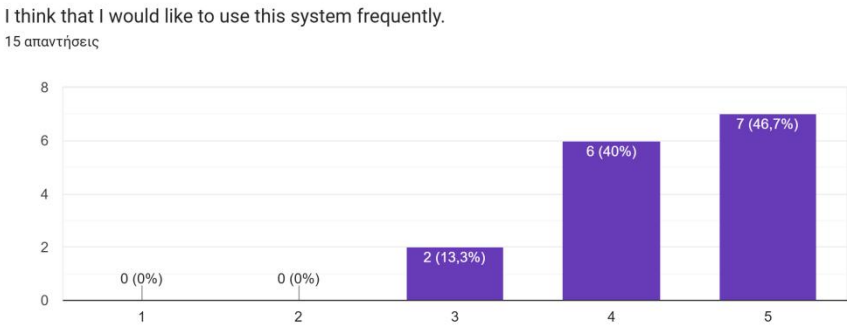


Figure 6-5 Διάγραμμα ροής απαντήσεων 1ης ερώτησης

Για το 2ο ερώτημα, όπως απεικονίζεται στο Σχήμα 6.6, οι περισσότεροι χρήστες απάντησαν "Διαφωνώ απόλυτα", δηλαδή δε θεωρούν το AutoKNN ότι είναι περιττά πολύπλοκο.

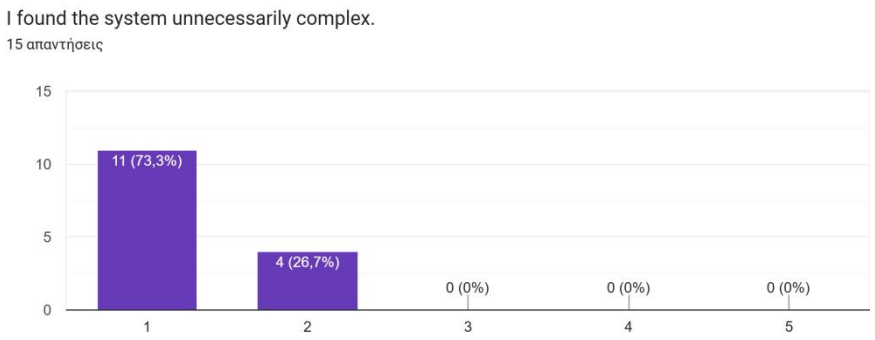


Figure 6-6 Διάγραμμα ροής απαντήσεων 2ης ερώτησης

Για το 3ο ερώτημα, όπως απεικονίζεται στο Σχήμα 6.7, σχεδόν όλοι οι χρήστες απάντησαν "Συμφωνώ απόλυτα", δηλαδή θεωρούν το AutoKNN ότι είναι εύκολο στη χρήση.

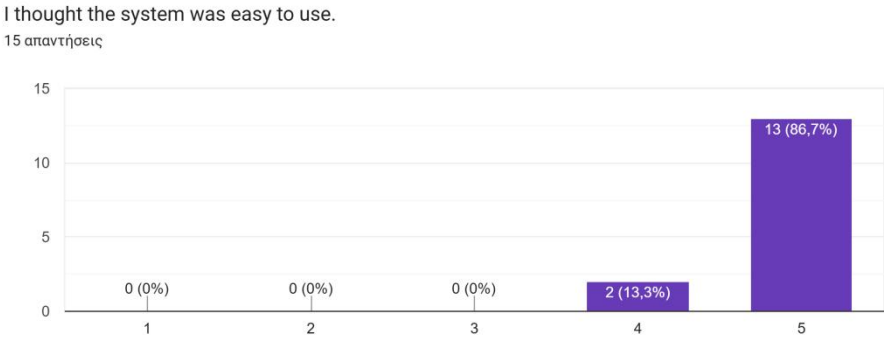


Figure 6-7 Διάγραμμα ροής απαντήσεων 3ης ερώτησης

Για το 4ο ερώτημα, όπως απεικονίζεται στο Σχήμα 6.8, οι πλειοψηφία των χρηστών απάντησε "Διαφωνώ απόλυτα" και "Διαφωνώ", δηλαδή δε χρειάζονται βοήθεια για τη χρήση του AutoKNN.

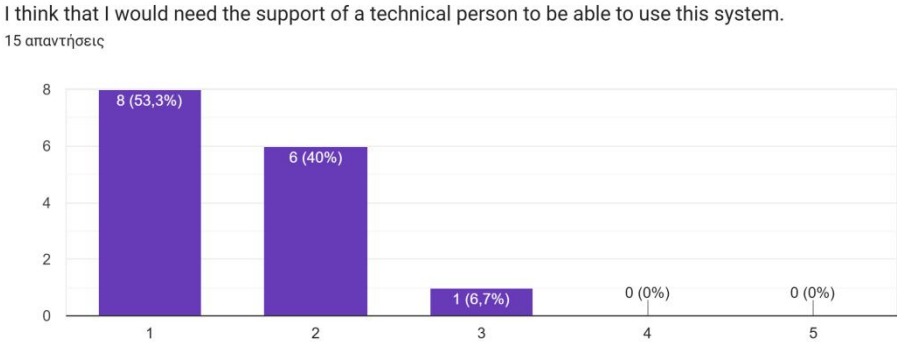


Figure 6-8 Διάγραμμα ροής απαντήσεων 4ης ερώτησης

Για το 5ο ερώτημα, όπως απεικονίζεται στο Σχήμα 6.9, οι πλειοψηφία των χρηστών απάντησε "Συμφωνώ απόλυτα", δηλαδή πιστεύουν πως οι λειτουργίες του AutoKNN είναι καλά ενσωματωμένες.

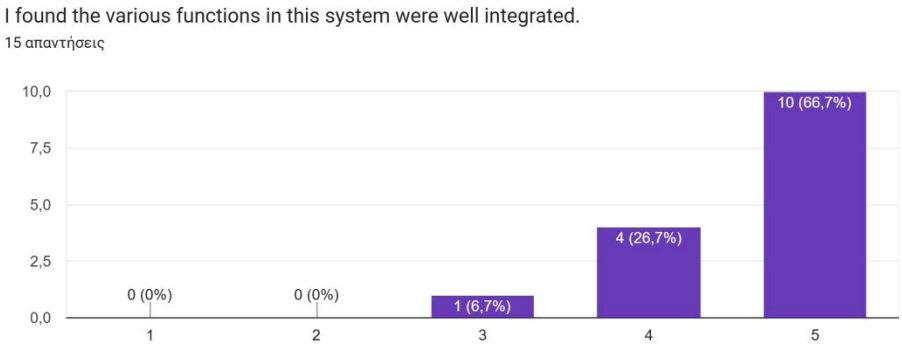


Figure 6-9 Διάγραμμα ροής απαντήσεων 5ης ερώτησης

Για το 6ο ερώτημα, όπως απεικονίζεται στο Σχήμα 6.10, σχεδόν όλοι οι χρήστες απάντησαν "Διαφωνώ απόλυτα", δηλαδή πιστεύουν ότι στο AutoKNN δεν υπήρξε μεγάλη ασυνέπεια.

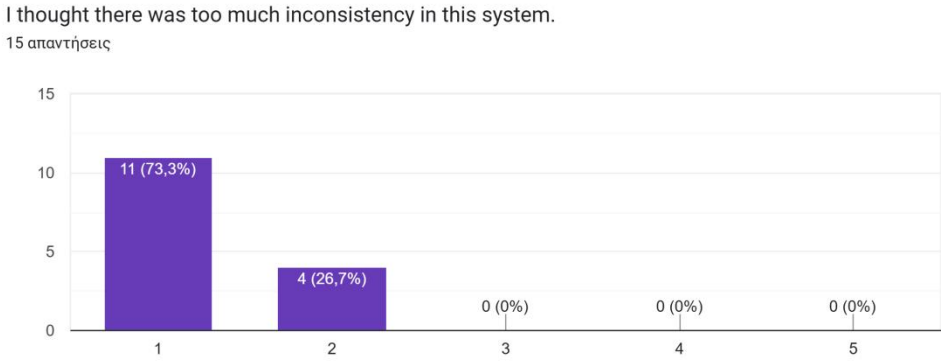


Figure 6-10 Διάγραμμα ροής απαντήσεων 6ης ερώτησης

Για το 7ο ερώτημα, όπως απεικονίζεται στο Σχήμα 6.11, οι απαντήσεις των χρηστών διαμοιράστηκαν στα "Συμφωνώ απόλυτα" και "Συμφωνώ", δηλαδή θεωρούν πως πολλοί θα μάθουν γρήγορα να αξιοποιούν την εφαρμογή στο έπακρο.

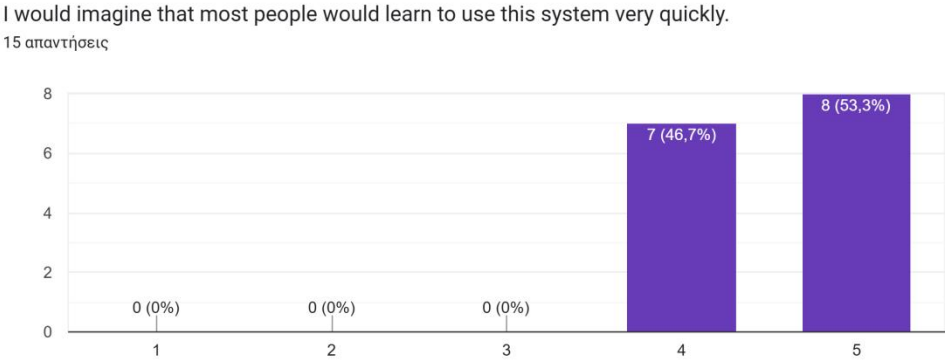


Figure 6-11 Διάγραμμα ροής απαντήσεων 7ης ερώτησης

Για το 8ο ερώτημα, όπως απεικονίζεται στο Σχήμα 6.12, οι περισσότεροι χρήστες απάντησαν "Διαφωνώ απόλυτα", δηλαδή δε θεωρούν το AutoKNN δύσκολο στη χρήση.

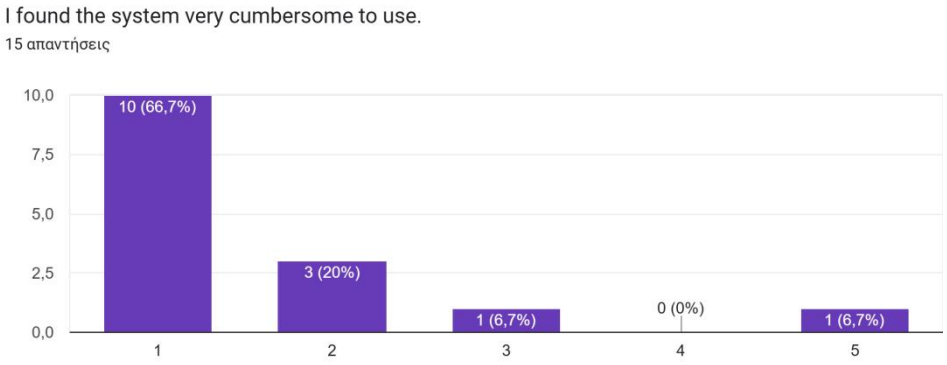


Figure 6-12 Διάγραμμα ροής απαντήσεων 8ης ερώτησης

Για το 9ο ερώτημα, όπως απεικονίζεται στο Σχήμα 6.13, όλοι οι χρήστες, με εξαίρεση έναν, απάντησαν "Συμφωνώ απόλυτα", δηλαδή έχουν αυτοπεποίθηση όσον αφορά τη χρήση του AutoKNN.

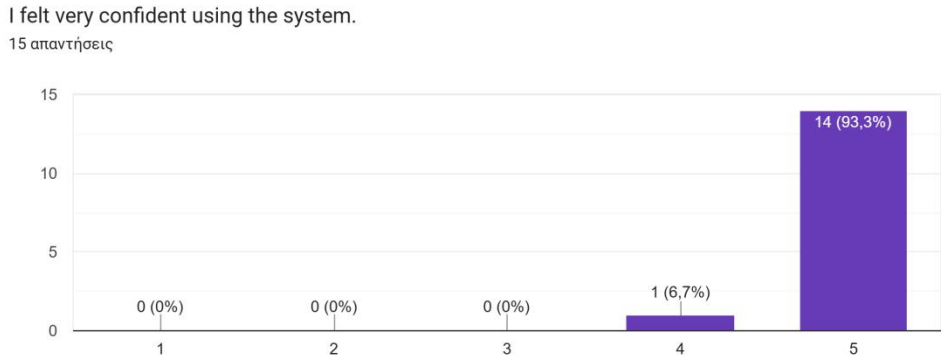


Figure 6-13 Διάγραμμα ροής απαντήσεων 9ης ερώτησης

Για το 10ο ερώτημα, όπως απεικονίζεται στο Σχήμα 6.14, οι πλειοψηφία των χρηστών απάντησε "Διαφωνώ", δηλαδή δε θεωρούν πως για τη πιο αποτελεσματική κι ορθή χρήση του AutoKNN προαπαιτούνται κατάλληλες γνώσεις.

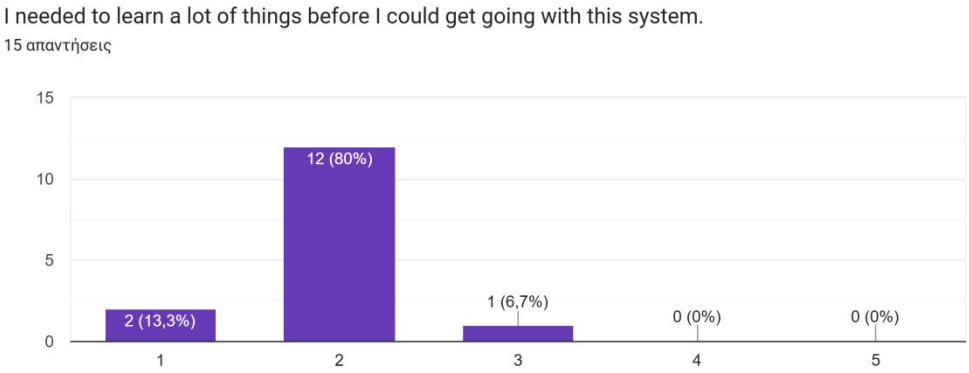


Figure 6-14 Διάγραμμα ροής απαντήσεων 10ης ερώτησης

6.2.3 Τελική βαθμολογία του SUS

Η βαθμολογία SUS προσδιορίζεται μέσω μιας συστηματικής μεθόδου που μετατρέπει τις ακατέργαστες απαντήσεις σε μια τυποποιημένη βαθμολογία ευχρηστίας του 100. Αρχικά, για τις ερωτήσεις που είναι θετικά διατυπωμένες (1, 3, 5, 7 και 9), αφαιρείται 1 από τη βαθμολογία που επιλέγει ο χρήστης. Για τις ερωτήσεις με αρνητική διατύπωση (2, 4, 6, 8 και 10), η επιλεγμένη βαθμολογία του χρήστη αφαιρείται από το 5. Αυτή η τυποποίηση εγγυάται ότι οι υψηλότερες βαθμολογίες αντικατοπτρίζουν σταθερά τη βελτιωμένη χρηστικότητα. Αφού τροποποιηθούν οι επιμέρους απαντήσεις, οι τιμές αθροίζονται, οδηγώντας σε μια συνολική βαθμολογία που κυμαίνεται από 0 έως 40. Το άθροισμα αυτό πολλαπλασιάζεται στη συνέχεια επί 2,5 για να προσαρμοστεί η τελική βαθμολογία SUS σε ένα εύρος από 0 έως 100, αν και αυτό δεν αντιπροσωπεύει ποσοστό. Μια βαθμολογία SUS μεγαλύτερη από 68 θεωρείται συνήθως ως άνω του μέσου όρου, ενώ χαμηλότερες βαθμολογίες υποδηλώνουν πιθανά προβλήματα ευχρηστίας [39], [40]. Αυτή η ομοιόμορφη μέθοδος καθιστά το SUS ένα αξιόπιστο και συχνά χρησιμοποιούμενο μέτρο για την αξιολόγηση της ευχρηστίας.

Ο Πίνακας 6.2 παρουσιάζει τη τελική βαθμολογία SUS της διαδικτυακής εφαρμογής AutoKNN.

Χρήστης	**Βαθμολογία**
1	90
2	90
3	92,5
4	95
5	95
6	95
7	95
8	92,5
9	90
10	87,5
11	80
12	80
13	87,5
14	87,5
15	80
Τελική Βαθμολογία	**89,16**

Table 6-2 Τελική βαθμολογία SUS του AutoKNN

Κεφάλαιο 7ο: Συμπεράσματα και Μελλοντικές επεκτάσεις

7.1 Συμπεράσματα & Μελλοντικές επεκτάσεις

Ο k-NN είναι ένας δημοφιλής κατηγοριοποιητής με εύκολη δομή και λειτουργικότητα. Παρόλα αυτά, η χρήση του πολλές φορές για τον απλό χρήστη δεν είναι εύκολη δουλειά, καθώς είναι αναγκαία η αξιοποίηση σχετικών γνώσεων και λογισμικών που πολλές φορές κοστολογούνται, με αποτέλεσμα περιορισμένης πρόσβασης και χρήσης.

Γι' αυτό το λόγο, στο πλαίσιο αυτής της διπλωματικής εργασίας, αναπτύχθηκε η διαδικτυακής εφαρμογή "AutoKNN". Το AutoKNN αποτελεί ένα δωρεάν κι ανοιχτού κώδικα λογισμικό με πρωταρχικό στόχο την απεριόριστη κι εύκολη χρήση του κατηγοριοποιητή k-NN. Συγκεκριμένα, δίνεται η δυνατότητα στον χρήστη να δημιουργήσει καινούργια μοντέλα βάσει προσωπικών προτιμήσεων του χρήστη, να πραγματοποιηθεί μια εκτίμηση της απόδοσης του νέου μοντέλου, και να αποθηκεύσει το νέο μοντέλο για μελλοντική χρήση, όπως για πρόβλεψη μη κατηγοριοποιημένων στιγμιοτύπων. Για τη πρόσβαση του χρήστη στη διαδικτυακή εφαρμογή παρέχεται μέσω ενός φιλικού GUI και μέσω ενός ελεύθερου Web API. Το AutoKNN φιλοξενείται από έναν server του Τμήματος Μηχανικών Πληροφορικής και Ηλεκτρονικών Συστημάτων, του Διεθνούς Πανεπιστημίου Ελλάδος. Επιπρόσθετα, μπορεί να είναι χρήσιμο για μια ομάδα οποιονδήποτε ενδιαφερόμενων χρηστών, πράγμα που συμπεραίνεται κι από τη διαδικτυακή αξιολόγηση της εφαρμογής από τους χρήστες.

Ανεξαρτήτως από το ευρύ φάσμα δυνατοτήτων που προσφέρει το AutoKNN για την ευκολότερη εκτέλεση του κατηγοριοποιητή k-NN από τους χρήστες, είναι εφικτή η περαιτέρω βελτίωση κι ανάπτυξης της συγκεκριμένης εφαρμογής σε μερικούς τομείς. Αυτοί οι τομείς είναι οι εξής:

1. **Ανάπτυξη εφαρμογής και για Android συσκευές:** Μία μελλοντική ανάπτυξη του AutoKNN, όσον αφορά το γενικευμένο σχεδιασμό του GUI, είναι η ανάπτυξη σχεδιασμού για Android συσκευές, όπως είναι τα smartphones και τα tablets. Είναι μια βελτίωση, κατά την οποία θα επωφελούσε σημαντικά τους χρήστες, καθώς θα μπορούν να αξιοποιούν το AutoKNN από οποιαδήποτε επιθυμητή συσκευή.
2. **Προσθήκη παραλλαγών του k-NN:** Επίσης μια εξίσου καλή μελλοντική ανάπτυξη του AutoKNN, είναι η προσθήκη κι άλλων επιπλέον κατηγοριοποιητών, όπου αποτελούν παραλλαγές του k-NN, όπως ο k-d Tree, Condensed Nearest Neighbor κι ο Radiud-based Nearest Neighbors. Μια τέτοια προσθήκη, συνεισφέρει στη παροχή μεγαλύτερης ευελιξίας και στη καλύτερη απόδοση σε πιο εξειδικευμένες περιπτώσεις κατηγοριοποιήσης.

ΒΙΒΛΙΟΓΡΑΦΙΑ

[1] Koturwar, P., Girase, S., & Mukhopadhyay, D. (2015). A survey of classification techniques in the area of big data. arXiv preprint arXiv:1503.07477.

[2] Osisanwo, F. Y., Akinsola, J. E. T., Awodele, O., Hinmikaiye, J. O., Olakanmi, O., & Akinjobi, J. (2017). Supervised machine learning algorithms: classification and comparison. International Journal of Computer Trends and Technology (IJCTT), 48(3), 128-138.

[3] Alnuaimi, A. F., & Albaldawi, T. H. (2024). An overview of machine learning classification techniques. In BIO Web of Conferences (Vol. 97, p. 00133). EDP Sciences.

[4] U. Narayanan, A. Unnikrishnan, V. Paul and S. Joseph, "A survey on various supervised classification algorithms," 2017 International Conference on Energy, Communication, Data Analytics and Soft Computing (ICECDS), Chennai, India, 2017, pp. 2118-2124, doi: 10.1109/ICECDS.2017.8389824. keywords: {Decision trees;Diseases;Neural networks;Classification algorithms;Support vector machines;Bayes methods;Feature extraction;Decision tree;K-Nearest Neighbour;Neural Network;Naive Bayes;Support vector machine}.

[5] Kanksha, Singh, H., Laxmi, V. (2021). Supervısed Learnıng Algorıthm: A Survey. In: Luhach, A.K., Jat, D.S., Bin Ghazali, K.H., Gao, XZ., Lingras, P. (eds) Advanced Informatics for Computing Research. ICAICR 2020. Communications in Computer and Information Science, vol 1393. Springer, Singapore. https://doi.org/10.1007/978-981-16-3660-8_7.

[6] Nasteski, V. (2017). An overview of the supervised machine learning methods. Horizons. b, 4(51-62), 56.

[7] Gupta, A., Joshi, R., Kanvinde, N., Gerela, P., Laban, R.M. (2023). Metric Effects Based on Fluctuations in Values of k in Nearest Neighbor Regressor. In: Jacob, I.J., Kolandapalayam Shanmugam, S., Izonin, I. (eds) Data Intelligence and Cognitive Informatics. Algorithms for Intelligent Systems. Springer, Singapore. https://doi.org/10.1007/978-981-19-6004-8_12.

[8] Abu Alfeilat, H. A., Hassanat, A. B., Lasassmeh, O., Tarawneh, A. S., Alhasanat, M. B., Eyal Salman, H. S., & Prasath, V. S. (2019). Effects of distance measure choice on k-nearest neighbor classifier performance: a review. Big data, 7(4), 221-248.

[9] Song, K. (2019). Adaptive nearest neighbor: A general framework for distance metric learning. arXiv preprint arXiv:1911.10674.

[10] Lodwich, A., Shafait, F., & Breuel, T. (2016). Efficient estimation of k for the nearest neighbors class of methods. *arXiv preprint arXiv:1606.02617*.

[11] Xin He, Kaiyong Zhao, Xiaowen Chu, AutoML: A survey of the state-of-the-art, Knowledge-Based Systems, Volume 212, 2021, 106622, ISSN 0950-7051, https://doi.org/10.1016/j.knosys.2020.106622.

[12] Shen, Z., Zhang, Y., Wei, L., Zhao, H., & Yao, Q. (2018). Automated Machine Learning: From Principles to Practices. arXiv preprint arXiv:1810.13306.

[13] Yi-Wei Chen, Qingquan Song, and Xia Hu. 2021. Techniques for Automated Machine Learning. SIGKDD Explor. Newsl. 22, 2 (December 2020), 35–50. https://doi.org/10.1145/3447556.3447567.

[14] T. Nagarajah and G. Poravi, "A Review on Automated Machine Learning (AutoML) Systems," 2019 IEEE 5th International Conference for Convergence in Technology (I2CT), Bombay, India, 2019, pp. 1-6, doi: 10.1109/I2CT45611.2019.9033810. keywords: {Machine learning;Machine learning algorithms;Optimization;Tuning;Task analysis;Data models;Bayes methods;autoML;hyperparameter;automation;AI}.

[15] Kramer, O. (2013). K-Nearest Neighbors. In: Dimensionality Reduction with Unsupervised Nearest Neighbors. Intelligent Systems Reference Library, vol 51. Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-642-38652-7_2.

[16] Yihua Liao, V.Rao Vemuri, Use of K-Nearest Neighbor classifier for intrusion detection11An earlier version of this paper is to appear in the Proceedings of the 11th USENIX Security Symposium, San Francisco, CA, August 2002, Computers & Security, Volume 21, Issue 5, 2002, Pages 439-448, ISSN 0167-4048, https://doi.org/10.1016/S0167-4048(02)00514-X.

[17] Pádraig Cunningham and Sarah Jane Delany. 2021. K-Nearest Neighbour Classifiers - A Tutorial. ACM Comput. Surv. 54, 6, Article 128 (July 2022), 25 pages. https://doi.org/10.1145/3459665.

[18] Chomboon, K., Chujai, P., Teerarassamee, P., Kerdprasop, K., & Kerdprasop, N. (2015, March). An empirical study of distance metrics for k-nearest neighbor algorithm. In Proceedings of the 3rd international conference on industrial application engineering (Vol. 2, p. 4).

[19] É.O. Rodrigues, Combining Minkowski and Chebyshev: New distance proposal and survey of distance metrics using k-nearest neighbours classifier, Pattern Recognition Letters, Volume 110, 2018, Pages 66-71, ISSN 0167-8655, https://doi.org/10.1016/j.patrec.2018.03.021.

[20] Jianping Gou, Hongxing Ma, Weihua Ou, Shaoning Zeng, Yunbo Rao, Hebiao Yang, A generalized mean distance-based k-nearest neighbor classifier, Expert Systems with Applications, Volume 115, 2019, Pages 356-372, ISSN 0957-4174, https://doi.org/10.1016/j.eswa.2018.08.021.

[21] Nasser, A.B. et al. (2023). A Robust Tuned K-Nearest Neighbours Classifier for Software Defect Prediction. In: Al-Sharafi, M.A., Al-Emran, M., Al-Kabi, M.N., Shaalan, K. (eds) Proceedings of the 2nd International Conference on Emerging Technologies and Intelligent Systems . ICETIS 2022. Lecture Notes in Networks and Systems, vol 573. Springer, Cham. https://doi.org/10.1007/978-3-031-20429-6_18.

[22] Inyang, U. G., Ijebu, F. F., Osang, F. B., Afolorunso, A. A., Udoh, S. S., & Eyoh, I. J. (2023). A Dataset-Driven Parameter Tuning Approach for Enhanced K-Nearest Neighbour Algorithm Performance. International Journal on Advanced Science, Engineering & Information Technology, 13(1).

[23] Y. Sari, M. Maulida, E. Gunawan and J. Wahyudi, "Artificial Intelligence Approach For BAZNAS Website Using K-Nearest Neighbor (KNN)," 2021 Sixth International Conference on Informatics and Computing (ICIC), Jakarta, Indonesia, 2021, pp. 1-4, doi: 10.1109/ICIC54025.2021.9632954. keywords: {Support vector machines;Soft sensors;Training data;Nearest neighbor methods;Decision trees;Informatics;Artificial intelligence;Zakat;Islam;K-Nearest Neighbor;BAZNAS}.

[24] Li, L., Song, D., Ma, R., Qiu, X., & Huang, X. (2021). KNN-BERT: fine-tuning pre-trained models with KNN classifier. arXiv preprint arXiv:2110.02523.

[25] A. Briliani, B. Irawan and C. Setianingsih, "Hate Speech Detection in Indonesian Language on Instagram Comment Section Using K-Nearest Neighbor Classification Method," 2019 IEEE International Conference on Internet of Things and Intelligence System (IoTaIS), Bali, Indonesia, 2019, pp. 98-104, doi: 10.1109/IoTaIS47347.2019.8980398. keywords: {Instagram;K-Nearest Neighbor;hate speech}.

[26] Siame, A., & Kunda, D. (2017). Evolution of PHP applications: A systematic literature review. International Journal of Recent Contributions from Engineering, Science & IT (iJES), 5(1), 28-39.

[27] Natalya Prokofyeva, Victoria Boltunova, Analysis and Practical Application of PHP Frameworks in Development of Web Information Systems, Procedia Computer Science, Volume 104, 2017, Pages 51-56, ISSN 1877-0509, https://doi.org/10.1016/j.procs.2017.01.059.

[28] K. J. Millman and M. Aivazis, "Python for Scientists and Engineers," in Computing in Science & Engineering, vol. 13, no. 2, pp. 9-12, March-April 2011, doi: 10.1109/MCSE.2011.36. keywords: {Special issues and sections;Computer languages;Programming;Scientific computing;Numerical models;Programming languages;Python;Scientific computing;interactive research;Python libraries;Python tools}.

[29] Python, W. (2021). Python. Python releases for windows, 24.

[30] Christudas, B. (2019). Practical microservices architectural patterns: event-based java microservices with spring boot and spring cloud. Apress.

[31] K. Rinartha and W. Suryasa, "Comparative study for better result on query suggestion of article searching with MySQL pattern matching and Jaccard similarity," 2017 5th International Conference on Cyber and IT Service Management (CITSM), Denpasar, Indonesia, 2017, pp. 1-4, doi: 10.1109/CITSM.2017.8089237. keywords: {Pattern matching;Radar;Indexes;Radar signal processing;Search engines;Article searching;query suggestions;MySQL Pattern Matching;Jaccard Similarity}.

[32] Hao, J., & Ho, T. K. (2019). Machine Learning Made Easy: A Review of Scikit-learn Package in Python Programming Language. Journal of Educational and Behavioral Statistics, 44(3), 348-361. https://doi.org/10.3102/1076998619832248 (Original work published 2019).

[33] Khliupko, V. (2017). Composer. In: Magento 2 DIY . Apress, Berkeley, CA. https://doi.org/10.1007/978-1-4842-2460-1_6.

[34] Eltahawey, A. O. (2016). Hyper Text Markup Language HTML: A Tutorial.

[35] Jensen, S.H., Møller, A., Thiemann, P. (2009). Type Analysis for JavaScript. In: Palsberg, J., Su, Z. (eds) Static Analysis. SAS 2009. Lecture Notes in Computer Science, vol 5673. Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-642-03237-0_17.

[36] D. Mazinanian and N. Tsantalis, "An Empirical Study on the Use of CSS Preprocessors," 2016 IEEE 23rd International Conference on Software Analysis, Evolution, and Reengineering (SANER), Osaka, Japan, 2016, pp. 168-178, doi: 10.1109/SANER.2016.18. keywords: {Cascading style sheets;Data preprocessing;Syntactics;Programming;Libraries;HTML;Color}.

[37] López-Gorozabel, O., Cedeño-Palma, E., Pinargote-Ortega, J., Zambrano-Romero, W., Pazmiño-Campuzano, M. (2021). Bootstrap as a Tool for Web Development and Graphic Optimization on Mobile Devices. In: Botto-Tobar, M., Cruz, H., Díaz Cadena, A. (eds) Artificial Intelligence,

Computer and Software Engineering Advances. CIT 2020. Advances in Intelligent Systems and Computing, vol 1326. Springer, Cham. https://doi.org/10.1007/978-3-030-68080-0_22.

[38] Joshi, B. (2012). Overview of jQuery. In: HTML5 Programming for ASP.NET Developers. Apress, Berkeley, CA. https://doi.org/10.1007/978-1-4302-4720-3_2.

[39] Grier, R. A., Bangor, A., Kortum, P., & Peres, S. C. (2013). The System Usability Scale: Beyond Standard Usability Testing. Proceedings of the Human Factors and Ergonomics Society Annual Meeting, 57(1), 187-191. https://doi.org/10.1177/1541931213571042 (Original work published 2013).

[40] Lewis, J. R. (2018). The System Usability Scale: Past, Present, and Future. International Journal of Human–Computer Interaction, 34(7), 577–590. https://doi.org/10.1080/10447318.2018.1455307.