ΔΙΕΘΝΕΣ ΠΑΝΕΠΙΣΤΗΜΙΟ ΤΗΣ ΕΛΛΑΔΟΣ

ΣΧΟΛΗ ΜΗΧΑΝΙΚΩΝ

ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΗΛΕΚΤΡΟΝΙΚΩΝ ΣΥΣΤΗΜΑΤΩΝ

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

AutoDBSCAN
Διαδικτυακή εφαρμογή για την εκτέλεση συσταδοποίησης με χρήση του αλγορίθμου DBSCAN

Φοιτητής:
Κεΐκογλου Ιωάννης
Student ID: 154464

Επιβλέπων:
Ουγιάρογλου Στέφανος

23 May 2023

Τίτλος Π.Ε. AutoDBSCAN Διαδικτυακή εφαρμογή για την εκτέλεση συσταδοποίησης με χρήση του αλγορίθμου DBSCAN
Κωδικός Π.Ε. 23230
Ονοματεπώνυμο φοιτητή Κεΐκογλου Ιωάννης
Ονοματεπώνυμο εισηγητή Ουγιάρογλου Στέφανος
Ημερομηνία ανάληψης Π.Ε. 25-03-2023
Ημερομηνία περάτωσης Π.Ε. 01-10-2023

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.

Αφιέρωση

Αρχικά, θα ήθελα να απευθύνω τις θερμότερες ευχαριστίες μου στον κ. Στέφανο Ουγιάρογλου, για την πολύτιμη εμπιστοσύνη που μου προσέδωσε όσον αφορά την εκπόνηση της παρούσας πτυχιακής. Τον ευγνωμονώ για τον χρόνο που διέθεσε, την άμεση καθοδήγηση και την αδιάλειπτη προθυμία του να βοηθήσει κατά τη διάρκεια της σύνταξης της εργασίας. Η συνεισφορά του ήταν ανεκτίμητη. Στη συνέχεια, είναι καθήκον μου να εκφράσω τη βαθιά μου ευγνωμοσύνη προς την οικογένειά μου, η οποία υπήρξε στήριξη και πηγή έμπνευσης καθ' όλη τη διάρκεια των σπουδαστικών μου χρόνων.

Πρόλογος

Στον ψηφιακό κόσμο που ζούμε, τα δεδομένα είναι πολύ περισσότερα από απλές αλφαριθμητικές σειρές: αποτελούν την καρδιά των σύγχρονων τεχνολογιών και τον πυλώνα της πληροφορίας. Ωστόσο, τα ατέλειωτα ψηφιακά ρεύματα που περιβάλλουν κάθε άκρη του κόσμου μας δεν προσφέρουν αξία αν δεν μπορούμε να τα αναλύσουμε και να τα κατανοήσουμε. Η ανάγκη για ισχυρά εργαλεία ανάλυσης δεδομένων έχει γίνει πιο πιεστική από ποτέ. Στο πλαίσιο αυτό, ο αλγόριθμος συσταδοποίησης DBSCAN προσφέρει μια ξεχωριστή προοπτική. Παρά τον φαινομενικά τεχνικό χαρακτήρα του, αυτός ο αλγόριθμος καταδεικνύει πώς οι μηχανές μπορούν να ”βλέπουν” και να ”κατανοούν” τον κόσμο μας με τρόπους που θα φαίνονταν αδύνατοι πριν από μερικές δεκαετίες. Σε αυτή την εργασία, θα εξερευνήσουμε τη δύναμη και την πολυπλοκότητα του DBSCAN, αποκαλύπτοντας το πώς μπορεί να μετατρέψει τον τρόπο που κατανοούμε τα δεδομένα στην εποχή της πληροφορίας.

Περίληψη

Με τον όρο συσταδοποίηση (clustering) αναφερόμαστε στη διαδικασία ταξινόμησης των δεδομένων σε ομάδες με βάση τα κοινά χαρακτηριστικά τους. Ένας από τους πιο ξεχωριστούς αλγόριθμους για τον σκοπό αυτό είναι ο DBSCAN, ο οποίος λειτουργεί με βάση την πυκνότητα των δεδομένων. Σε αντίθεση με τις παραδοσιακές μεθόδους συσταδοποίησης, ο DBSCAN μπορεί να αναγνωρίσει συστάδες διαφορετικών σχημάτων και μεγεθών. Εντούτοις, ένα σημαντικό ζήτημα που αντιμετωπίζουν οι ερευνητές και οι αναλυτές είναι η εύρεση των βέλτιστων τιμών για τις παραμέτρους Eps και Minpts του αλγορίθμου. Η επιλογή ακατάλληλων τιμών μπορεί να οδηγήσει σε ανεπιθύμητα αποτελέσματα συσταδοποίησης. Μετά από διεξοδική έρευνα, παρατηρήσαμε την έλλειψη σύγχρονων εργαλείων που να παρέχουν ολοκληρωμένες λύσεις για την αυτοματοποίηση της διαδικασίας επιλογής παραμέτρων στον DBSCAN. Για τον λόγο αυτό, αναπτύξαμε την εφαρμογή AutoDBSCAN. Η AutoDBSCAN είναι μια διαδικτυακή εφαρμογή που επιτρέπει στους χρήστες να φορτώσουν σύνολα δεδομένων και να λάβουν βελτιστοποιημένες προτάσεις για τις τιμές των παραμέτρων του αλγορίθμου. Τα αποτελέσματα της συσταδοποίησης παρουσιάζονται στον χρήστη με μια φιλική διεπαφή, ενώ ταυτόχρονα παρέχεται η δυνατότητα λήψης των τελικών συστάδων και των σχετικών διαγραμμάτων. Με την εφαρμογή AutoDBSCAN, οι χρήστες μπορούν να αποφύγουν τον χρονοβόρο πειραματισμό με διάφορες τιμές παραμέτρων και να καταλήξουν σε αποτελεσματικές συσταδοποιήσεις με βάση τα δεδομένα τους.

Abstract

The term clustering refers to the process of sorting data into groups based on their common characteristics. One of the most distinctive algorithms for this purpose is DBSCAN, which operates based on the density of the data. Unlike traditional clustering methods, DBSCAN can identify clusters of different shapes and sizes. However, a significant challenge faced by researchers and analysts is finding the optimal values for the parameters Eps and Minpts of the algorithm. Choosing inappropriate values can lead to undesirable clustering results. After thorough research, we observed a lack of modern tools that provide comprehensive solutions for automating the parameter selection process in DBSCAN. For this reason, we developed the application AutoDBSCAN. AutoDBSCAN is a web application that allows users to upload datasets and receive optimized parameter value suggestions for the algorithm. The clustering results are presented to the user through a user-friendly interface, while also offering the option to obtain the final clusters and relevant diagrams. With the AutoDBSCAN application, users can avoid time-consuming experimentation with various parameter values and achieve effective clustering based on their data.

Περιεχόμενα

Κατάλογος Σχημάτων

Κατάλογος Πινάκων

Κεφάλαιο 1

Εισαγωγή

1.1 Συσταδοποίηση δεδομένων

Η υσταδοποίηση αναφέρεται στην τεχνική της ομαδοποίησης των δεδομένων όπου τα στοιχεία του συνόλου διαχωρίζονται σε ομάδες ή συστάδες βάσει των παρόμοιων χαρακτηριστικών που κοινοποιούν. Ο κεντρικός στόχος είναι να εντοπιστούν εσωτερικές ομοιότητες μεταξύ των στοιχείων που ανήκουν στην ίδια συστάδα, ενώ ταυτόχρονα να είναι διακριτά από τα στοιχεία που ανήκουν σε διαφορετικές συστάδες. Ουσιαστικά, η συσταδοποίηση επιδιώκει να αναδείξει μοτίβα, δομές και σχέσεις μεταξύ των δεδομένων που ενδέχεται να μην είναι ευδιάκριτα σε πρώτη ματιά. Κάθε συστάδα αντιστοιχεί σε μια ομάδα δεδομένων που είναι παρόμοια μεταξύ τους, είτε από την άποψη των χαρακτηριστικών είτε από τη γεωμετρική τους διάταξη. Το αποτέλεσμα της συσταδοποίησης είναι η ομαδοποίηση των δεδομένων σε διαφορετικές και αυτόνομες ομάδες, κάθε μία από τις οποίες αντιπροσωπεύει ένα συγκεκριμένο πρότυπο ή συμπεριφορά. Η συσταδοποίηση είναι ιδιαίτερα χρήσιμη όταν δεν έχουμε προκαθορισμένες κατηγορίες ή ετικέτες για τα δεδομένα μας και θέλουμε να εξάγουμε φυσικές δομές και σχέσεις μεταξύ τους.

Οι πρώτες προσπάθειες για συσταδοποίηση ανήκουν στα μέσα του 20ο αιώνα. Ο μαθηματικός και στατιστικός George W. Brown ανέπτυξε τον αλγόριθμο “k-means” το 1957, που είναι ένας από τους πρώτους αλγορίθμους συσταδοποίησης[1]. Παρόλα αυτά, η συσταδοποίηση με βάση την πυκνότητα και την ανίχνευση συστάδων με μη κανονικό σχήμα και διάφορα μεγέθη αποκτά ακόμα μεγαλύτερο ενδιαφέρον στις αρχές της δεκαετίας του 1990.

Είναι σημαντικό να αντιληφθούμε τη διαφοροποίηση μεταξύ δύο κρίσιμων εννοιών: της συσταδοποίησης και της κατηγοριοποίησης (Classification) στο πεδίο της μηχανικής μάθησης. Στην κατηγοριοποίηση, που ανήκει στην κατηγορία της εποπτευόμενης μάθησης, χρησιμοποιούμε σύνολα δεδομένων με ετικέτες για να εκπαιδεύσουμε τους αλγορίθμους να ομαδοποιούν τα δεδομένα ή να προβλέπουν αποτελέσματα με ακρίβεια.[2]. Το μοντέλο, εδώ, χρησιμοποιεί τις ετικέτες για να αντιληφθεί τη σημασία των διαφορετικών χαρακτηριστικών και να προσαρμοστεί στο γνωστό αποτέλεσμα με το πέρασμα του χρόνου

Αντίθετα, η συσταδοποίηση ανήκει στους αλγορίθμους της μη εποπτευόμενης μάθησης. Σε αυ-

τήν την προσέγγιση, χρησιμοποιούμε αλγορίθμους για να ομαδοποιήσουμε τα δεδομένα και να ανιχνεύσουμε κρυμμένα πρότυπα ή σχέσεις ανάμεσα τους χωρίς την ανάγκη για ανθρώπινη επίβλεψη. Αυτοί οι αλγόριθμοι αποκαλύπτουν μοτίβα που μπορεί να μην είναι εμφανή σε μια πρώτη ματιά, καθιστώντας την διαδικασία ανίχνευσης δομών και τάσεων πιο αυτόνομη.[3]

Η ουσιαστική διάκριση μεταξύ αυτών των προσεγγίσεων έγκειται στο γεγονός ότι η κατηγοριοποίηση αναλαμβάνει να εκπαιδεύσει τους αλγορίθμους με βάση ετικέτες, ενώ η συσταδοποίηση αποσκοπεί στην εξαγωγή μοτίβων από τα δεδομένα χωρίς την προκαθορισμένη κατηγοριοποίηση. Τα μοντέλα εποπτευόμενης μάθησης είναι πιο ακριβή αλλά απαιτούν ανθρώπινη επίβλεψη κατά τη διαδικασία επεξεργασίας των δεδομένων για τη διασφάλιση της κατάλληλης επισημασίας.

1.2 Κατηγορίες αλγορίθμων συσταδοποίησης

Οι αλγόριθμοι συσταδοποίησης είναι τεχνικές που χρησιμοποιούνται για τον διαχωρισμό ενός συνόλου δεδομένων σε ομάδες (συστάδες) με παρόμοια χαρακτηριστικά. Ανάλογα με τον τρόπο με τον οποίο εκτελούν τον διαχωρισμό και τις απαιτήσεις των δεδομένων, οι αλγόριθμοι συσταδοποίησης μπορούν να κατηγοριοποιηθούν σε διάφορες κατηγορίες. Οι κύριες κατηγορίες αλγορίθμων συσταδοποίησης περιλαμβάνουν:

Ιεραρχική Συσταδοποίηση (Hierarchical Clustering):
Η ιεραρχική συσταδοποίηση είναι μια διαδικασία συσταδοποίησης που αναδεικνύει την δομή των δεδομένων με τη δημιουργία ενός ιεραρχικού δέντρου από συστάδες. Σε αυτό το δέντρο, κάθε φύλλο αντιπροσωπεύει ένα μοναδικό δείγμα, ενώ οι κόμβοι πάνω από τα φύλλα αντιπροσωπεύουν τις συστάδες στις οποίες ανήκουν τα δείγματα. Υπάρχουν δύο βασικές προσεγγίσεις για την ιεραρχική συσταδοποίηση: η αγλοειδής ιεραρχία και η κάτω προς τα πάνω ιεραρχία.

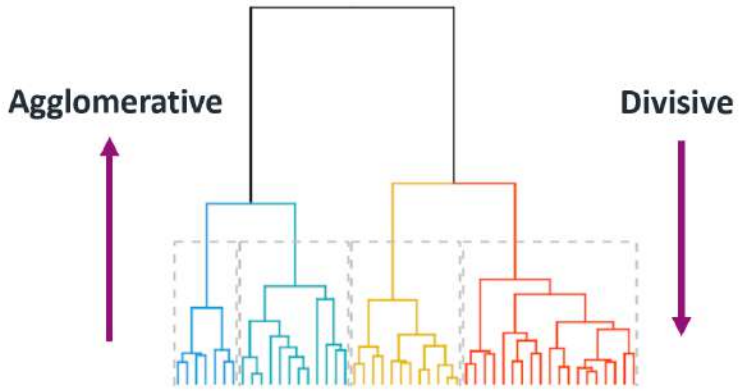
- Αγλοειδής Ιεραρχία (Agglomerative Hierarchical Clustering): Στην αγλοειδή ιεραρχία, ξεκινάμε με κάθε δείγμα σε ξεχωριστή συστάδα και στη συνέχεια συνενώνουμε τις πιο παρόμοιες συστάδες μέχρι να φτάσουμε σε μια μεγάλη συστάδα που περιέχει όλα τα δείγματα. Η διαδικασία επαναλαμβάνεται, συνενώνοντας συστάδες σε όλο και μεγαλύτερα επίπεδα, μέχρι να δημιουργηθεί το ιεραρχικό δέντρο.

- Κάτω προς τα Πάνω Ιεραρχία (Divisive Hierarchical Clustering): Στην κάτω προς τα πάνω ιεραρχία, ξεκινάμε με μια μεγάλη συστάδα που περιέχει όλα τα δείγματα και διαιρούμε αυτήν την συστάδα σε μικρότερες και πιο παρόμοιες συστάδες. Η διαδικασία συνεχίζεται αποσπώντας υποσυστάδες σε όλο και μικρότερα επίπεδα, μέχρι να φτάσουμε στα φύλλα του ιεραρχικού δέντρου.

Οι αγλοειδείς και κάτω προς τα πάνω προσεγγίσεις μπορούν να υλοποιηθούν με διάφορους αλγορίθμους. Ο κοινός παρονομαστής είναι η δημιουργία του ιεραρχικού δέντρου, που είναι η κύρια έξοδος αυτών των αλγορίθμων.

Πλεονεκτήματα της ιεραρχικής συσταδοποίησης περιλαμβάνουν τη δυνατότητα οπτικοποίησης της δομής των δεδομένων μέσω του ιεραρχικού δέντρου και την δυνατότητα εύρεσης του

βέλτιστου αριθμού συστάδων. Ωστόσο, η ιεραρχική συσταδοποίηση μπορεί να είναι χρονοβόρα για μεγάλα σύνολα δεδομένων.



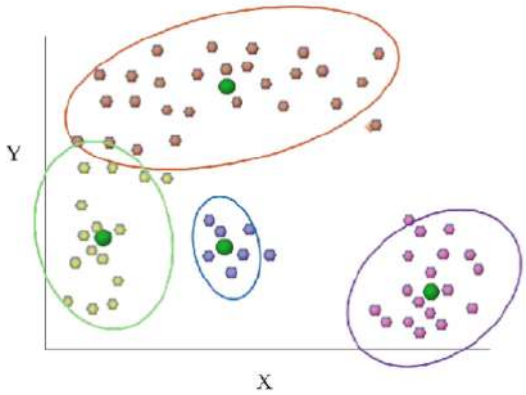
Σχήμα 1.1: Agglomerative-Divisive Hierarchical Clustering

Συσταδοποίηση με Κέντρο (Centroid-based Clustering):
Η centroid-based συσταδοποίηση είναι μια δημοφιλής προσέγγιση στον τομέα της συσταδοποίησης, όπου οι συστάδες ορίζονται από τα κέντρα των συστάδων. Στην επίλυση του προβλήματος της centroid-based συσταδοποίησης, το κύριο ερώτημα είναι πώς θα βρούμε τα κέντρα αυτά, τοποθετώντας τα στη μέση των δεδομένων των συστάδων.
Οι δύο πιο διαδεδομένοι αλγόριθμοι centroid-based συσταδοποίησης είναι οι k-means και k-medoids.

- K-means: Ο αλγόριθμος k-means χωρίζει τα δεδομένα σε k συστάδες, όπου το k είναι ένα προκαθορισμένο πλήθος συστάδων που πρέπει να δημιουργηθούν. Ο αλγόριθμος επαναλαμβάνει δύο βήματα: αρχικά, εκτιμά το κέντρο της κάθε συστάδας ως το μέσο όρο όλων των δειγμάτων στην συστάδα, και στη συνέχεια, ανατιθέται κάθε δείγμα στην συστάδα με το πλησιέστερο κέντρο. Η διαδικασία επαναλαμβάνεται μέχρι να σταθεροποιηθούν τα κέντρα[4].

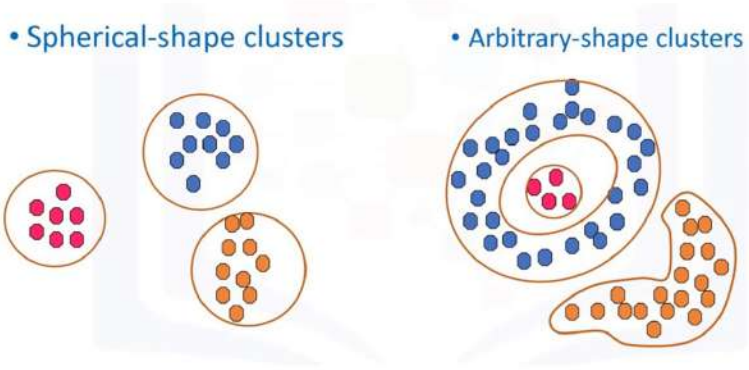
- K-medoids: Ο αλγόριθμος k-medoids αποτελεί μια παραλλαγή του k-means, όπου αντί για το μέσο όρο των δειγμάτων, χρησιμοποιεί το κεντρικότερο δείγμα της συστάδας ως το κέντρο της. Αυτό κάνει τον αλγόριθμο πιο ανθεκτικό σε ακραίες τιμές και θόρυβο[3].

Και οι δύο αυτοί αλγόριθμοι απαιτούν αρχικές εκτιμήσεις των κεντρικών σημείων των συστάδων και επαναλαμβανόμενα βήματα ενημέρωσης αυτών των εκτιμήσεων για την εύρεση των τελικών κέντρων. Οι αλγόριθμοι αυτοί είναι αποτελεσματικοί και ευέλικτοι για ποικίλες εφαρμογές, αλλά μπορεί να υπάρχουν περιπτώσεις όπου σταματούν σε τοπικά ελάχιστα.



Σχήμα 1.2: Centroid-based Clustering

Συσταδοποίηση με Πυκνότητα (Density-based Clustering):
Η Density-based συσταδοποίηση είναι μια προσέγγιση στον τομέα της συσταδοποίησης που βασίζεται στην πυκνότητα των δεδομένων[1]. Αντίθετα από την centroid-based προσέγγιση που απαιτεί προκαθορισμένο αριθμό συστάδων, οι αλγόριθμοι που βασίζονται στην πυκνότητα ανιχνεύουν συστάδες με βάση την πυκνότητα των δεδομένων και τη συνεχή συνδεσιμότητα μεταξύ των γειτονικών σημείων.
Ο πιο γνωστός αλγόριθμος στην κατηγορία αυτή είναι ο DBSCAN (Density-Based Spatial Clustering of Applications with Noise). Ο DBSCAN ομαδοποιεί τα δεδομένα σε συστάδες με βάση την πυκνότητα, εντοπίζοντας περιοχές υψηλής πυκνότητας και χωρίζοντας τις με χαμηλής πυκνότητας περιοχές (άκυρα δείγματα ή θόρυβο). Ο αλγόριθμος αναπτύχθηκε για να αντιμετωπίσει τη δυσκολία της αποκαλούμενης “κοιλότητας” (density holes) που αντιμετωπίζουν άλλοι αλγόριθμοι, όπως οι k-means. Στο επόμενο κεφάλαιο θα υπάρξει περαιτέρω αναλυση για τον συγκεκριμένο αλγόριθμο[5]

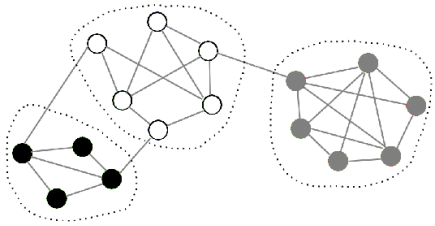


Σχήμα 1.3: Density-based Clustering

Συσταδοποίηση με Γράφους (Graph-based Clustering):
Η συσταδοποίηση με γράφους (Graph-based Clustering) είναι μια προσέγγιση στη συσταδοποίηση που βασίζεται στην ανάλυση των σχέσεων μεταξύ των δειγμάτων μέσω της αναπαράστασής τους σε μορφή γράφου[6]. Στην αναπαράσταση αυτή, τα δείγματα αντιστοιχούν σε κόμβους του γράφου, ενώ οι σχέσεις μεταξύ τους αντιπροσωπεύονται με ακμές.
Οι αλγόριθμοι συσταδοποίησης με γράφους επικεντρώνονται στην ανάλυση των συνδέσεων μεταξύ των δειγμάτων και την ανακάλυψη κοινοτήτων ή συστάδων που σχηματίζονται μέσω των γραφικών συνδέσεων. Οι βασικές αρχές της συσταδοποίησης με γράφους περιλαμβάνουν την ανίχνευση των πιο σημαντικών ακμών μεταξύ των δειγμάτων και την ανάθεση των δειγμάτων σε συστάδες με βάση τις γραφικές τους συνδέσεις.

Ένας αλγόριθμος που εμπίπτει στην κατηγορία της συσταδοποίησης με γράφους είναι ο Spectral Clustering. Ο Spectral Clustering λειτουργεί με τα εξής βήματα:[7]

- Δημιουργία γράφου: Αρχικά, δημιουργείται ένας γράφος με τα δείγματα ως κόμβους και τις σχέσεις μεταξύ τους ως ακμές.
- Υπολογισμός Μήτρας Ομοιότητας: Υπολογίζεται η μήτρα ομοιότητας μεταξύ των δειγμάτων. Αυτή η μήτρα αποτελεί το βάσιμο για τον επόμενο βήμα.
- Υπολογισμός Φασματικής Ανάλυσης: Εκτελείται φασματική ανάλυση στη μήτρα ομοιότητας προκειμένου να εξάγουμε τα κυρίαρχα στοιχεία.
- Συσταδοποίηση Κυρίων Στοιχείων: Τα κυρίαρχα στοιχεία χρησιμοποιούνται για την συσταδοποίηση των δειγμάτων, π.χ. με την μέθοδο K-Means.



Σχήμα 1.4: Graphed-based Clustering

Η προσέγγιση της συσταδοποίησης με γράφους είναι ιδιαίτερα χρήσιμη για δεδομένα που δεν μπορούν να αναπαρασταθούν καλά σε χαμηλές διαστάσεις, καθώς επιτρέπει την αξιοποίηση των διαστάσεων που αντιπροσωπεύουν τις γραφικές σχέσεις.

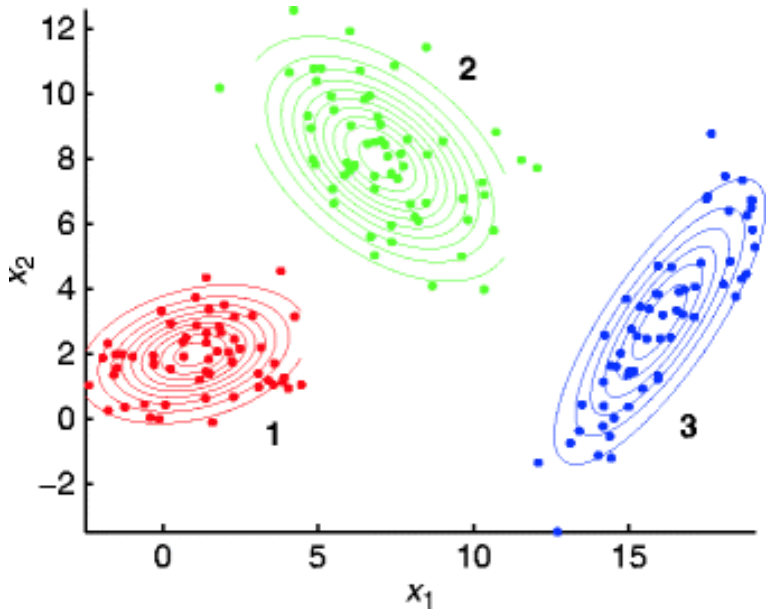
Συσταδοποίηση με Μοντέλα Πιθανοτικών Διαδικασιών (Probabilistic Model-based Clustering):

Η συσταδοποίηση με μοντέλα πιθανοτικών διαδικασιών (Probabilistic Model-based Clustering) ανήκει σε μια κατηγορία αλγορίθμων συσταδοποίησης που επιδιώκουν να εξηγήσουν την ανάμειξη των δεδομένων μέσω πιθανοτικών μοντέλων. Οι αλγόριθμοι αυτοί μοντελοποιούν τη διασπορά των δεδομένων εντός των συστάδων και επιδιώκουν να εκτιμήσουν τα κρυμμένα χαρακτηριστικά των συστάδων, όπως οι παράμετροι των κατανομών πιθανοτήτων που περιγράφουν τη δομή των δεδομένων.

Ένας γνωστός αλγόριθμος συσταδοποίησης με μοντέλα πιθανοτικών διαδικασιών είναι ο αλγόριθμος Expectation-Maximization (EM). Ο EM αλγόριθμος βασίζεται στην εκτίμηση των παραμέτρων μιας πιθανοτικής κατανομής που αντιπροσωπεύει τα δεδομένα. Αρχικά, οι συστάδες αντιστοιχίζονται σε πιθανοτικές κατανομές, και στη συνέχεια οι παράμετροι αυτών των κατανομών υπολογίζονται με βάση τα δεδομένα. Ο αλγόριθμος επαναλαμβάνει τον υπολογισμό των συσταδών και των παραμέτρων μέχρι να συγκλίνει σε μια σταθερή λύση[8].

Ένα άλλο παράδειγμα είναι ο αλγόριθμος Gaussian Mixture Model (GMM), όπου υποθέτουμε ότι κάθε συστάδα ανήκει σε μια γκαουσιανή (κανονική) κατανομή και προσπαθούμε να εκτιμήσουμε τις παραμέτρους αυτών των κατανομών[9].

Οι αλγόριθμοι συσταδοποίησης με μοντέλα πιθανοτικών διαδικασιών είναι ιδιαίτερα χρήσιμοι για την ανίχνευση συστάδων με πολύπλοκες δομές και γεωμετρίες. Ωστόσο, οι αλγόριθμοι αυτοί απαιτούν την εκτέλεση των εκτιμήσεων πιθανοτήτων και των επαναλαμβανόμενων υπολογισμών, οι οποίοι μπορεί να είναι χρονοβόροι για μεγάλα σύνολα δεδομένων.



Σχήμα 1.5: Probabilistic Model-based Clustering

1.3 Αυτόματη Μηχανική Μάθηση (Auto ML)

Η αυτόματη μηχανική μάθηση (AutoML) αναπαριστά μια σημαντική προσέγγιση στον κόσμο της τεχνητής νοημοσύνης, απλοποιώντας και επιταχύνοντας την δημιουργία μοντέλων μηχανικής μάθησης. Σε αντίθεση με την παραδοσιακή μηχανική μάθηση που απαιτούσε ανθρώπινη εμπλοκή στην επιλογή χαρακτηριστικών και τη ρύθμιση παραμέτρων, η AutoML επιτρέπει στους υπολογιστές να αναλάβουν αυτά τα καθήκοντα.

Αποτελεί μια σύγχρονη προσέγγιση που αποσκοπεί στη διευκόλυνση και επιτάχυνση της διαδικασίας δημιουργίας, εκπαίδευσης και βελτιστοποίησης μοντέλων μηχανικής μάθησης, με σκοπό τη μείωση της ανθρώπινης παρέμβασης και την επίτευξη αξιόπιστων αποτελεσμάτων [10].
Η πρώτη βασική αρχή της AutoML επικεντρώνεται στην αυτοματοποίηση της επιλογής μοντέλων, με την αυτόματη επιλογή κατάλληλων αλγορίθμων και αρχιτεκτονικών δικτύων, λαμβάνοντας υπόψη τη φύση του προβλήματος και των δεδομένων.
Η δεύτερη αρχή εστιάζει στην αυτοματοποίηση της προεπεξεργασίας των δεδομένων, περιλαμβάνοντας την αυτόματη αντιμετώπιση των απουσιαζουσών τιμών, την κωδικοποίηση κατηγορικών μεταβλητών και την εξαγωγή νέων χαρακτηριστικών.
Η τρίτη αρχή αφορά την αυτοματοποίηση της βελτιστοποίησης υπερπαραμέτρων, με την εφαρμογή αυτόματων τεχνικών βελτιστοποίησης για την εύρεση των βέλτιστων υπερπαραμέτρων.
Η τέταρτη αρχή αφορά την αυτοματοποίηση της αξιολόγησης και σύγκρισης μοντέλων, με την αυτόματη χρήση μετρικών απόδοσης και δεικτών χρηστικότητας για την επιλογή του κατάλληλου μοντέλου [11].

Βασικές αρχές της AutoML περιλαμβάνουν μια ποικιλία τεχνικών και μεθόδων που αυτοματοποιούν τη διαδικασία της μηχανικής μάθησης. Οι κύριες τεχνικές περιλαμβάνουν:

- Αναζήτηση Χώρου Υποθέσεων (Hyperparameter Search): Η AutoML αναζητά αυτόματα τις βέλτιστες τιμές υπερπαραμέτρων για ένα μοντέλο, βελτιστοποιώντας την απόδοσή του.
- Επιλογή Μοντέλων (Model Selection): Επιλέγει το κατάλληλο μοντέλο για ένα πρόβλημα, εξερευνώντας διάφορες αρχιτεκτονικές μοντέλων.
- Αυτόματη Μεταφορά Μάθησης (AutoML Transfer Learning): Εκμεταλλεύεται τη μεταφορά μάθησης για να μεταφέρει γνώση από παρόμοια προβλήματα και βελτιώνει την επίδοση του μοντέλου.
- Επιλογή Χαρακτηριστικών (Feature Selection): Αυτοματοποιεί την επιλογή των πιο σημαντικών χαρακτηριστικών των δεδομένων για τη βελτίωση της απόδοσης και τη μείωση της υπερ-προσαρμογής.
- Συνδυασμός Προβλέψεων (Ensemble Learning): Συνδυάζει τις προβλέψεις από πολλά μοντέλα για τη βελτίωση της ακρίβειας και της γενικότητας των προβλέψεων.

Αυτές οι τεχνικές χρησιμοποιούνται για τη δημιουργία και τη βελτιστοποίηση των μοντέλων μηχανικής μάθησης χωρίς την ανάγκη εμπειρογνωμόνων. Τα έργα των Hutter, Kotthoff και

Vanschoren (2019) και του He και Wu (2019) περιγράφουν τις παραπάνω αρχές και τεχνικές που χρησιμοποιεί η AutoML[10].

Η Αυτόματη Μηχανική Μάθηση (AutoML) αποτελεί μια καινοτόμο προσέγγιση στο πεδίο της μηχανικής μάθησης, η οποία αποσκοπεί στην αυτοματοποίηση των διαδικασιών επιλογής, προεπεξεργασίας, εκπαίδευσης και αξιολόγησης μοντέλων μηχανικής μάθησης. Ο κυρίαρχος στόχος της AutoML είναι να διευκολύνει τον ερευνητή ή τον επαγγελματία της μηχανικής μάθησης στην αντιμετώπιση πολυπλοκών προβλημάτων χωρίς την ανάγκη ειδίκευσης στη σχεδίαση μοντέλων.

Για την επίτευξη του στόχου αυτού, η AutoML εφαρμόζει μια σειρά τεχνικών και μεθοδολογιών. Αρχικά, περιλαμβάνει την αυτόματη εξερεύνηση του χώρου υποθέσεων, όπου διάφοροι συνδυασμοί υπερπαραμέτρων εκτιμώνται και αξιολογούνται. Έπειτα, προβαίνει στην εκπαίδευση και αξιολόγηση των μοντέλων, χρησιμοποιώντας διαφορετικά υποσύνολα δεδομένων. Στη συνέχεια, η AutoML μπορεί να εφαρμόσει συνδυασμό μοντέλων για βελτιστοποίηση της απόδοσης. Στο επίπεδο της αναζήτησης μοντέλων, εξετάζονται διάφορες αρχιτεκτονικές μοντέλων, όπως νευρωνικά δίκτυα, δένδρα αποφάσεων και μοντέλα με βάση κανόνες, για την εύρεση του καταλληλότερου[11].

Τα πεδία εφαρμογής της AutoML είναι ευρέως ποικίλα, καλύπτοντας την κατηγοριοποίηση, την πρόβλεψη, τη συσταδοποίηση, την ανάλυση σειρών, την αναγνώριση φωνής και κειμένου, καθώς και την εξόρυξη γνώσης από δεδομένα. Σε πραγματικά προβλήματα, όπως η αναγνώριση προτύπων, η κατηγοριοποίηση και η πρόβλεψη, η AutoML επιτρέπει την αποτελεσματική χρήση της μηχανικής μάθησης ακόμη και από άτομα που δεν έχουν ειδίκευση στον τομέα[10].

Η εξέλιξη της Αυτόματης Μηχανικής Μάθησης (AutoML) έχει έναν σημαντικό αντίκτυπο στη σύγχρονη κοινωνία και τον τρόπο λειτουργίας των οικονομιών, ενώ έχει επίσης σημαντικές επιπτώσεις στον τομέα της απασχόλησης. Καταρχάς, η AutoML έχει δυναμική επίδραση στην καινοτομία και την ανάπτυξη των επιχειρήσεων[10]. Οι αυτοματοποιημένες διαδικασίες επιλογής και εκπαίδευσης μοντέλων επιτρέπουν στις εταιρείες να ανταποκριθούν γρηγορότερα στις ανάγκες της αγοράς και να αναπτύξουν προϊόντα και υπηρεσίες υψηλής ποιότητας και απόδοσης. Αυτό οδηγεί σε αύξηση της ανταγωνιστικότητας και της καινοτομίας σε διάφορους κλάδους της οικονομίας.

Ταυτόχρονα, όμως, η ευρεία υιοθέτηση της AutoML μπορεί να επηρεάσει την απασχόληση σε ορισμένους τομείς. Η αυτοματοποίηση των διαδικασιών εκπαίδευσης μοντέλων μπορεί να οδηγήσει σε μείωση της ανάγκης για εξειδικευμένες γνώσεις στον τομέα της μηχανικής μάθησης. Ταυτόχρονα, όμως, η ζήτηση για ειδικούς στην ανάπτυξη, τη διαχείριση και την επικοινωνία με τα μοντέλα μηχανικής μάθησης αυξάνεται. Επομένως, η μετάβαση στην χρήση της AutoML μπορεί να οδηγήσει σε αναδιαμόρφωση των απαιτούμενων δεξιοτήτων στον εργασιακό τομέα[10].

Σε κοινωνικό επίπεδο, η διάδοση της AutoML μπορεί να έχει επιπτώσεις στην πρόσβαση στην τεχνολογία και την ανισότητα. Ενώ η αυτοματοποίηση της μηχανικής μάθησης μπορεί να δημιουργήσει πιο εύκολη πρόσβαση σε εργαλεία και τεχνικές, η έλλειψη πρόσβασης σε αυτά λόγω

οικονομικών ή εκπαιδευτικών περιορισμών μπορεί να ενισχύσει τις υφιστάμενες ανισότητες.

Οι επιπτώσεις της AutoML στην κοινωνία και την απασχόληση είναι πολύπλοκες και ποικίλες. Είναι απαραίτητο να υπάρχει προσεκτική παρακολούθηση και ανάλυση των κοινωνικών και οικονομικών επιπτώσεων της εισαγωγής της AutoML, προκειμένου να διασφαλιστεί μια θετική προσαρμογή και αξιοποίησή της[11].

Η εφαρμογή της Αυτόματης Μηχανικής Μάθησης (AutoML) συνεπάγεται σημαντικές ευκαιρίες και προκλήσεις από δεοντολογική άποψη. Ενώ η αυτοματοποίηση της διαδικασίας επιλογής και εκπαίδευσης μοντέλων μπορεί να βελτιώσει την απόδοση και την αποτελεσματικότητα των συστημάτων μηχανικής μάθησης, είναι σημαντικό να ληφθούν υπόψη ορισμένες δεοντολογικές πτυχές προκειμένου να αποφευχθούν πιθανά προβλήματα και συνέπειες.

Ένα σημαντικό ηθικό ζήτημα είναι η διαφάνεια και η ευθύνη. Καθώς η AutoML εκτελεί αυτοματοποιημένα τη διαδικασία επιλογής και εκπαίδευσης μοντέλων, είναι σημαντικό να είναι διαφανείς οι αποφάσεις που λαμβάνονται από το σύστημα. Οι χρήστες πρέπει να έχουν κατανόηση για το πώς λειτουργεί η AutoML και ποιες είναι οι αποφάσεις που παίρνει, προκειμένου να μπορούν να εκτιμήσουν την αξιοπιστία των αποτελεσμάτων.

Επιπλέον, πρέπει να αντιμετωπιστούν θέματα δικαιοσύνης και προκατάληψης. Η AutoML εκπαιδεύει μοντέλα χρησιμοποιώντας ιστορικά δεδομένα, τα οποία μπορεί να περιέχουν προκαταλήψεις που αντανακλώνται στις αποφάσεις του μοντέλου. Είναι αναγκαίο να υπάρξει προσοχή ώστε να αποφευχθεί η ενίσχυση διακρίσεων και ανισοτήτων που ενδέχεται να προκύψουν από τα μοντέλα μηχανικής μάθησης.

Η διασφάλιση της ιδιωτικότητας και της προστασίας των δεδομένων είναι επίσης ζωτικής σημασίας. Καθώς η AutoML χρησιμοποιεί μεγάλο όγκο δεδομένων για την εκπαίδευση και τη βελτιστοποίηση των μοντέλων, πρέπει να ληφθούν μέτρα για να διασφαλιστεί ότι τα προσωπικά δεδομένα προστατεύονται και χρησιμοποιούνται σύμφωνα με τους νόμους και τις δεοντολογικές κατευθυντήριες γραμμές.

Η εφαρμογή της AutoML συνεπάγεται την ανάγκη για προσεκτική διαχείριση των δεοντολογικών πτυχών. Η διαφάνεια, η δικαιοσύνη και η προστασία των δεδομένων αποτελούν βασικούς πυλώνες για την αποδοχή και την επιτυχή χρήση της AutoML στην κοινωνία[10].

1.4 Κίνητρο και Συνεισφορά

Η ανάγκη για την υλοποίηση της εφαρμογής AutoDBSCAN απορρέει από την αυξανόμενη σύνθεση και πολυπλοκότητα των συγχρόνων αναλύσεων δεδομένων. Καθώς οι ποσότητες των διαθέσιμων δεδομένων αυξάνονται εκθετικά, η ανάγκη για αποτελεσματικούς και αυτοματοποιημένους αλγορίθμους είναι επιτακτική. Στον σύγχρονο κόσμο της επιστήμης των δεδομένων και της αναλυτικής πληροφορικής, η διερεύνηση των δεδομένων απαιτεί προηγμένα εργαλεία που θα διευκολύνουν τους χρήστες στην εφαρμογή και την αξιοποίηση αλγορίθμων μηχανικής μάθησης.

Ένας από τους κυριότερους λόγους που καθιστά επιθυμητή την υλοποίηση της εφαρμογής AutoDBSCAN είναι η απαιτούμενη εγκατάσταση λογισμικού για την εκτέλεση του αλγορίθμου DBSCAN. Ο αλγόριθμος αυτός, παρότι εξαιρετικά αποτελεσματικός για την ανακάλυψη συστάδων σε δεδομένα, είναι υπολογιστικά προσκοπτικός και απαιτητικός από άποψη υπολογιστικών πόρων. Οι τρέχουσες υλοποιήσεις του αλγορίθμου συχνά απαιτούν τη χρήση σύνθετων περιβαλλόντων και βιβλιοθηκών που μπορεί να μην είναι εύκολο να εγκατασταθούν από τους μη εξειδικευμένους χρήστες.

Αυτό το εμπόδιο μπορεί να αποθαρρύνει πολλούς ενδιαφερόμενους από το να αξιοποιήσουν τον αλγόριθμο DBSCAN για τις ανάγκες τους. Είναι εδώ που η εφαρμογή AutoDBSCAN αναλαμβάνει τον ρόλο της. Παρέχοντας έναν φιλικό προς τον χρήστη διαδικτυακό τομέα, η εφαρμογή διευκολύνει την εφαρμογή του αλγορίθμου DBSCAN, απαλλάσσοντας τους χρήστες από το βάρος της εγκατάστασης και ρύθμισης λογισμικού. Αυτή η προσέγγιση ενθαρρύνει την ευρύτερη χρήση του αλγορίθμου DBSCAN, ενισχύοντας την έρευνα και την εφαρμογή στον τομέα της ανάλυσης δεδομένων.

Αξίζει να αναφερθεί οτι στο πλαίσιο της εφαρμογής AutoDBSCAN, παρέχεται στους χρήστες μια πρόταση για τις τιμές των παραμέτρων εύρεσης συστάδων του αλγορίθμου DBSCAN, το επίπεδο επικάλυψης (επιτρεπτό σφάλμα) eps και το ελάχιστο αριθμό σημείων minPts. Ο σκοπός της πρότασης αυτής είναι να διευκολύνει τους χρήστες, ιδιαίτερα αν δεν έχουν εμπειρία στον τομέα της ανάλυσης δεδομένων, να ξεκινήσουν την ανάλυσή τους με βάση μία σχετικά κατάλληλη αρχική ρύθμιση. Ωστόσο, πρέπει να γίνει σαφές ότι οι προτεινόμενες τιμές είναι αυτόματες εκτιμήσεις και πρέπει να εξετάσουν και να προσαρμόσουν τις τιμές ανάλογα με τα δεδομένα και τους στόχους τους. Θα μπορούσε δηλαδή να καταταχθεί ως μέρος της AutoML καθώς εν μέρη δίνεται μια μορφή αυτοματοποίησης της διαδικασίας ομαδοποίηδης δεδομένων.

Η συνεισφορά της εφαρμογής AutoDBSCAN στον τομέα της ανάλυσης δεδομένων είναι σημαντική και πολυπρόσωπη. Εξετάζοντας τον τρόπο που η εφαρμογή αυτή συμβάλλει στην επιστημονική και επαγγελματική κοινότητα, μπορούμε να διακρίνουμε τις ακόλουθες πτυχές της συνεισφοράς.

Η εφαρμογή AutoDBSCAN συμβάλλει στον τομέα της ανάλυσης δεδομένων προσφέροντας μια απλή και φιλική προς τον χρήστη διεπαφή. Ενώ ο αλγόριθμος DBSCAN μπορεί να είναι πε-

ρίπλοκος στην υλοποίηση, η εφαρμογή αυτή αφαιρεί το βάρος της υπολογιστικής πολυπλοκότητας από τον χρήστη, επιτρέποντάς του να εφαρμόσει τον αλγόριθμο με μερικά απλά βήματα.

Συνεισφέρει στον τομέα ανάλυσης δεδομένων προσφέροντας γρήγορη και αποτελεσματική εφαρμογή του αλγορίθμου DBSCAN. Οι χρήστες δεν χρειάζεται πλέον να αποδεικνύουν τεχνική εμπειρογνωσία για να υλοποιήσουν τον αλγόριθμο στα δεδομένα τους, εξοικονομώντας χρόνο και πόρους.

Συμβάλλει επίσης στην ανάλυση δεδομένων προσφέροντας δυνατότητες προσαρμογής και επέκτασης. Το προγραμματιστικό της πλαίσιο παρέχει APIs που επιτρέπουν στους χρήστες να δημιουργήσουν τις δικές τους προσαρμοσμένες εφαρμογές, ενισχύοντας την ευελιξία και την προσαρμοστικότητα του συστήματος.

Επιπλέον, αναδεικνύει τη σημαντική προσφορά της στον τομέα της έρευνας και της καινοτομίας, προσφέροντας ένα εργαλείο που διευκολύνει την απλή δοκιμή και αξιολόγηση του αλγορίθμου DBSCAN σε νέα σενάρια και δεδομένα. Αυτό ανοίγει τον δρόμο για την ανακάλυψη νέων προσεγγίσεων και την εξαγωγή νέων ενδιαφέρουσων αποτελεσμάτων στον τομέα της ανάλυσης δεδομένων.

Συνολικά, συμβάλλει δυναμικά στον τομέα της ανάλυσης δεδομένων, παρέχοντας μια προσβάσιμη, αποτελεσματική και προσαρμόσιμη λύση για την εφαρμογή του αλγορίθμου DBSCAN. Με την αφαίρεση των εμποδίων και την αύξηση της ευκολίας και της ταχύτητας της ανάλυσης δεδομένων, η εφαρμογή αυτή ενισχύει την προώθηση της έρευνας, της καινοτομίας στον τομέα και συντελεί στην επιστημονική πρόοδο σε αυτόν τον δυναμικό και αναπτυσσόμενο τομέα.

1.5 Οργάνωση της εργασίας

Η δομή της εργασίας οργανώνεται σε διάφορες ενότητες, και σε αυτό το κείμενο θα παρουσιάσουμε μια επισκόπηση των κύριων ενοτήτων χωρίς να επικεντρωθούμε στις λεπτομέρειες κάθε μιας από αυτές.

Στην Ενότητα 2, προσεγγίζουμε το θέμα της Συσταδοποίησης βάση πυκνότητας. Εδώ, εξετάζουμε την πυκνότητα των δεδομένων ως κριτήριο για τη δημιουργία συστάδων και παρουσιάζουμε τον αλγόριθμο DBSCAN. Επίσης, αναφέρουμε τη σημασία των παραμέτρων eps και minpts στον αλγόριθμο, καθώς και τα πλεονεκτήματα και μειονεκτήματά του. Επιπλέον, εξετάζουμε παραλλαγές του αλγορίθμου.

Στη Ενότητα 3, ασχολούμαστε με την επιλογή τεχνολογιών. Εδώ, εξετάζουμε τις τεχνολογίες που χρησιμοποιούνται τόσο στον server όσο και στον client για την υλοποίηση του AutoDBSCAN. Συμπεριλαμβάνουμε επίσης την εξήγηση της χρήσης της TypeScript.

Στη Ενότητα 4, εστιάζουμε στον Σχεδιασμό και την Υλοποίηση του AutoDBSCAN. Εδώ, παρουσιάζουμε τις λειτουργικές απαιτήσεις του συστήματος, την αρχιτεκτονική του AutoDBSCAN, και την υλοποίηση του εξυπηρετητή και του πελάτη. Επιπλέον, αναφέρουμε το Github repository που περιέχει τον κώδικα του συστήματος.

Στη Ενότητα 5, παρουσιάζουμε το AutoDBSCAN, αναλύοντας διάφορες σελίδες και λειτουργίες που περιλαμβάνονται σε αυτό, όπως την αρχική σελίδα, την εγγραφή χρήστη, τη σύνδεση χρήστη στο σύστημα, την επεξεργασία προσωπικών στοιχείων, τη διαγραφή λογαριασμού, καθώς και διάφορες λειτουργίες που σχετίζονται με τη συσταδοποίηση DBSCAN.

Στη Ενότητα 6, αξιολογούμε το AutoDBSCAN και εξετάζουμε την εμπειρία χρήσης μέσω της μετρικής SUS.

Τέλος, στη Ενότητα 7, παρουσιάζουμε τα συμπεράσματα της εργασίας και αναφέρουμε μελλοντικές επεκτάσεις και βελτιώσεις που μπορούν να εφαρμοστούν στο AutoDBSCAN.

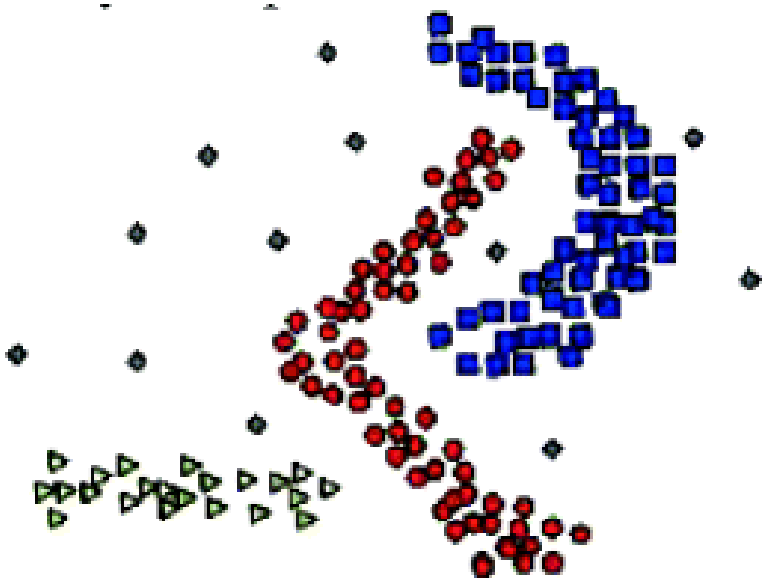
Κεφάλαιο 2

Συσταδοποίηση βάση πυκνότητας

2.1 Η Πυκνότητα των δεδομένων ως κριτήριο δημιουργίας συστάδων

Η πυκνότητα των δεδομένων είναι ένας ορισμός που συχνά χρησιμοποιείται στην εξόρυξη δεδομένων και τη μηχανική μάθηση, κυρίως στην τεχνική της συσταδοποίησης. Σε αυτό το πλαίσιο, η πυκνότητα αναφέρεται στον αριθμό των δεδομένων εντός ενός συγκεκριμένου χώρου και πώς αυτοί είναι διασκορπισμένοι στον χώρο αυτόν [12].

Η χρήση της πυκνότητας στη δημιουργία συστάδων αποτελεί στρατηγική που επιτρέπει την οργάνωση των δεδομένων βάσει της κατανομής τους μέσα στο χώρο των χαρακτηριστικών[1]. Η πυκνότητα καθορίζεται ως ο αριθμός των σημείων εντός μιας σφαίρας συγκεκριμένης ακτίνας γύρω από κάθε σημείο, χρησιμοποιώντας αυτή την έννοια για να εντοπίσει περιοχές που περιέχουν μεγάλο αριθμό γειτονικών σημείων [5]. Αλγόριθμοι όπως ο DBSCAN εξειδικεύονται στη χρήση της πυκνότητας για τη δημιουργία συστάδων, χωρίς την ανάγκη καθορισμού του αριθμού των συστάδων εκ των προτέρων και εντοπίζουν περίπλοκες δομές[1]. Επιπλέον αλγόριθμοι όπως ο OPTICS παρέχουν λύσεις για τη διαχείριση περίπλοκων δομών και θορύβου, αποφεύγοντας τα προβλήματα που εμφανίζουν οι παραδοσιακές μέθοδοι. Οι μέθοδοι αυτοί έχουν εφαρμογές σε διάφορους τομείς όπως η εξόρυξη δεδομένων, η ανάλυση εικόνων, και η βιοπληροφορική[1]. Συνολικά, η χρήση της πυκνότητας στη δημιουργία συστάδων προσφέρει μια ισχυρή και ευέλικτη προσέγγιση που μπορεί να αποκαλύψει την υποκείμενη δομή των δεδομένων, αποτελώντας σημαντικό εργαλείο στην εξόρυξη δεδομένων και τη μηχανική μάθηση[4].



Σχήμα 2.1: Density Based Clustering Example

Η πυκνότητα στη συσταδοποίηση διαδραματίζει κρίσιμο ρόλο και έχει ποικίλες εφαρμογές σε διάφορους τομείς. Ορίζοντας την πυκνότητα ως τον αριθμό των σημείων εντός ενός συγκεκριμένου χώρου, η πυκνότητα αποτελεί τη βάση για αλγορίθμους συσταδοποίησης όπως ο DBSCAN[1].

- Εξόρυξη Δεδομένων: Η ανίχνευση περιοχών υψηλής πυκνότητας βοηθάει στην ανακάλυψη χρήσιμων πληροφοριών από τα δεδομένα και την εποπτεία των προτύπων[13].
- Ανάλυση Εικόνων: Στην ανάλυση εικόνων, η πυκνότητα χρησιμοποιείται για την εντοπισμός και την ομαδοποίηση των αντικειμένων μέσα σε μια εικόνα.
- Βιοπληροφορική: Στη βιοπληροφορική, η πυκνότητα βοηθά στην κατηγοριοποίηση των πρωτεϊνών και των γονιδίων, επιτρέποντας την κατανόηση της βιολογικής συμπεριφοράς[5].

Η χρήση της πυκνότητας στη συσταδοποίηση προσφέρει πολλαπλά πλεονεκτήματα. Πρώτον, επιτρέπει την ανακάλυψη περίπλοκων δομών, όπως συστάδες μη συμμετρικού σχήματος[1]. Δεύτερον, δεν απαιτεί την προκαθορισμένη επιλογή του αριθμού των συστάδων. Τρίτον, είναι ανθεκτική στο θόρυβο, εντοπίζοντας συστάδες ακόμη και μέσα από θορυβώδη δεδομένα[5]. Εν κατακλείδι, η πυκνότητα προσφέρει μια δυναμική και ευέλικτη προσέγγιση στη συσταδοποίηση, καθιστώντας την κατάλληλη για πολλές εφαρμογές και τομείς.

2.2 Ο Αλγόριθμος DBSCAN

Ο αλγόριθμος DBSCAN (Density-Based Spatial Clustering of Applications with Noise) είναι ένας από τους πιο δημοφιλείς αλγορίθμους συσταδοποίησης που βασίζεται στην πυκνότητα[1]. Στόχος του είναι η εύρεση περιοχών υψηλής πυκνότητας που ξεπερνούν κάποιο καθορισμένο όριο, διαχωρίζοντας ταυτόχρονα τα δείγματα χαμηλής πυκνότητας ως θόρυβο.

Είναι ένας πρωτοποριακός αλγόριθμος συσταδοποίησης που αναπτύχθηκε από τους Martin Ester, Hans-Peter Kriegel, Jörg Sander, και Xiaowei Xu. Ο αλγόριθμος προτάθηκε για πρώτη φορά το 1996 και ενσωμάτωσε την ιδέα της πυκνότητας στην εξεύρεση συστάδων, σε αντίθεση με τους προηγούμενους αλγορίθμους που βασίζονταν κυρίως στις αποστάσεις μεταξύ των σημείων[1].

Ο DBSCAN καθιέρωσε ένα νέο παράδειγμα στη συσταδοποίηση, επιτρέποντας την ανακάλυψη συστάδων διαφορετικών σχημάτων και μεγεθών, χωρίς την ανάγκη για τον προκαθορισμό του αριθμού των συστάδων. Αυτό τον καθιστά ιδιαίτερα χρήσιμο σε πολλές πραγματικές εφαρμογές, όπου οι δομές των δεδομένων δεν είναι πάντα γνωστές εκ των προτέρων.

Η εφαρμογή του αλγορίθμου σε τόσο ευρεία γκάμα εφαρμογών, όπως η γεωγραφική πληροφορική, η βιολογία, και η ψηφιακή επεξεργασία εικόνων, οδήγησε σε αναθεώρηση και βελτίωση του αλγορίθμου καθώς και στη δημιουργία πολλών παραλλαγών του[5].

Ο αλγόριθμος DBSCAN αποτελεί ένα από τα κλασικά εργαλεία στην εξόρυξη δεδομένων και παραμένει σημαντικός ακόμη και μετά από δύο δεκαετίες, με την έρευνα να συνεχίζεται για τη βελτίωση και προσαρμογή του σε νέες εφαρμογές.

Διαδικασία Λειτουργίας του Αλγορίθμου DBSCAN
Παρακάτω περιγράφεται αναλυτικά η διαδικασία λειτουργίας του:

1. Είσοδος Παραμέτρων:
   Ο αλγόριθμος δέχεται δύο κύριες παραμέτρους: το ε, που καθορίζει το εύρος της γειτονιάς, και το MinPts, που καθορίζει τον ελάχιστο αριθμό σημείων για να θεωρηθεί μια γειτονιά πυκνή.

2. Εύρεση Γειτόνων:
   Για κάθε σημείο, ο αλγόριθμος βρίσκει όλα τα σημεία που βρίσκονται σε απόσταση μικρότερη ή ίση από ε. Αυτά τα σημεία αποτελούν τη γειτονιά του σημείου.

3. Καθορισμός Κεντρικών Σημείων:
   Αν ένα σημείο έχει τουλάχιστον MinPts σημεία στη γειτονιά του, τότε θεωρείται κεντρικό σημείο.

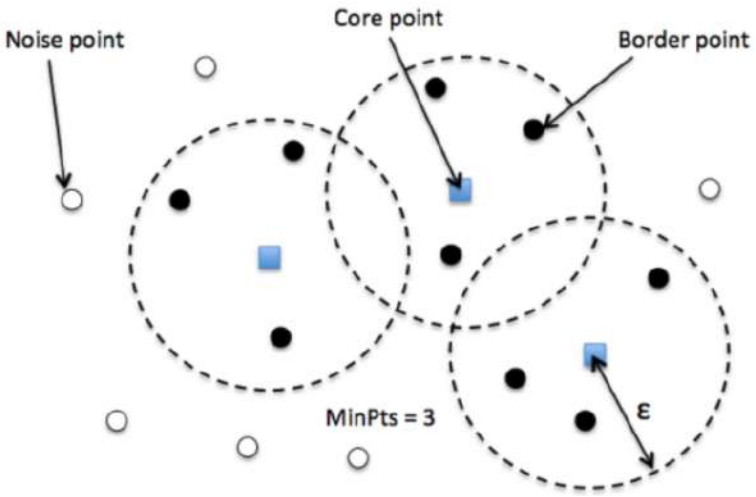
4. Εύρεση Συστάδων:

   - Επιλέγεται ένα τυχαίο σημείο που δεν έχει εξεταστεί.

- Εάν το σημείο είναι κεντρικό, δημιουργείται μια νέα συστάδα, και όλα τα σημεία της γειτονιάς προστίθενται στη συστάδα. Η διαδικασία επαναλαμβάνεται για τα σημεία της γειτονιάς και τα σημεία των γειτονιών τους.
- Εάν το σημείο δεν είναι κεντρικό και δεν ανήκει σε καμία γειτονιά, τότε θεωρείται θόρυβος.
- Η διαδικασία επαναλαμβάνεται μέχρι να εξεταστούν όλα τα σημεία.
-

5. Έξοδος

Ο αλγόριθμος επιστρέφει τις βρεθείσες συστάδες και τα σημεία θορύβου.



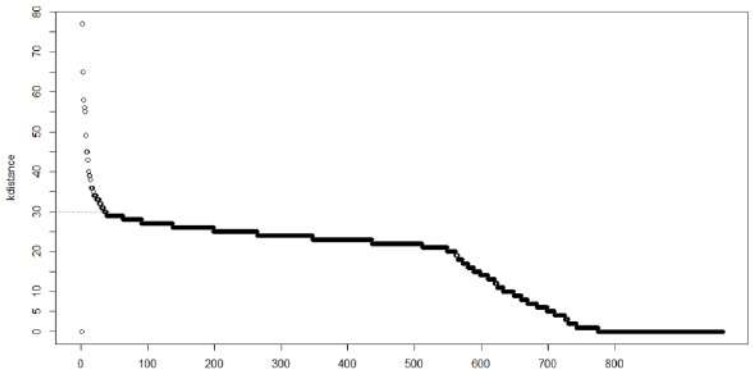
Σχήμα 2.2: DBSCAN

2.3 Προσδιορισμός των παραμέτρων eps και minpts

Προσδιορισμός της παραμέτρου *MinPts*
Για να προκύψει η ελάχιστη τιμή για το *MinPts*, ένας καλός κανόνας είναι να το ορίσετε ώστε να είναι μεγαλύτερο ή ίσο με τον αριθμό των διαστάσεων *D* στο σύνολο δεδομένων, δηλαδή $MinPts \geq D + 1$.
Μια χαμηλή τιμή του *MinPts*, όπως 1, δεν έχει νόημα επειδή οδηγεί στη δημιουργία συστάδας για κάθε σημείο.
Εάν $MinPts \leq 2$, το αποτέλεσμα θα είναι παρόμοιο με ιεραρχική συσταδοποίηση με τη μέτρηση μονού συνδέσμου, όπου το δενδρόγραμμα κόβεται στο ύψος *eps*. Επομένως, το *MinPts* πρέπει να είναι τουλάχιστον 3.
Ωστόσο, μεγαλύτερες τιμές είναι συνήθως καλύτερες για σύνολα δεδομένων με θόρυβο, καθώς οδηγούν σε πιο σημαντικές συστάδες[1]. Μια τιμή του $MinPts = 2 \cdot D$ είναι ένας καλός κανόνας, αλλά ενδέχεται ακομα μεγαλύτερες τιμές να είναι απαραίτητες για μεγάλα ή θορυβώδη σύνολα δεδομένων ή αυτά που περιέχουν πολλά διπλότυπα.

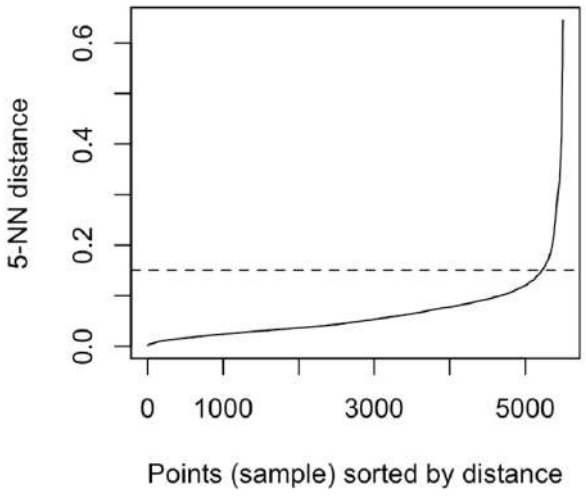
K-Distance plot

Σε μια συσταδοποίηση με $MinPts = k$, περιμένουμε ότι η k-απόσταση των βασικών και των συνοριακών σημείων θα είναι εντός ενός συγκεκριμένου εύρους, ενώ τα σημεία θορύβου μπορεί να έχουν πολύ μεγαλύτερη k-απόσταση, και έτσι μπορούμε να παρατηρήσουμε ένα σημείο αύξησης της καμπύλης στο διάγραμμα k-απόστασης[5]. Ωστόσο, μερικές φορές μπορεί να μην υπάρχει προφανές σημείο αύξησης, ή μπορεί να υπάρχουν πολλαπλά σημεία αύξησης, τα οποία καθιστούν την απόφαση δύσκολη.



Σχήμα 2.3: k-distance plot

Η μέθοδος που προτείνεται εδώ αποτελείται από τον υπολογισμό των αποστάσεων των k-πλησιέστερων γειτόνων σε έναν πίνακα σημείων. Η ιδέα είναι να υπολογίσουμε τον μέσο όρο των αποστάσεων κάθε σημείου προς τους k-πλησιέστερους γείτονές του. Η τιμή του *k* θα καθοριστεί από τον χρήστη και αντιστοιχεί στο *MinPts*. Στη συνέχεια, αυτές οι k-αποστάσεις απεικονίζονται με αύξουσα σειρά. Ο στόχος είναι να προσδιοριστεί το *knee*, το οποίο αντιστοιχεί στη βέλτιστη παράμετρο *eps*. Ένα *knee* αντιστοιχεί σε ένα κατώφλι όπου προκύπτει μια οξεία αλλαγή κατά μήκος της καμπύλης k-απόστασης[5].



Σχήμα 2.4: K-Dist graph example

Προσδιορισμός της παραμέτρου *eps*
Η επιλογή της κατάλληλης τιμής για το *eps* επηρεάζει απευθείας το αποτέλεσμα της συσταδοποίησης και είναι μια από τις πιο δύσκολες πτυχές της χρήσης του DBSCAN. Ακολουθεί μια αναλυτική περιγραφή του πώς μπορεί να προσδιοριστεί η τιμή του *eps*.

Η κατανόηση των δεδομένων είναι το πρώτο βασικό βήμα. Πρέπει να κατανοήσουμε τη φύση και την κλίμακα των δεδομένων πάνω στα οποία εργαζόμαστε. Τα διάφορα χαρακτηριστικά των δεδομένων και οι διαφορές στις μονάδες μέτρησης μπορεί να απαιτούν διαφορετικές τιμές του *eps*. Ένας συνηθισμένος τρόπος εκτίμησης του *eps* είναι να χρησιμοποιήθεί ένα διάγραμμα k-απόστασης όπως εξηγήθηκε στην προηγούμενη παράγραφο. Οι αυξήσεις καμπυλότητας στο διάγραμμα μπορεί να δείξουν μια κατάλληλη τιμή για το *eps*. Η πειραματική ρύθμιση της τιμής του *eps* σε συνδυασμό με τεχνικές διασταυρούμενης επικύρωσης μπορεί να βοηθήσει στην εύρεση μιας τιμής που αποσκοπεί στην καλύτερη απόδοση για μια συγκεκριμένη εφαρμογή[5].
Σε περιπτώσεις όπου υπάρχει προηγούμενη γνώση ή εμπειρία με τα δεδομένα, η επιλογή της τιμής του *eps* μπορεί να γίνει με βάση την ειδική γνώση.
Η επιλογή του κατάλληλου *eps* είναι κρίσιμη για την επιτυχία του αλγορίθμου DBSCAN και απαιτεί κατανόηση των δεδομένων, πειραματισμό, και πιθανώς ειδική γνώση. Δεν υπάρχει μια τιμή που ταιριάζει σε όλα, και η κατανόηση της ειδικής περίπτωσης χρήσης είναι συχνά απαραίτητη για την επιλογή της σωστής τιμής

2.4 Πλεονεκτήματα του DBSCAN

Διαχείριση Θορύβου
Ο αλγόριθμος ορίζει σαφώς τα "πυρηνικά" σημεία που βρίσκονται σε περιοχές υψηλής πυκνότητας και τα σημεία στα όρια των συστάδων, ενώ ταυτόχρονα είναι σε θέση να κατατάξει τα υπόλοιπα σημεία ως θόρυβο. Αυτό επιτρέπει την απομόνωση των σημείων που δεν ανήκουν σε καμία συστάδα και θεωρούνται "θορυβώδη"[1]. Το χαρακτηριστικό αυτό καθιστά τον DBSCAN ιδιαίτερα χρήσιμο σε εφαρμογές όπου η παρουσία θορύβου είναι σημαντικός παράγοντας και πρέπει να αγνοηθεί κατά την προσπάθεια εύρεσης των πραγματικών συστάδων στα δεδομένα[5].

Ευελιξία στο Σχήμα των Συστάδων
Ο αλγόριθμος DBSCAN είναι γνωστός για την ευελιξία του στο σχήμα των συστάδων που μπορεί να ανιχνεύσει. Σε αντίθεση με τους αλγορίθμους συσταδοποίησης που βασίζονται σε αποστάσεις και προσπαθούν να αναγκάσουν τις συστάδες σε σφαιρικά ή υπερσφαιρικά σχήματα, ο DBSCAN εργάζεται με βάση την πυκνότητα. Αυτό σημαίνει ότι μπορεί να ανιχνεύσει συστάδες διαφορετικών σχημάτων και μεγεθών, χωρίς να υποθέτει μια συγκεκριμένη γεωμετρική δομή[1]. Η ικανότητα αυτή έχει καταστήσει τον DBSCAN έναν πολύτιμο εργαλείο σε πολλούς τομείς, όπου τα δεδομένα μπορεί να παρουσιάζουν πολύπλοκες δομές, όπως τη βιολογία, την αστρονομία και τη γεωγραφική πληροφορική[14].

Εφαρμόζεται Χωρίς Προκαθορισμένο Αριθμό Συστάδων
Διαφέρει από πολλούς άλλους αλγορίθμους συσταδοποίησης καθώς δεν απαιτεί τον προκαθορισμό του αριθμού των συστάδων. Αντί για αυτό, ο DBSCAN χρησιμοποιεί την έννοια της

πυκνότητας, όπου μια συστάδα αποτελείται από τουλάχιστον έναν ελάχιστο αριθμό σημείων *MinPts* που βρίσκονται μέσα σε μια δεδομένη ακτίνα *eps*. Αυτό επιτρέπει στον αλγόριθμο να ανακαλύψει αυτόματα τον αριθμό των συστάδων που είναι απαραίτητος για να περιγράψει τα δεδομένα, βάσει των εννοιών της πυκνότητας που έχουν καθοριστεί[1]. Αυτή η ιδιότητα κάνει τον DBSCAN ιδιαίτερα χρήσιμο σε σενάρια όπου ο αριθμός των συστάδων δεν είναι γνωστός εκ των προτέρων, ή όπου η δομή των δεδομένων είναι πολύπλοκη και δύσκολη να προσδιοριστεί με μεθόδους που απαιτούν προκαθορισμένο αριθμό συστάδων[14].

Ανθεκτικός στις Εξωτερικές Επιρροές

Αυτή η ικανότητα προκύπτει από τη χρήση της έννοιας της πυκνότητας και την απόρριψη των σημείων που δεν ικανοποιούν τα κριτήρια της πυκνότητας ως θόρυβο [1]. Συγκεκριμένα, τα σημεία που δεν ανήκουν σε καμία συστάδα με βάση τα καθορισμένα κριτήρια πυκνότητας (*MinPts* και *eps*) απορρίπτονται ως θόρυβος, επιτρέποντας έτσι στον αλγόριθμο να εντοπίζει συστάδες που βρίσκονται σε περιοχές υψηλής πυκνότητας και να αγνοεί τα απομονωμένα σημεία που ενδεχομένως να είναι αποτέλεσμα θορύβου ή εξωτερικών επιρροών[15]. Αυτό καθιστά τον DBSCAN ιδιαίτερα χρήσιμο σε περιπτώσεις όπου τα δεδομένα περιέχουν πολλές ασυνήθειες ή θόρυβο.

Ικανότητα Εύρεσης Συστάδων Υψηλής Πυκνότητας

Αυτό επιτυγχάνεται μέσω της χρήσης της έννοιας της πυκνότητας, όπου μια συστάδα ορίζεται ως μια περιοχή στον χώρο των δεδομένων που περιέχει περισσότερα σημεία από κάποιο καθορισμένο κατώφλι[1]. Ο αλγόριθμος αρχίζει από ένα τυχαίο σημείο και επεκτείνει τη συστάδα προσθέτοντας γειτονικά σημεία που βρίσκονται εντός ενός καθορισμένου εύρους *eps* και έχουν τουλάχιστον *MinPts* γείτονες εντός αυτού του εύρους. Η δυνατότητα αυτή να εντοπίζει περιοχές υψηλής πυκνότητας καθιστά τον DBSCAN κατάλληλο για εφαρμογές όπου η πυκνότητα των δεδομένων είναι κρίσιμη για την εξαγωγή πληροφορίας[5].

Παρέχει Διαστασιακή Ανεξαρτησία

Αντί να εξετάζει την απόσταση μεταξύ των σημείων σε μια ευκλείδια βάση, όπως κάνουν οι περισσότεροι άλλοι αλγόριθμοι συσταδοποίησης, ο DBSCAN λειτουργεί βάσει πυκνότητας, επιτρέποντας την ανίχνευση συστάδων ανεξάρτητα από την γεωμετρική τους μορφή και τον αριθμό των διαστάσεων του δειγματοληπτικού χώρου[1]. Αυτή η προσέγγιση καθιστά τον DBSCAN ευέλικτο και ικανό να χειριστεί δεδομένα με διάφορες διαστάσεις, προσφέροντας ένα πλεονέκτημα σε σχέση με αλγορίθμους που βασίζονται σε ειδικότερες υποθέσεις για την δομή των δεδομένων.

Απλότητα και Εύκολη Εφαρμογή

Σε αντίθεση με πολλούς άλλους αλγορίθμους συσταδοποίησης που απαιτούν την προκαθορισμένη επιλογή του αριθμού των συστάδων, ο DBSCAN διεξάγει συσταδοποίηση βάσει πυκνότητας, με δύο βασικές παραμέτρους: το εύρος γειτονιάς *eps* και τον ελάχιστο αριθμό σημείων που απαιτούνται για τη δημιουργία μιας συστάδας *MinPts*. Αυτό τον καθιστά πιο άμεσο και εύκολο στη χρήση, χωρίς την ανάγκη για περίπλοκες αποφάσεις σχετικά με την αρχικοποίηση και τις υποθέσεις για τη γεωμετρία των συστάδων. Επιπλέον, πολλές βιβλιοθήκες λογισμικού παρέχουν υλοποιήσεις του DBSCAN, γεγονός που διευκολύνει την εφαρμογή του σε πραγματικά προβλήματα[1].

2.5 Μειονεκτήματα του DBSCAN

Ευαισθησία στις Παραμέτρους
Ο αλγόριθμος DBSCAN είναι ευαίσθητος στις παραμέτρους *eps* και *MinPts*, οι οποίες πρέπει να επιλεγούν προσεκτικά για κάθε σύνολο δεδομένων [1]. Η επιλογή ακατάλληλων παραμέτρων μπορεί να οδηγήσει σε ανεπιθύμητα αποτελέσματα συσταδοποίησης.

Δυσκολία με Διαφορετικές Πυκνότητες
Όταν οι συστάδες διαφέρουν σημαντικά σε πυκνότητα, ο DBSCAN μπορεί να αποτύχει να τις αναγνωρίσει σωστά. Αυτό οφείλεται στο ότι οι παράμετροι είναι σταθεροί για όλο το σύνολο δεδομένων [5].

Δυσκολία με Δεδομένα Υψηλής Διάστασης
Η απόδοση του DBSCAN μπορεί να μειωθεί σε υψηλές διαστάσεις λόγω της κατάρας της διαστατικότητας. Η εκτίμηση της πυκνότητας γίνεται πιο πολύπλοκη [16].

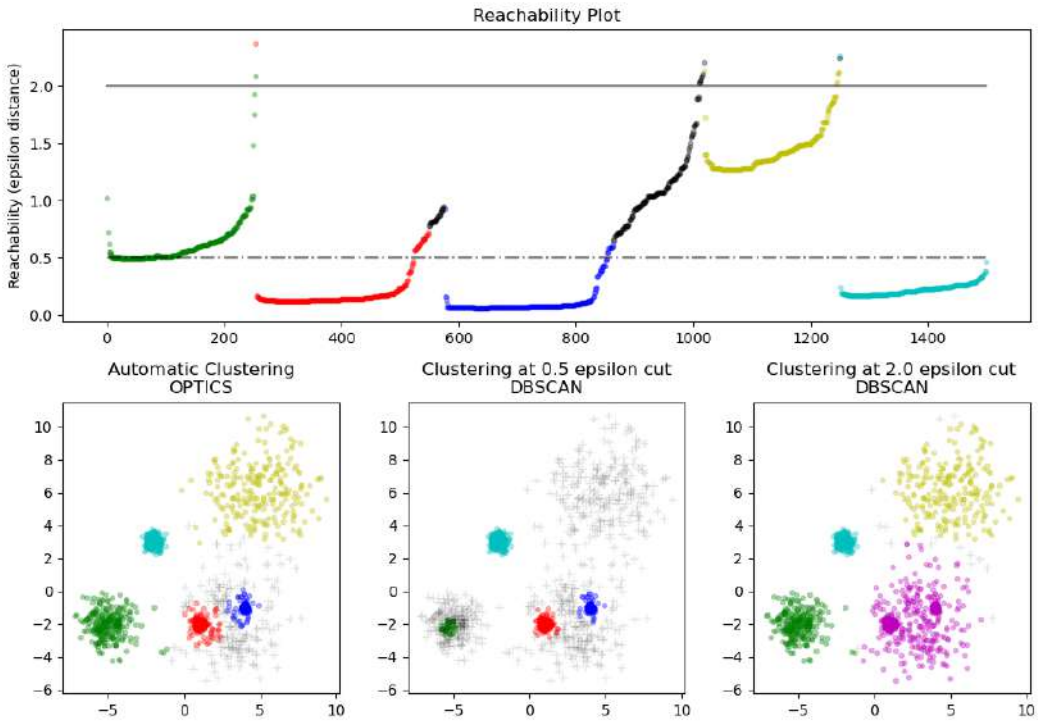
Απαιτήσεις σε Υπολογιστικούς Πόρους
Η ανάγκη για εύρεση γειτόνων μπορεί να απαιτήσει σημαντικούς υπολογιστικούς πόρους, ιδιαίτερα για μεγάλα δεδομένα, κάτι που μπορεί να καθιστά τον αλγόριθμο λιγότερο ελκυστικό για ορισμένες εφαρμογές [12].

Ανικανότητα Ανίχνευσης Συστάδων Χαμηλής Πυκνότητας
Η ανικανότητα του DBSCAN να ανιχνεύσει συστάδες χαμηλής πυκνότητας μπορεί να περιορίζει την εφαρμογή του σε ορισμένα σύνολα δεδομένων [17].

2.6 Παραλλαγές του αλγορίθμού

OPTICS (Ordering Points To Identify the Clustering Structure)

Ο αλγόριθμος OPTICS (Ordering Points To Identify the Clustering Structure) είναι μια παραλλαγή του DBSCAN που σχεδιάστηκε για να διορθώσει την ευαισθησία του DBSCAN σε μία ενιαία παράμετρο πυκνότητας. Σε αντίθεση με τον DBSCAN, ο OPTICS δεν απαιτεί την επιλογή μιας ενιαίας τιμής εμβέλειας για την εύρεση γειτόνων, αλλά αντίθετα διαπραγματεύεται την εμβέλεια γειτονιάς στο διάστημα των δεδομένων, επιτρέποντας την αναγνώριση συστάδων με διαφορετικές πυκνότητες. Αυτό καθιστά τον OPTICS ικανό να αντιμετωπίσει δεδομένα που παρουσιάζουν διαφορετικές κλίμακες και πυκνότητες σε διαφορετικά τμήματα του χώρου. Το αποτέλεσμα είναι μια διάταξη των δεδομένων που παρέχει μια πολυεπίπεδη αναπαράσταση της δομής συστάδων, επιτρέποντας την περαιτέρω επεξεργασία και ανάλυση [18].



Σχήμα 2.5: Scikit learn Demo Example for OPTICS

HDBSCAN (Hierarchical DBSCAN)

Ο αλγόριθμος HDBSCAN (Hierarchical DBSCAN) είναι μια επέκταση του κλασικού DBSCAN, που χρησιμοποιεί ιεραρχική δενδροδόμηση για την εύρεση των συστάδων. Αντί να χρησιμοποιεί μια σταθερή τιμή εύρους, ο HDBSCAN επιτρέπει την εύρεση συστάδων διαφόρων μεγεθών και σχημάτων μέσω της ρύθμισης ενός παράγοντα απότομης αλλαγής που ελέγχει την ένταση της συμπύκνωσης των συστάδων. Αυτό τον καθιστά ιδιαίτερα ευέλικτο στη διαχείριση διαφόρων δομών δεδομένων [19]. Το αποτέλεσμα είναι ένας αλγόριθμος που μπορεί να αναδείξει πολύπλοκες δομές στα δεδομένα, παρέχοντας ένα πιο πλούσιο και ενδιαφέρον αποτέλεσμα σε σύγκριση με τον αρχικό DBSCAN.

GDBSCAN (Grid-based DBSCAN)

Ο αλγόριθμος GDBSCAN (Grid-based DBSCAN) είναι μια παραλλαγή του κλασικού DBSCAN που χρησιμοποιεί μια προκαθορισμένη δομή πλέγματος για να αυξήσει την αποδοτικότητα της επεξεργασίας. Τα δεδομένα διαμερίζονται σε κελιά πλέγματος, και η διαδικασία συσταδοποίησης εφαρμόζεται μόνο στα κελιά που περιέχουν δεδομένα. Αυτό επιτρέπει στον αλγόριθμο να παραλείπει την επεξεργασία των κενών περιοχών, μειώνοντας έτσι σημαντικά τον χρόνο υπολογισμού [20]. Αν και ο GDBSCAN διατηρεί πολλές από τις ιδιότητες του αρχικού DBSCAN, η εισαγωγή της πλέγματος δομής μπορεί να έχει επιπτώσεις στην ποιότητα της συσταδοποίησης, ανάλογα με την επιλογή του μεγέθους του κελιού.

Ενσωματωμένος DBSCAN

Ο Ενσωματωμένος DBSCAN είναι μια παραλλαγή του κλασικού DBSCAN που περιλαμβάνει την ενσωμάτωση επιπρόσθετων χαρακτηριστικών ή πληροφοριών στη διαδικασία συσταδοποίησης. Συγκεκριμένα, η παραλλαγή αυτή μπορεί να χρησιμοποιήσει πληροφορίες όπως οι προηγούμενες κατηγοριοποιήσεις ή τα βάρη των χαρακτηριστικών για να καθοδηγήσει τη συσταδοποίηση [21]. Η ενσωμάτωση αυτών των επιπρόσθετων πληροφοριών μπορεί να βοηθήσει στην ανάδειξη πιο περίπλοκων μοτίβων και στην παραγωγή πιο εύστοχων συστάδων. Η εφαρμογή του Ενσωματωμένου DBSCAN απαιτεί, ωστόσο, προσεκτική επιλογή και προετοιμασία των επιπρόσθετων πληροφοριών για να αποφευχθούν παραμορφώσεις στα αποτελέσματα.

Κεφάλαιο 3

Επιλογή τεχνολογιών

Οι τεχνολογίες που επιλέγουμε για την ανάπτυξη ενός software project, όπως το AutoDBSCAN, αποτελούν τον πυρήνα της δημιουργίας του. Οι επιλογές αυτές διαμορφώνουν όχι μόνο τον τρόπο με τον οποίο θα αναπτυχθεί το project, αλλά και την απόδοση, την ασφάλεια, την ευελιξία και την προσαρμοστικότητα του τελικού προϊόντος.

Σε αυτήν την ενότητα, θα εξετάσουμε πώς οι συγκεκριμένες τεχνολογίες που επιλέχθηκαν για το AutoDBSCAN - τόσο για το backend (Node.js, Express.js, JSON Web Tokens) όσο και για το frontend (Next.js, React, Axios) - συνέβαλαν στην επίτευξη των στόχων του project. Με αυτόν τον τρόπο, θα μπορείτε να κατανοήσετε τις σκέψεις και τις διαδικασίες που οδήγησαν στην επιλογή των συγκεκριμένων τεχνολογιών και πώς αυτές συνεισφέρουν στην ολοκληρωμένη λειτουργικότητα του AutoDBSCAN.

3.1 Τεχνολογίες Server side

Node.js
Το Node.js είναι ένα ανοιχτού κώδικα, cross-platform, back-end, JavaScript runtime περιβάλλον που εκτελείται σε επίπεδο εξυπηρετητή. Δημιουργήθηκε το 2009 από τον Ryan Dahl και αποτελείται από την JavaScript V8 μηχανή της Google, με την προσθήκη μιας συλλογής βιβλιοθηκών για εργασίες επιπέδου εξυπηρετητή, όπως δρομολόγηση URLs και I/O αρχείων [22].

Μερικές από τις κύριες δυνατότητες του Node.js περιλαμβάνουν:

- Ασύγχρονη I/O: Το Node.js προσφέρει μη-αποκλειστική I/O, που επιτρέπει την παράλληλη εκτέλεση διαφόρων εργασιών και αποτρέπει την αναμονή για ολοκλήρωση μίας εργασίας πριν προχωρήσει στην επόμενη .
- Single Threaded Event Loop: Το Node.js λειτουργεί σε μια μονονηματική λούπα γεγονότων, το οποίο επιτρέπει την επεξεργασία πολλαπλών αιτήσεων παράλληλα χωρίς την ανάγκη για πολλαπλά threads[23].
- Cross-platform: Το Node.js μπορεί να λειτουργήσει σε διάφορες πλατφόρμες, όπως Windows, Linux, Unix, Mac OS X, κ.λπ. Αυτό το καθιστά ιδανικό για την ανάπτυξη εφαρμογών που

πρέπει να λειτουργούν σε πολλά διαφορετικά συστήματα.

- NPM (Node Package Manager): Το Node.js περιλαμβάνει το NPM, ένα εργαλείο εγκατάστασης πακέτων και διαχείρισης εξαρτήσεων που επιτρέπει την ευκολότερη και γρηγορότερη ανάπτυξη εφαρμογών.

Το Node.js παίζει ζωτικό ρόλο στην απόδοση και αξιοπιστία του AutoDBSCAN επιτρέποντας την ασύγχρονη επεξεργασία των αιτημάτων και την ταχεία επικοινωνία μεταξύ των διαφορετικών τμημάτων του συστήματος.

Καταρχάς, η δυνατότητα του Node.js για ασύγχρονη I/O επεξεργασία μειώνει σημαντικά την αναμονή κατά την εκτέλεση εργασιών που απαιτούν μεγάλο χρόνο, όπως η επεξεργασία μεγάλων συνόλων δεδομένων. Στην περίπτωση του AutoDBSCAN, αυτό μπορεί να σημαίνει ταχύτερη ανάλυση των δεδομένων και ταχύτερη παροχή αποτελεσμάτων στους χρήστες [22].

Επιπλέον, το Node.js υποστηρίζει τον απλοποιημένο χειρισμό σφαλμάτων, πράγμα που συμβάλλει στην αξιοπιστία της εφαρμογής. Ειδικά σε μια εφαρμογή όπως το AutoDBSCAN, όπου η επεξεργασία μεγάλων συνόλων δεδομένων και η αλληλεπίδραση με διάφορα συστήματα (όπως οι βάσεις δεδομένων και οι υπηρεσίες αυθεντικοποίησης) μπορεί να εγείρουν ζητήματα, η δυνατότητα του Node.js να διαχειριστεί και να αποκαταστήσει τα σφάλματα αυξάνει την αξιοπιστία του συστήματος [22].

Τέλος, η ευελιξία του Node.js, όπως η δυνατότητα ενσωμάτωσης διάφορων βιβλιοθηκών και εργαλείων, καθιστά πιο εύκολη την προσαρμογή του AutoDBSCAN σε μεταβαλλόμενες ανάγκες ή απαιτήσεις, παρέχοντας μια ρομποτική και επεκτάσιμη βάση για την εφαρμογή.

Συνολικά, το Node.js προσφέρει μια σταθερή, αξιόπιστη και αποδοτική βάση για την κατασκευή του AutoDBSCAN, διευκολύνοντας την ανάπτυξη και την επέκταση της εφαρμογής ενώ ταυτόχρονα διασφαλίζει την υψηλή απόδοση και αξιοπιστία.

Python

Η γλώσσα προγραμματισμού Python είναι μια δυναμικά πληκτρολογούμενη, ερμηνευμένη γλώσσα υψηλού επιπέδου που δημιουργήθηκε από τον Guido van Rossum και πρωτοεμφανίστηκε το 1991 [24]. Έχει επικεντρωθεί στην αναγνωσιμότητα και την ευελιξία, προσφέροντας μια πλούσια στάνταρ βιβλιοθήκη [25].

Η Python πήρε το όνομά της από την τηλεοπτική σειρά “Monty Python’s Flying Circus” και όχι από τον φίδι. Η ανάπτυξη της Python ξεκίνησε τα τέλη της δεκαετίας του ’80 ως μια διαδοχή της γλώσσας ABC [24].

Στη διάρκεια των επόμενων δεκαετιών, η Python έγινε μια από τις πιο δημοφιλείς γλώσσες προγραμματισμού στον κόσμο, χρησιμοποιούμενη σε πολλούς τομείς, όπως η επιστήμη δεδομένων, η ιστοσελίδα και η εφαρμογή ανάπτυξης, και τα εκπαιδευτικά συστήματα.

Η Python είναι γνωστή για την αναγνωσιμότητά της, με καθαρό συντακτικό κώδικα που τονίζει τη σαφήνεια και την οργάνωση. Η χρήση εσοχών αντί για συμβολισμούς όπως οι αγκύλες επιτρέπει τη δημιουργία καθαρών και καλά οργανωμένων μπλοκ κώδικα.

Είναι μια ερμηνευμένη γλώσσα, που σημαίνει πως ο κώδικας τρέχει γραμμή προς γραμμή, διευκολύνοντας τη διαδικασία αποσφαλμάτωσης και ανάπτυξης. Ως γλώσσα υψηλού επιπέδου, αφαιρεί τον προγραμματιστή από λεπτομέρειες όπως η διαχείριση της μνήμης, καθιστώντας την πιο προσιτή.

Ένα από τα κεντρικά χαρακτηριστικά της Python είναι η δυναμική πληκτρολόγηση, όπου ο τύπος της μεταβλητής προσδιορίζεται κατά την εκτέλεση, χωρίς την ανάγκη δήλωσης. Αυτό προσφέρει ευελιξία και επιταχύνει την ανάπτυξη.

Η φορητότητα είναι επίσης σημαντική, καθώς η Python λειτουργεί σε διάφορα λειτουργικά συστήματα, όπως Windows, macOS, και Linux. Η εκτεταμένη της βιβλιοθήκη προσφέρει μια πλούσια συλλογή από εργαλεία και δυνατότητες που καλύπτουν πολλούς τομείς της πληροφορικής.

Η Python υποστηρίζει πολλαπλά παραδείγματα προγραμματισμού, όπως τον διαδικαστικό, τον αντικειμενοστρεφή και τον λειτουργικό προγραμματισμό, προσφέροντας ευελιξία στους προγραμματιστές.

Η κοινότητα και η υποστήριξη που προσφέρει η Python είναι ακόμη δύο παράγοντες που την καθιστούν δημοφιλή. Υπάρχει μια ενεργή και αυξανόμενη κοινότητα προγραμματιστών που συνεισφέρει στη βελτίωση και την επέκταση της γλώσσας.

Τέλος, η αυτόματη διαχείριση της μνήμης μέσω της συλλογής σκουπιδιών και η επεκτασιμότητα με γλώσσες όπως η C και η C++, καθιστούν την Python μια ισχυρή και πρακτική γλώσσα προγραμματισμού για πολλές εφαρμογές και βιομηχανίες.

Η επιλογή της Python για το συγκεκριμένο project ήταν καταλυτική, κυρίως λόγω της βιβλιοθήκης scikit-learn. Η scikit-learn είναι μια ανοιχτού κώδικα βιβλιοθήκη που προσφέρει απλές και αποτελεσματικές εργαλειοθήκες για την εξόρυξη δεδομένων και την ανάλυση δεδομένων, καθιστώντας την ιδανική για εργασίες μηχανικής μάθησης. Η εύκολη ενσωμάτωση και η ευρεία υποστήριξη αλγορίθμων που προσφέρει η scikit-learn καθιστούν την Python μια προτιμώμενη επιλογή για το project, επιτρέποντας την αποτελεσματική ανάλυση και επεξεργασία των δεδομένων, μειώνοντας τόσο τον χρόνο ανάπτυξης όσο και την πολυπλοκότητα[26].

Express.js

Το Express.js είναι ένα ισχυρό, ευέλικτο και δημοφιλές πλαίσιο για την ανάπτυξη εφαρμογών διακομιστή σε Node.js. Έχει σχεδιαστεί για την απλοποίηση της διαδικασίας ανάπτυξης web εφαρμογών, παρέχοντας ένα πλήρως λειτουργικό πλαίσιο για τη διαχείριση των αιτημάτων και των απαντήσεων του HTTP, των δρομολογήσεων, των προβλέψεων και των μεσαίων λογισμικών[27]. Οι πληροφορίες που παρέχει το Express.js αναδεικνύουν τη δυνατότητά του

να ενισχύει την ανάπτυξη και την ευελιξία του AutoDBSCAN.

Πρώτον, το Express.js επιτρέπει την ευκολία δημιουργίας και διαχείρισης δρομολογήσεων, το οποίο είναι ζωτικής σημασίας για την επεξεργασία διάφορων αιτημάτων HTTP που λαμβάνει η εφαρμογή. Δεύτερον, το Express.js υποστηρίζει την οργάνωση της εφαρμογής με την χρήση "middleware". Τα middleware είναι διαμεσολαβητικές λειτουργίες που μπορούν να χρησιμοποιηθούν για την προεπεξεργασία των αιτημάτων πριν φτάσουν στις διαδρομές τους, για παράδειγμα, για τον έλεγχο των επικεφαλίδων αυθεντικοποίησης ή για την αντιμετώπιση των σφαλμάτων[27].

Τέλος, η ενσωμάτωση του Express.js με άλλες βιβλιοθήκες και εργαλεία της Node.js (όπως οι βάσεις δεδομένων και οι υπηρεσίες αυθεντικοποίησης) κάνει το AutoDBSCAN εξαιρετικά ευέλικτο και επεκτάσιμο, επιτρέποντας την αποτελεσματική και εύκολη προσαρμογή σε νέες απαιτήσεις ή ανάγκες.

JSON Web Tokens

Τα JSON Web Tokens (JWTs) είναι μια ανοικτής προδιαγραφής (RFC 7519) μέθοδος για την ασφαλή μεταφορά δεδομένων μεταξύ δύο μερών μέσω JSON αντικειμένων. Αυτό τα κάνει εξαιρετικά χρήσιμα σε σενάρια αυθεντικοποίησης και έκδοσης δικαιωμάτων, καθώς και για τη μεταφορά ασφαλών πληροφοριών μεταξύ διαφορετικών συστημάτων.

Ένα JWT αποτελείται από τρία μέρη: ένα κεφαλίδα, μια διεκδίκηση (ή φορτίο) και μια υπογραφή. Το κεφαλίδα κωδικοποιείται με Base64 και περιέχει πληροφορίες για τον τύπο του token και τον αλγόριθμο υπογραφής. Η διεκδίκηση είναι επίσης κωδικοποιημένη με Base64 και περιέχει τα ασφαλή δεδομένα (όπως οι πληροφορίες του χρήστη) που πρέπει να μεταφερθούν. Η υπογραφή δημιουργείται με την κωδικοποίηση του κεφαλίδα και της διεκδίκησης μαζί με ένα μυστικό κλειδί και εξασφαλίζει την ακεραιότητα του token .

Η χρήση JWTs στο AutoDBSCAN βοηθάει στην αύξηση της ασφάλειας και της αυθεντικοποίησης των χρηστών. Όταν ο χρήστης συνδέεται στην εφαρμογή, το σύστημα δημιουργεί ένα JWT με τις πληροφορίες του χρήστη και το στέλνει στον client. Σε κάθε επόμενο αίτημα, ο client θα πρέπει να συμπεριλάβει αυτό το token στην επικεφαλίδα του αιτήματος. Αυτό επιτρέπει στον server να ταυτοποιεί τον χρήστη και να επαληθεύει την ακεραιότητα των δεδομένων χωρίς την ανάγκη για μια επιπλέον ερώτηση στη βάση δεδομένων[28].

SWAGGER

Το Swagger είναι ένα σετ εργαλείων ανοιχτού λογισμικού που χρησιμοποιούνται για την ανάπτυξη, την τεκμηρίωση, και την δοκιμή RESTful Web APIs. Δημιουργήθηκε από την SmartBear Software και είναι γνωστό για την ικανότητά του να παρέχει ενημερωμένη και προσβάσιμη τεκμηρίωση, που επιτρέπει στους προγραμματιστές να κατανοήσουν και να δοκιμάσουν το API γρήγορα[29].

Το Swagger υποστηρίζει ένα αποτελεσματικό workflow ανάπτυξης, από το σχεδιασμό μέχρι την παραγωγή, και περιλαμβάνει μια σουίτα εργαλείων που καλύπτει διάφορες φάσεις της

ανάπτυξης. Το Swagger UI είναι ένα από τα δημοφιλέστερα εργαλεία του, προσφέροντας μια διαδραστική διεπαφή όπου οι χρήστες μπορούν να πειραματιστούν με τα API calls. Το Swagger Editor επιτρέπει την επεξεργασία των αρχείων τεκμηρίωσης, ενώ το Swagger Codegen μπορεί να παράγει κώδικα σε διάφορες γλώσσες προγραμματισμού[29].

Η τεκμηρίωση στο Swagger διαμορφώνεται σε μορφή YAML ή JSON και συμμορφώνεται με το OpenAPI Specification (OAS), το οποίο είναι ένα πρότυπο που καθορίζει τη διεπαφή του RESTful API. Αυτό καθιστά την ενσωμάτωση με διάφορες πλατφόρμες και γλώσσες προγραμματισμού πιο εύκολη και αυτοματοποιημένη, διευκολύνοντας τόσο την ανάπτυξη όσο και τη συντήρηση των APIs[29].

Η επιλογή της τεχνολογίας Swagger στο συγκεκριμένο project έγινε μετά από προσεκτική αξιολόγηση, λαμβάνοντας υπόψη πολλούς παράγοντες. Καταρχάς, η Swagger παρέχει εύκολη στην κατανόηση και αναπαραγωγή τεκμηρίωση για τα RESTful APIs, κάτι που επιταχύνει την ανάπτυξη και τη δοκιμή. Επιπλέον, παρέχει αυτοματοποιημένους ελέγχους, καθώς και ένα καλαίσθητο γραφικό περιβάλλον που διευκολύνει την αλληλεπίδραση με το API. Η υποστήριξη για διάφορες γλώσσες προγραμματισμού και η ενσωμάτωση με διάφορα εργαλεία ελέγχου κώδικα κάνει την Swagger μια ευέλικτη και ολοκληρωμένη λύση. Αυτές οι ιδιότητες ανταποκρίθηκαν άμεσα στις απαιτήσεις και τους στόχους του project, καθιστώντας την Swagger την καταλληλότερη επιλογή.

XAMPP

Το XAMPP είναι ένα δημοφιλές, δωρεάν και ανοιχτού κώδικα λογισμικό που λειτουργεί ως πακέτο ανάπτυξης εφαρμογών ιστού. Παρέχει έναν ολοκληρωμένο διακομιστή Apache, με την υποστήριξη PHP, MySQL και Perl, επιτρέποντας στους προγραμματιστές να δημιουργούν και να δοκιμάζουν ιστοσελίδες και εφαρμογές ιστού στο τοπικό τους σύστημα. Το phpMyAdmin, από την άλλη πλευρά, είναι ένα εργαλείο διαχείρισης βάσεων δεδομένων MySQL που λειτουργεί μέσω περιηγητή. Συχνά εγκαθίσταται μαζί με το XAMPP, και προσφέρει μια γραφική διεπαφή για τη διαχείριση και τη διαμόρφωση βάσεων δεδομένων. Η συνεργασία του XAMPP με το phpMyAdmin καθιστά την ανάπτυξη και τη διαχείριση ιστοσελίδων απλούστερη και πιο προσιτή, ακόμη και για όσους δεν έχουν εμπειρία στη διαχείριση βάσεων δεδομένων[30].

MySQL

Το MySQL είναι ένα ανοιχτού κώδικα σύστημα διαχείρισης βάσεων δεδομένων σχεσιακού τύπου (RDBMS). Χρησιμοποιεί τη γλώσσα ερωτημάτων SQL (Structured Query Language), την πλέον γνωστή γλώσσα για τη διαχείριση και την επεξεργασία δεδομένων σε σχεσιακές βάσεις δεδομένων.

Το MySQL αναπτύχθηκε και διανέμεται από την Oracle Corporation και είναι διαθέσιμο σε διάφορες εκδόσεις, μερικές από τις οποίες περιλαμβάνουν επιπρόσθετες λειτουργίες και υποστήριξη. Είναι δημοφιλές εργαλείο για την ανάπτυξη ιστοσελίδων και εφαρμογών λόγω της απλότητάς του, της αξιοπιστίας και της απόδοσης.

Το MySQL υποστηρίζει διάφορους τύπους δεδομένων, όπως κείμενο, αριθμούς και ημερομηνίες, και προσφέρει δυνατότητες όπως την εφαρμογή περιορισμών, τη χρήση ευρετηρίων για

βελτιωμένη απόδοση ερωτημάτων, και τη διαχείριση συναλλαγών για τη διασφάλιση της συνέπειας των δεδομένων.

Διαθέτει επίσης εργαλεία για τη διαχείριση και την ανάλυση της βάσης δεδομένων, όπως το MySQL Workbench, το οποίο προσφέρει γραφική διεπαφή για τη διαμόρφωση, τη διαχείριση και την ανάπτυξη των βάσεων δεδομένων. Είναι διαθέσιμο σε πολλά λειτουργικά συστήματα, συμπεριλαμβανομένων των Windows, Linux, και macOS.

3.2 Τεχνολογίες Client side

Η διεπαφή χρήστη, ή frontend, είναι το μέρος της εφαρμογής που βλέπουν και με το οποίο αλληλεπιδρούν οι χρήστες. Στο πλαίσιο της κατασκευής του AutoDBSCAN, δόθηκε ιδιαίτερη προσοχή στην επιλογή των τεχνολογιών που θα χρησιμοποιούνταν για την εμπρόσθια έκδοση, λαμβάνοντας υπόψη την ευχρηστία, την απόδοση, την ασφάλεια και την ευελιξία.

Διάφορες τεχνολογίες, όπως το React, το Next.js και η βιβλιοθήκη Axios, επιλέχθηκαν για τον σχεδιασμό και την υλοποίηση της εμπρόσθιας έκδοσης. Επίσης, η χρήση των JSON Web Tokens (JWTs) επιτρέπει την ασφαλή αυθεντικοποίηση των χρηστών.

Στις επόμενες υποενότητες, θα αναλύσουμε καθένα από αυτά τα εργαλεία και τις τεχνολογίες, εξηγώντας γιατί επιλέχθηκαν και πώς συνεισφέρουν στην κατασκευή του AutoDBSCAN.

React
Η React είναι μια δημοφιλής JavaScript βιβλιοθήκη για την κατασκευή διεπαφών χρήστη, που αναπτύχθηκε από το Facebook. Μέσα από τη δημιουργία “components”, που αντιπροσωπεύουν διάφορα στοιχεία της διεπαφής χρήστη, η React επιτρέπει την κατασκευή ευέλικτων και αποδοτικών web εφαρμογών[31].

Κύρια χαρακτηριστικά της React περιλαμβάνουν:

- Component-Based: Η React χρησιμοποιεί επαναχρησιμοποίηση components για να διευκολύνει την ανάπτυξη και την συντηρηση κώδικα. Κάθε component είναι αυτόνομο και έχει τη δυνατότητα να διαχειρίζεται τη δική του κατάσταση, κάνοντας τον κώδικα πιο καθαρό και ευανάγνωστο.
- Virtual DOM (Document Object Model): Η React χρησιμοποιεί την τεχνική του Virtual DOM για να βελτιστοποιήσει τις επιδόσεις της εφαρμογής. Αντί να ενημερώνει ολόκληρο το DOM κάθε φορά που γίνεται μια αλλαγή, η React δημιουργεί ένα Virtual DOM και ενημερώνει μόνο τα στοιχεία που χρειάζεται.
- Declarative Programming: Στην React, αναλαμβάνετε να περιγράψετε την εμφάνιση της διεπαφής χρήστη ανεξάρτητα από την κατάσταση της εφαρμογής, και η React ενημερώνει αυτόματα την διεπαφή χρήστη όταν αλλάζει η κατάσταση.

Σε συνδυασμό με άλλες βιβλιοθήκες, όπως το Redux για τη διαχείριση κατάστασης, η React είναι ιδιαίτερα ισχυρή και ευέλικτη, γεγονός που την καθιστά ιδανική για τη δημιουργία εμπρόσθιας έκδοσης για το AutoDBSCAN.

Next.js
Το Next.js είναι ένα ανοικτού κώδικα JavaScript framework που επιτρέπει την ανάπτυξη τόσο της πλευράς του πελάτη (client-side), όσο και της πλευράς του διακομιστή (server-side) ιστοσελίδων, με βάση την React. Δημιουργήθηκε από την Vercel και έχει κερδίσει μεγάλη δημοτικότητα λόγω των πλούσιων χαρακτηριστικών του και της αυξημένης απόδοσης.

Ακολουθούν μερικά από τα κύρια χαρακτηριστικά του Next.js:

- Υποστήριξη για Server Side Rendering (SSR) και Static Site Generation (SSG): Το Next.js δίνει την δυνατότητα στους προγραμματιστές να δημιουργήσουν ιστοσελίδες που υποστηρίζουν Server Side Rendering (SSR) ή Static Site Generation (SSG). Αυτό είναι εξαιρετικά χρήσιμο για την βελτίωση της απόδοσης και της SEO των ιστοσελίδων.
- File-based Routing: Το Next.js χρησιμοποιεί ένα απλό σύστημα δρομολόγησης βασισμένο στη δομή των αρχείων. Κάθε αρχείο στον κατάλογο “pages” αντιστοιχεί σε μια διαδρομή ιστοσελίδας.
- Hot Code Reloading: Όπως και η React, το Next.js υποστηρίζει το “hot reloading”, το οποίο σημαίνει ότι οι αλλαγές που κάνετε στον κώδικά σας εφαρμόζονται αυτόματα στον browser χωρίς να χρειάζεται να ανανεώσετε τη σελίδα.
- Built-in CSS and Sass Support, Image Optimization: Το Next.js προσφέρει ενσωματωμένη υποστήριξη για CSS και Sass, καθώς και βελτιστοποίηση εικόνων.

Συνολικά, το Next.js είναι ένα ισχυρό εργαλείο για τη δημιουργία υψηλής απόδοσης και ευέλικτων εφαρμογών frontend, παρέχοντας πολλά χαρακτηριστικά που ενισχύουν την ανάπτυξη και την απόδοση.

Axios
Το Axios είναι μια βιβλιοθήκη JavaScript που παρέχει μια υψηλού επιπέδου διεπαφή για την εκτέλεση HTTP requests, είτε στον πελάτη (frontend) είτε στον διακομιστή (backend). Είναι μια πολύ δημοφιλής επιλογή για προγραμματιστές JavaScript, ιδιαίτερα για εκείνους που εργάζονται με React ή Vue.js.

Μερικά από τα βασικά χαρακτηριστικά και πλεονεκτήματα του Axios περιλαμβάνουν:

- Promise-based API: Το Axios χρησιμοποιεί τον μηχανισμό promises της JavaScript, το οποίο προσφέρει μια καλύτερη μέθοδο για τη διαχείριση των ασύγχρονων λειτουργιών.
- Αυτόματη μετατροπή JSON: Οι αιτήσεις και οι αποκρίσεις JSON μετατρέπονται αυτόματα, γεγονός που εξοικονομεί χρόνο και κάνει τον κώδικα πιο καθαρό.
- Προστασία από XSRF (Cross-Site Request Forgery): Το Axios παρέχει μηχανισμούς για την ασφάλεια και την προστασία από επιθέσεις XSRF.

- Διαμόρφωση σε επίπεδο instance: Με το Axios, μπορείτε να διαμορφώσετε τις προεπιλεγμένες ρυθμίσεις για τις αιτήσεις σε επίπεδο instance, γεγονός που καθιστά ευκολότερη την εξατομίκευση.

Η χρήση του Axios ενισχύει την επικοινωνία μεταξύ frontend και backend στο AutoDBSCAN με διάφορους τρόπους. Πρώτον, με την χρήση promises, το Axios καθιστά απλούστερη την διαχείριση των ασύγχρονων αιτήσεων και αποκρίσεων. Δεύτερον, η αυτόματη μετατροπή σε JSON σημαίνει ότι τα δεδομένα είναι έτοιμα για χρήση χωρίς περαιτέρω επεξεργασία. Τέλος, το Axios διαχειρίζεται αυτόματα την προστασία CSRF, προσφέροντας μια επιπλέον στρώση ασφάλειας.

Postman

Το Postman είναι ένα δημοφιλές εργαλείο που χρησιμοποιείται κυρίως για την ανάπτυξη, δοκιμή και διαχείριση APIs (Διεπαφές Προγραμματισμού Εφαρμογών). Προσφέρει μια γραφική διεπαφή χρήστη που επιτρέπει στους προγραμματιστές να δημιουργούν, να στέλνουν και να λαμβάνουν HTTP αιτήματα και απαντήσεις, διευκολύνοντας τη διαδικασία δοκιμής και επικύρωσης των APIs. Το Postman καθιστά τη διαδικασία πειραματισμού με διάφορους τύπους αιτημάτων και παραμέτρων γρήγορη και απλή, ενώ παρέχει επίσης δυνατότητες για την αυτοματοποίηση των δοκιμών. Αυτό το εργαλείο είναι ιδιαίτερα χρήσιμο για ομάδες που εργάζονται σε περίπλοκα έργα, καθώς επιτρέπει την κοινή χρήση και την τεκμηρίωση των APIs με ευκολία[32].

JWT

Τα JSON Web Tokens (JWTs) είναι μια σημαντική τεχνολογία για την ασφάλεια και την αυθεντικοποίηση σε εφαρμογές web. Είναι μια ανοικτής προδιαγραφής (RFC 7519) μέθοδος για την ασφαλή μεταφορά δεδομένων μεταξύ δύο μερών μέσω μιας ψηφιακής υπογραφής. Ένα JWT είναι ένα κωδικοποιημένο JSON αντικείμενο που περιέχει πληροφορίες (ονομαζόμενες διεκδικήσεις ή claims) που επαληθεύουν την ταυτότητα του χρήστη. Στην περίπτωση του AutoDBSCAN, τα JWTs χρησιμοποιούνται για τη διασφάλιση των επικοινωνιών μεταξύ του client (frontend) και του server (backend). Όταν ένας χρήστης συνδέεται, ο server δημιουργεί ένα μοναδικό JWT για αυτόν τον χρήστη, το οποίο στη συνέχεια αποθηκεύεται σε ένα cookie στον browser του χρήστη. Αυτό το JWT χρησιμοποιείται σε όλες τις επόμενες αιτήσεις από τον client προς τον server, επιτρέποντας στον server να επαληθεύσει την ταυτότητα του χρήστη. Το cookie χρησιμοποιείται για την αποθήκευση του JWT, διασφαλίζοντας ότι ο server μπορεί να το ανακτήσει κατά τη διάρκεια κάθε αιτήματος. Αυτό προσφέρει πολλά πλεονεκτήματα, όπως η επέκταση της συνεδρίας χρήστη κατά τη διάρκεια διαφορετικών συνεδριών περιήγησης και η δυνατότητα για τον server να προσφέρει εξατομικευμένες υπηρεσίες βάσει των πληροφοριών του χρήστη.

Tailwind CSS

Το Tailwind CSS είναι ένα πλαίσιο CSS χαμηλού επιπέδου που επιτρέπει στους προγραμματιστές να δημιουργούν γρήγορα και εύκολα σχέδια χωρίς να πρέπει να γράψουν εξατομικευμένο CSS. Αντί να παρέχει έτοιμες κλάσεις για ορισμένα συγκεκριμένα στοιχεία σχεδίασης, το Tailwind παρέχει μια σειρά από βοηθητικές κλάσεις που μπορούν να συνδυαστούν για να δημιουργήσουν οποιοδήποτε σχεδιαστικό στοιχείο[33].

Για παράδειγμα, αντί να γράψετε τον δικό σας κώδικα CSS για να καθορίσετε το πλάτος, το ύψος, τα περιθώρια, τις αποστάσεις κ.λπ., μπορείτε να χρησιμοποιήσετε τις προκαθορισμένες κλάσεις του Tailwind για να επιτύχετε το ίδιο αποτέλεσμα. Αυτό καθιστά τη διαδικασία ανάπτυξης ταχύτερη και πιο συνεπή.
Το Tailwind επίσης παρέχει εργαλεία για την προσαρμογή των προεπιλεγμένων κλάσεων, καθώς και για τη δημιουργία δικών σας, γεγονός που το καθιστά ευέλικτο και κατάλληλο για ποικίλες ανάγκες σχεδίασης.
Με αυτόν τον τρόπο, το Tailwind CSS βοηθά στην επιτάχυνση της ανάπτυξης και παρέχει μια δομημένη βάση για τη δημιουργία αποτελεσματικών, αποκρίσιμων σχεδίων ιστοσελίδων και εφαρμογών.

3.3 Type Script

Η TypeScript είναι μια ανοιχτού κώδικα γλώσσα προγραμματισμού που αναπτύχθηκε από τη Microsoft και παρέχει την δυνατότητα στατικών τύπων στη JavaScript, επιτρέποντας έτσι στους προγραμματιστές να χρησιμοποιούν τυπολογικά ασφαλείς συναρτήσεις και μεταβλητές. Αυτό σημαίνει ότι ο τύπος μιας μεταβλητής γνωρίζεται κατά τη συγγραφή του κώδικα, επιτρέποντας στην TypeScript να ελέγχει την συμβατότητα των τύπων και να αναδεικνύει τυχόν λάθη τύπων πριν από την εκτέλεση του κώδικα.

Ενσωματώνει πρόσθετες λειτουργίες της JavaScript που δεν είναι ακόμη πλήρως υποστηριζόμενες σε όλα τα περιβάλλοντα της JavaScript, ενισχύοντας έτσι την ευελιξία και την ισχύ της γλώσσας. Αυτά τα χαρακτηριστικά περιλαμβάνουν διασαφηνίσεις τύπων, γενικά (generics), διανυσματικά (tuples), σύμβολα, decorators και πολλά άλλα.

Οι διασαφηνίσεις τύπων βοηθούν στην ανάπτυξη κώδικα που είναι λιγότερο επιρρεπής σε σφάλματα, ενώ οι γενικές τύπων και τα διανυσματικά επιτρέπουν τη δημιουργία ευέλικτων και επαναχρησιμοποιήσιμων δομών δεδομένων και συναρτήσεων. Τα σύμβολα και οι decorators μπορούν να χρησιμοποιηθούν για την προσθήκη μετα-πληροφοριών στα αντικείμενα και τις συναρτήσεις της JavaScript, καθιστώντας τη γλώσσα πιο παραγωγική και ευχάριστη για τους προγραμματιστές. Αυτά τα χαρακτηριστικά, μαζί με την υποστήριξη των τύπων, συμβάλλουν στη δημιουργία κώδικα που είναι πιο ευανάγνωστος, ευκολότερος στη διαχείριση και λιγότερο επιρρεπής σε σφάλματα, ενισχύοντας έτσι την παραγωγικότητα και την αξιοπιστία της διαδικασίας ανάπτυξης του λογισμικού.

Είναι επίσης συμβατή με όλες τις υπάρχουσες JavaScript βιβλιοθήκες και frameworks, γεγονός που σημαίνει ότι οι προγραμματιστές μπορούν να ενσωματώσουν TypeScript στα υπάρχοντα projects τους με ελάχιστη προσπάθεια.

Ως υπερσύνολο της JavaScript, προσφέρει τη δυνατότητα προκαθορισμού τύπων για μεταβλητές, παραμέτρους και τιμές επιστροφής συναρτήσεων. Αυτό, γνωστό ως ασφάλεια τύπων, μετατρέπει τα λάθη τύπων που συνήθως ανιχνεύονται κατά την εκτέλεση σε λάθη που ανιχνεύονται κατά την συγγραφή, δηλαδή στη φάση ανάπτυξης του λογισμικού.
Ας πούμε, για παράδειγμα, ότι έχετε μια συνάρτηση που πρέπει να λαμβάνει έναν αριθμό. Σε

JavaScript, αν τροφοδοτήσετε αυτήν τη συνάρτηση με μια συμβολοσειρά αντί για έναν αριθμό, θα εμφανιστεί ένα σφάλμα κατά την εκτέλεση. Ωστόσο, με την TypeScript, ο προγραμματιστής θα λάβει μια προειδοποίηση για το λάθος κατά την συγγραφή του κώδικα, προτού ο κώδικας εκτελεστεί.
Ανιχνεύει επίσης περιπτώσεις στις οποίες μια μεταβλητή πρόκειται να χρησιμοποιηθεί χωρίς να έχει αρχικοποιηθεί, ή όταν μια παράμετρος που αναμένεται να είναι παρούσα λείπει. Αυτό αποτρέπει την εκτέλεση κώδικα που είναι πιθανό να προκαλέσει σφάλματα στην εκτέλεση. Με αυτόν τον τρόπο, βοήθησε στην αποφυγή λαθών και bugs στον κώδικα του AutoDBSCAN, επιτρέποντας την ανίχνευση και τη διόρθωση των πιθανών σφαλμάτων πριν από την εκτέλεση του κώδικα.

Συμβάλλει σημαντικά στην αύξηση της παραγωγικότητας και της διατηρησιμότητας του κώδικα. Οι λεπτομερείς τύποι που παρέχει καθιστούν τον κώδικα πιο αυτο-εξηγηματικό και κατανοητό. Οι προγραμματιστές μπορούν να δουν αμέσως τι τύπος δεδομένων αναμένεται για κάθε παράμετρο ή τι επιστρέφει μια συνάρτηση, χωρίς να χρειάζεται να κατανοήσουν όλη τη λειτουργικότητα του κώδικα. Αυτό αυξάνει την παραγωγικότητα κατά την ανάπτυξη του λογισμικού, καθώς οι προγραμματιστές μπορούν να γράψουν και να τροποποιήσουν τον κώδικα πιο γρήγορα και με μεγαλύτερη ακρίβεια, ελαχιστοποιώντας την πιθανότητα σφαλμάτων.
Επιπλέον, η δυνατότητα παραγωγής καθαρού και καλά δομημένου κώδικα καθιστά το project πιο διαχειρίσιμο και διατηρήσιμο. Το σύνολο των διακριτών τύπων και η ασφάλεια τύπων της TypeScript σημαίνει ότι ο κώδικας είναι πιο προβλέψιμος και λιγότερο ευάλωτος σε σφάλματα. Το αποτέλεσμα είναι ότι οι προγραμματιστές μπορούν να επαναχρησιμοποιήσουν τον κώδικα, να επεκτείνουν τη λειτουργικότητα ή να διορθώνουν προβλήματα με μικρότερη προσπάθεια και χρόνο. Αυτό αυξάνει την επεκτασιμότητα και την διατηρησιμότητα του λογισμικού.

Μπορεί να ενσωματωθεί αποτελεσματικά με πολλές δημοφιλείς τεχνολογίες ανάπτυξης λογισμικού, επιτρέποντας στους προγραμματιστές να αποκομίζουν τα πλεονεκτήματα της ασφάλειας τύπων και των αυξημένων δυνατοτήτων που παρέχει.
Στην περίπτωση του AutoDBSCAN μας, η TypeScript ενσωματώθηκε άψογα με το Node.js, το React και το Next.js.
Στο backend, επέκτεινε την JavaScript που τρέχει στο Node.js, προσθέτοντας τύπους και διευκολύνοντας την ανίχνευση λαθών κατά τη διάρκεια της ανάπτυξης. Αυτό οδήγησε σε μια πιο ασφαλή, αξιόπιστη και διαχειρίσιμη εφαρμογή backend.
Στο frontend, η χρησιμοποιήθηκε μαζί με τη βιβλιοθήκη React για τη δημιουργία σύνθετων διεπαφών χρήστη. Τα ισχυρά χαρακτηριστικά τύπων της διευκόλυναν τη διαχείριση της κατάστασης της εφαρμογής και των props, ενώ παράλληλα βελτίωσαν την παραγωγικότητα και την ασφάλεια του κώδικα.Επίσης ενσωματώθηκε με το Next.js, προσφέροντας αυξημένες δυνατότητες στην εφαρμογή μας, όπως αυτόματη διαχείριση routes, server-side rendering και static site generation, ενώ παράλληλα παρέχει την ασφάλεια των τύπων και τις αυξημένες δυνατότητες της TypeScript.

Κεφάλαιο 4

Σχεδίαση και Υλοποίηση του AutoDBSCAN

4.1 Λειτουργικές απαιτήσεις

Μια λειτουργική απαίτηση αποτελεί μια δήλωση των υπηρεσιών ή των δυνατοτήτων που πρέπει να προσφέρει ένα σύστημα λογισμικού. Αυτή η απαίτηση περιλαμβάνει την ανάλυση και τον προσδιορισμό των διαδικασιών εισόδου στο σύστημα, της συμπεριφοράς που πρέπει να έχει το σύστημα απέναντι σε αυτές τις εισόδους και των αντίστοιχων εξόδων που προκύπτουν από αυτό. Με άλλα λόγια, περιγράφει τι πρέπει να κάνει το λογισμικό, αντί για το πώς πρέπει να το κάνει.

Οι λειτουργικές απαιτήσεις του AutoDBSCAN είναι:

- Εγγραφή νέου χρήστη
  Ένας χρήστης θα μπορεί να κάνει εγγραφή στο σύστημα μέσω μιας φόρμας η οποία θα αποτελείτε από τέσσερα πεδία: Όνομα, email, Κωδικός πρόσβασης.

- Σύνδεση χρήστη στο σύστημα
  Ο χρήστης με βάση τα στοιχεία που έκανε εγγραφη, θα εχει την δυνατότητα να αποθηκεύει τα δικά του σύνολα δεδομένων για μελλοντική χρήση

- Επεξεργασία προσωπικών στοιχείων και κωδικού πρόσβασης
  Ο χρήστης θα μπορεί να τροποποιήσει τα δικά του προσωπικά στοιχεία, όπως το όνομα και τον κωδικό πρόσβασης.

- Διαγραφή Λογαριασμού
  Ο χρήστης θα έχει τη δυνατότητα να διαγράψει τον λογαριασμό του

- Ανάκτηση κωδικού πρόσβασης
  Ο χρήστης θα μπορεί να αλλάξει τον κωδικό πρόσβασής του σε περίπτωση που τον ξεχάσει, διαδικασία που θα πραγματοποιηθεί μέσω επιβεβαίωσης μέσω email.

- Αρχική σελίδα
  Μια αρχική σελίδα που θα περιγράφει την εφαρμογή

- Σελίδα DBSCAN
  Αυτη θα είναι η σελίδα που οι χρήστες θα μπορούν να εκτελούν τις υπηρεσίες που προσφέρουμε σχετικά με τον αλγόριθμο DBSCAN. Η σελίδα θα είναι προσβάσιμη απο όλους τους χρήστες. Οι επισκέπτες θα μπορούν να χρησιμοποιήσουν τις υπηρεσίες μονο για τα public σύνολα δεδομένων ενω οι εγγεγραμένοι χρήστες θα μπορούν να τις εφαρμόζουν και στα δικά τους.

- Σελίδα Profile
  Αυτή η σελίδα θα προσφέρει την δυνατότητα στον χρήστη να επεξεργαστεί τα στοιχεία του.

- Ανέβασμα αρχείου
  Ο χρήστης θα μπορεί να ανεβάσει ενα σύνολο δεδομένων. Αποδεκτοί τύποι ειναι csv. Το αρχείο μπορεί να είναι είτε ιδιωτικό είτε δημοσιο ανάλογα την επιλογή του χρήστη και τα δικαιώματα που έχει.

- Ανάγνωση αρχείου
  Ο χρήστης θα έχει την δυνατότητα να βλέπει το σύνολο δεδομένων του. Η ανάγνωση του αρχείου θα πρέπει αν γίνεται τμηματικά ώστε για τα μεγάλα σύνολα να μην αποστέλεται ολόκληρο το αρχείο

- Διαγραφή αρχείου
  Ο χρήστης θα έχει την δυνατότητα να διαγράψει κάποιο σύνολο δεδομένων δημόσιο ή ιδιωτικό ανάλογα τα δικαιώματα του.

- Μέθοδος προσδιορισμού eps
  Ο χρήστης θα μπορεί να επιλέξει ενα αριθμό K-κοντινότερων γειτόνων. Θα του επιστραφούν προτάσεις για το eps για το K που επέλεξε καθως και για τα επόμενα 5 μαζι με τα αντίστοιχα γραφηματα

- Συσταδοποίηση DBSCAN
  Ο χρήστης θα μπορει να εφαρμόσει συσταδοποίηση με τον DBSCAN δίνοντας τιμες για το MinPts και eps. Θα του επιστραφεί ενας πίνακας τμηματικα με τα δεδομένα του αρχείου και μια εξτρα στήλη για την συστάδα που έχουν ενταχθεί. Ο χτηστη θα έχει την δυνατότητα να κατεβασει το αρχείο ή να δεί το γράφημα των συστάδων

- Public API
  Θα πρέπει να παρέχεται ελεύθερη πρόσβαση σε ένα WEB API για την εκτέλεση όλων των λειτουργιών μέσω αιτημάτων HTTP

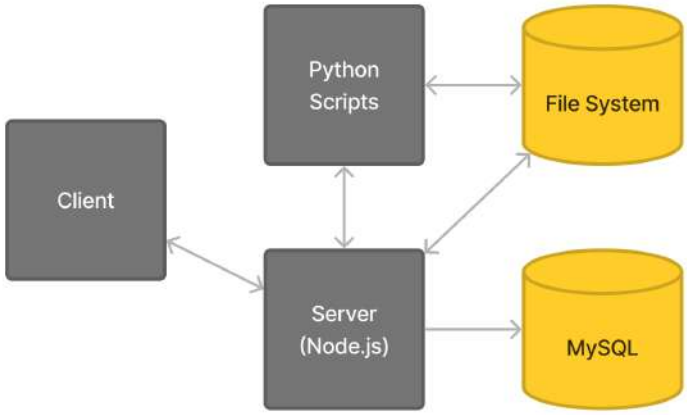
- Διαχείρηση χρηστών
  Ο διαχειρηστής της σελίδας θα μπορει να διαχειρίζεται το ποιος χρήστης έχει πρόσβαση στην διαχείρηση των δημοσίων συνόλων δεδομένων

- Αξιολόγηση Εμπειρίας Χρήσης
  Ο διαχειρηστής της εφάρμογης θα πρέπει να λαμβάνει ανατροφοδότηση από τους χρήστες μέσω του System Usability Scale (SUS), ώστε να βελτιώνεται συνεχώς η εμπειρία χρήσης της εφαρμογής.

4.2 Αρχιτεκτονική του AutoDBSCAN

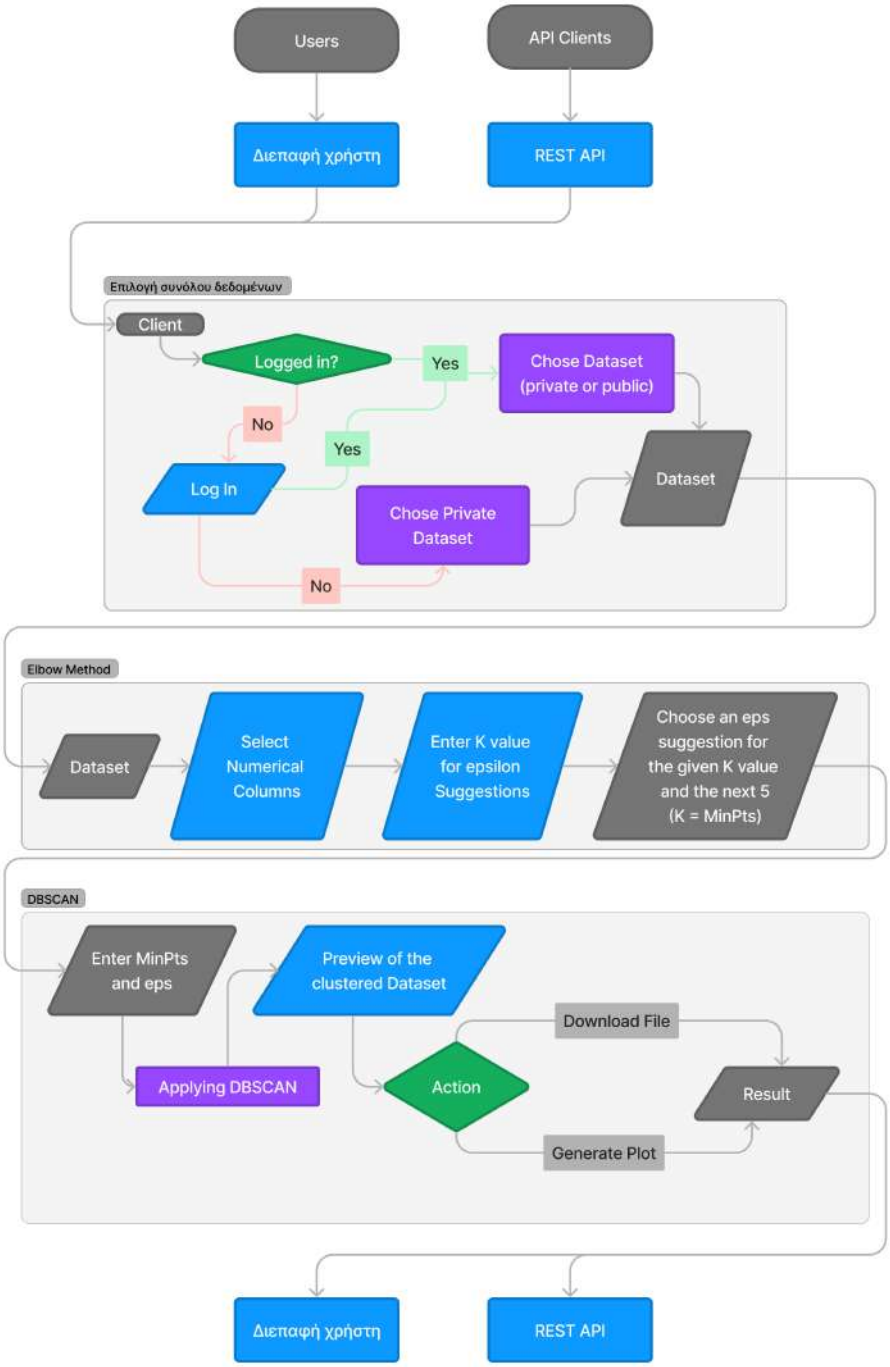
Η εφαρμογή έχει ως σκοπό την ανάπτυξη διαδικτυακής πλατφόρμας, όπου ο κάθε χρήστης θα μπορεί να ανεβάζει σύνολα δεδομένων για συσταδοποίηση. Η εφαρμογή θα κατασκευάζει το γράφημα της μεθόδου του K-dist graph, προτείνοντας στον χρήστη πιθανή τιμή για την παράμετρο εύρους (eps) και το ελάχιστο αριθμό δειγμάτων (MinPts). Στη συνέχεια, ο χρήστης θα μπορεί να εφαρμόσει τον αλγόριθμο DBSCAN δίνοντας τις προτεινόμενες παραμέτρους. Η ίδια διαδικασία θα είναι δυνατόν να γίνει μέσω ενός REST API.

Server: Όπως φαίνεται στο σχήμα 4.1 ο server διαχειρίζεται τα API calls για ολες τις υπηρεσιες που προαναφέρθηκαν. Για τις υπηρεσίες της αυθεντικοποίησης αποθηκεύει τα απαραίτητα δεδομένα στην SQL βαση. Για την αποθήκευση των συνόλων δεδομένων είτε για προσωρινή χρήση, είτε για μόνιμη αποθήκευση χρησιμοποιέι το file system ενώ για την επεξεργασία τρέχει κάποια custom scripts σε python που χρησιμοποιούν τις βιβλιοθήκες της Scikit Learn.



Σχήμα 4.1: Διάγραμμα αρχιτεκτονικής της εφαρμογής

Client: Ο χρήστης πρέπει να συνδεθεί με τα προσωπικά του στοιχεία (email, κωδικός πρόσβασης), και εάν δεν έχει λογαριασμό, πρέπει να εγγραφεί. Αφού συνδεθεί, θα έχει τη δυνατότητα να ανεβάσει σύνολα δεδομένων στον διακομιστή μέσω της διεπαφής. Στη συνέχεια, μπορεί να επιλέξει ένα από τα διαθέσιμα σύνολα δεδομένων και να τρέξει τη μέθοδο προσιορισμού eps για να λάβει την προτεινόμενη τιμή της παραμέτρου εύρους και του ελάχιστου αριθμού δειγμάτων για τον αλγόριθμο DBSCAN. Εν συνεχεία, μπορεί να τρέξει τη συσταδοποίηση με τις επιλεγμένες παραμέτρους. Η διεπαφή θα εμφανίζει τα αποτελέσματα της συσταδοποίησης και θα παρέχει στον χρήστη την επιλογή να τα κατεβάσει στον υπολογιστή του. Η λογική αυτή περιγράφεται στο σχήμα 4.2.



Σχήμα 4.2: Διάγραμμα Ροής

Δημόσια σύνολα δεδομένων και χρήστες
Στην εφαρμογή, τα δημόσια σύνολα δεδομένων είναι προσβάσιμα από όλους τους εγγεγραμμένους χρήστες και μπορούν να τα χρησιμοποιήσουν ελεύθερα. Ωστόσο, για λόγους ασφαλείας, η δυνατότητα ανεβάσματος δημόσιων συνόλων δεδομένων περιορίζεται σε όσους έχουν λάβει άδεια από έναν διαχειριστή (admin). Οι χρήστες έχουν επίσης τη δυνατότητα να ανεβάζουν προσωπικά σύνολα δεδομένων, τα οποία είναι ορατά και χρησιμοποιήσιμα μόνο από εκείνους, ενώ τα δημόσια σύνολα δεδομένων είναι ορατά σε όλους.

Τα δημόσια σύνολα δεδομένων είναι ιδιαίτερα χρήσιμα για εκπαιδευτικούς και καθηγητές που πρέπει να μοιράζονται δεδομένα με τους μαθητές ή τους φοιτητές τους για διδακτικούς και πειραματικούς σκοπούς.
Η διαχείριση της πρόσβασης στα σύνολα δεδομένων γίνεται μέσω της κατηγοριοποίησης των χρηστών ανάλογα με το επίπεδο προσβασιμότητάς τους, και οι ρόλοι που καθορίζονται στην εφαρμογή έχουν ως εξής:

- Μη εγγεγραμμένος
  Ο τύπος χρήστη που μπορεί να χρησιμοποιήσει μονο δημόσια σύνολα δεδομένων για τον αλγόριθμο DBSCAN

- Εγγεγραμμένος
  Ο εγγεγραμμένος χρήστης μπορεί να πραγματοποιήσει συσταδοποίηση είτε από τη διεπαφή ιστού είτε μέσω της υπηρεσίας REST API. Ο μόνος περιορισμός είναι το ανέβασμα δημόσιου συνόλου δεδομένων.

- Δημιουργός δημόσιου συνόλου δεδομένων
  Αυτός ο χρήστης έχει τα δικαιώματα ενός εγγεγραμμένου χρήστη συν τη δυνατότητα δημιουργίας δημόσιων συνόλων δεδομένων. Ένας εγγεγραμμένος χρήστης πρέπει να αναβαθμιστεί σε δημιουργό δημόσιου συνόλου δεδομένων από κάποιον διαχειριστή ώστε να ανεβάσει δημόσιο σύνολο δεδομένων.

- Διαχειριστής
  Έχει πρόσβαση στη βαση μέσω του phpMyAdmin η οποιουδήποτε άλλου client της MySQL. Η ανάθεση του ρόλου για την δημιουργία δημοσίων συνόλων δεδομένων γίνεται μόνο μέσω της βάσης.

4.3 Υλοποίηση του Server

Database Για τη διαχείριση των χρηστών χρειάστηκε να δημιουργήσουμε μια σχεσιακή βάση δεδομένων. Η MySQL είναι ένα από τα πιο δημοφιλή συστήματα διαχείρισης σχεσιακών βάσεων δεδομένων (RDBMS) που είναι ανοικτού κώδικα και προσφέρει υψηλή απόδοση, αξιοπιστία και ευελιξία.
Οι πίνακες της βάσης είναι ο *users* και ο *resetPassTokens*. Ο πίνακας *users* (Σχήμα 4.3) αποτελείται από τα εξής πεδία:

- Το πεδίο name είναι το ονοματεπώνυμο του χρήστη.
- Το πεδίο email είναι η διεύθυνση ηλεκτρονικού ταχυδρομείου με την οποία κάνει εγγραφή ο χρή- στης. Επίσης είναι το κύριο κλειδί του πίνακα.
- Το πεδίο password είναι ο κωδικός του χρήστη με τον οποίο κάνει εγγραφή και χρειάζεται για την είσοδο του στην εφαρμογή. Ο κωδικός είναι αποθηκευμένος σε hashed μορφή με τον αλ- γόριθμο Blowfish για λόγους ασφαλείας. Το Bcrypt είναι ένας αλγόριθμος κατακερματισμού κωδικών πρόσβασης που σχεδιάστηκε από τους Niels Provos και David Mazières βασισμένος στον κρυπτογραφικό αλγόριθμο Blowfish. Το όνομα “bcrypt” αποτελείται από δύο μέρη: το b και το crypt, όπου το b στέκει για Blowfish και το crypt είναι το όνομα της συνάρτησης κατακερματισμού που χρησιμοποιείται από το σύστημα κωδικών του Unix[34].
- Το πεδίο apiKey χρησιμοποιείται για να διμιουργηθεί ενας μοδανικός φάκελος για κάθε χρήστη ώστε να μπορεί να αποθηκεύει τα δικά του σύνολα δεδομένων. Δημιουργείται με τον συνδυασμό δύο τιμών που μετατρέπονται σε συμβολοσειρές και συνενώνονται. Το Date.now().toString() μετατρέπει τον αριθμό του τρέχοντος χρόνου σε χιλιοστά του δευτερολέπτου από την 1η Ιανουαρίου 1970, 00:00:00 UTC. σε συμβολοσειρά και συνεννόνεται με ενα τυχαίο αριθμο χρησιμοποιόντας το Math.floor(Math.random() * 1000000).
- Το πεδίο publicAccess παίρνει τις τιμές 0 και 1 και χρησιμοποιείται για να επιβεβαιώσουμε αν ο χρήστης έχει άδεια να ανεβάσει δημόσιο σύνολο δεδομένων.

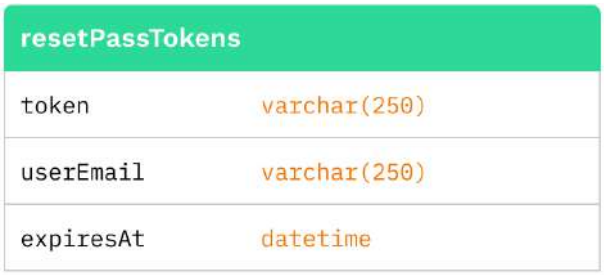
Σχήμα 4.3: Πίνακας Χρηστών

Ο πίνακας *resetPassTokens* (Σχήμα 4.4) αποθηκεύει τα tokens των χρηστών που κάνουν αίτημα ανάκτησης του κωδικού πρόσβασης. Αποτελείται απο τα εξής πεδία:

- email
  Το πεδίο email είναι η διεύθυνση ηλεκτρονικού ταχυδρομείου με την οποία κάνει εγγραφή ο χρή- στης. Επίσης είναι το κύριο κλειδί του πίνακα.

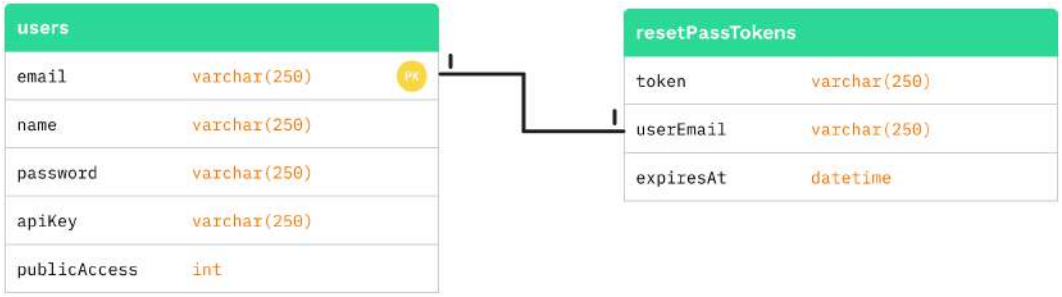
- token
  Το token ειναι ένα τυχαίο string που χρησιμοποιείται για την επαλήθευση της επαναφοράς του κωδικού. Διμιουργείται απο την βιλβιοθήκη crypto.randomBytes. Καλώντας την crypto.randomBytes(size), δημιουργείται ένα buffer τυχαίων byte. Μετατρέποντάς το σε συμβολοσειρά (π.χ., 'hex' ή 'base64'). Στην προκειμένη περίπτωση χρησιμοποιείται το 'hex' για τη μετατροπή.

- expiresAt
  Το πεδίο expiresAt υπάρχει για να δίνει μια ημερομηνία λήξης στο token για λόγους ασφαλείας. Του δίνουμε σαν ημερομηνία την επόμενη ώρα απο την ώρα του αιτήματος σε ms.



Σχήμα 4.4: Πίνακας Reset Password Tokens

Ένα διάγραμμα σχέσης οντοτήτων είναι ένα διάγραμμα που αναπαριστά σχέσεις μεταξύ οντοτήτων σε μια βάση δεδομένων. Είναι κοινώς γνωστό ως διάγραμμα ER. Ένα διάγραμμα ER στο DBMS παίζει σημαντικό ρόλο στο σχεδιασμό της βάσης δεδομένων[35]. Το διάγραμμα ER για την εφαρμογή AutoDBSCAN φαίνεται στο σχήμα 4.5.



Σχήμα 4.5: Διάγραμμα ER εφαρμογής

Rest API
Στον παρακάτων πινακα βλέπουμε μια λίστα με όλα τα διαθέσιμα endpoints της εφαρμογής.

Πίνακας 4.1: Κατάλογος Endpoints

Κατηγορία	**Περιγραφή**
Authentication	
POST	`/api/login` - Σύνδεση Χρήστη
POST	`/api/register` - Εγγραφή Χρήστη
POST	`/updateUserInfo` - Ενημέρωση Πληροφοριών Χρήστη
GET	`/fetchUserInfo` - Ανάκτηση Πληροφοριών Χρήστη
POST	`/forgotPassword` - Αίτηση Επαναφοράς Κωδικού
DELETE	`/deleteUser` - Διαγραφή Χρήστη
POST	`/resetPassword` - Επαναφορά Κωδικού Χρήστη
POST	`/changePassword` - Αλλαγή Κωδικού Χρήστη
Dataset Manipulation	
GET	`/api/fetchPublicDatasets` - Ανάκτηση Δημόσιων Dataset
GET	`/api/fetchPublicDataset` - Ανάκτηση Κομματιού Δημόσιου Dataset
GET	`/api/fetchPrivateDatasets` - Ανάκτηση Ιδιωτικών Datasets για Συγκεκριμένο Χρήστη
GET	`/api/fetchPrivateDataset` - Ανάκτηση Κομματιού Ιδιωτικού Dataset
DELETE	`/api/datasets/deleteDataset` - Διαγραφή Dataset
GET	`/api/datasets/downloadDataset` - Λήψη Dataset
POST	`/api/datasets/uploadDataset` - Μεταφόρτωση Dataset
Clustering Analysis Endpoints	
GET	`/api/findEpsilonAsGuest` - Υπολογισμός Εψιλον
GET	`/api/applyDBSCAN` - Εφαρμογή DBSCAN
GET	`/api/fetchParallelPlot` - Ανάκτηση Parallel Plot
Files Manipulation	
DELETE	`/api/deleteTempFiles` - Διαγραφή Προσωρινών Αρχείων
GET	`/api/fetchPlotImage` - Ανάκτηση Εικόνας Διάγραμματος
GET	`/api/downloadPlotImage` - Λήψη Εικόνας Διάγραμματος

Το REST API είναι ζωτικής σημασίας για την εφαρμογή, καθώς μπορεί να κληθεί είτε από το front-end της εφαρμογής είτε από άλλους προγραμματιστές για τις δικές τους ανάγκες, επιτρέποντας την εκτέλεση όλων των λειτουργιών που έχουν περιγραφεί σε προηγούμενες ενότητες. Γενικά, για να καλέσουμε μια λειτουργία του REST API, κάνουμε αίτημα στο αντίστοιχο endpoint, το οποίο μας επιστρέφει τα αποτελέσματα σε μορφή JSON. Ένα endpoint είναι ένα URI (Uniform Resource Identifier) που καλεί το αντίστοιχο script στον server.

Το REST API της εφαρμογής υλοποιήθηκε με την γλώσσα προγραμματισμού Node.js και χρησιμοποιεί το framework Express. Κάθε φορά που γίνεται κλήση σε ένα endpoint, το αντίστοιχο script σε Node.js εκτελείται. Πρέπει να περάσουμε κάποιες παραμέτρους, όπως για παράδειγμα παραμέτρους που απαιτούνται για την είσοδο στο script, και αυτές μπορεί να επηρεάσουν τον τρόπο λειτουργίας του. Οι παράμετροι περνιούνται ανάλογα με τον τύπο της μεθόδου του endpoint: για παράδειγμα, αν το endpoint είναι τύπου GET, οι παράμετροι περνιούνται μέσω του URI, ενώ αν το endpoint είναι τύπου POST, DELETE, PUT κλπ., οι παράμετροι περνιούνται μέσω του "body" του αιτήματος.

Για να εξήσουμε καλύτερα την διαδικασία λειτουργίας ενός endpoint ας αναλύσουμε το */fetchPrivateDataset* το οποίο καλύπτει ενα ευρύ φάσμα μεθόδων που χρησιμοποιούνται. Ξεκινάμε δημιουργώντας έναν router χρησιμοποιώντας τη μέθοδο express.Router()

```
import {Router} from "express";

const router = Router();
router.get(
    "/fetchPrivateDataset",
    (req, res, next) => {
        // Εδώ εκτελείται η λογική του middleware
        next() // Προχωράμε στο επόμενο middleware ή endpoint
    },
    async (req, res) => {
    //Εδώ εκτελείται η λογική του endpoint
    }
)
```

Στην περίπτωσή μας, ο middleware χρησιμοποιείται για τη διαδικασία της αυθεντικοποίησης του χρήστη. Αναλαμβάνει να εξετάσει τον κεφαλίδα Authorization, επαληθεύει το token και ανακτά το email του χρήστη, το οποίο στη συνέχεια περνά στη λογική του endpoint, όπως φαίνεται στο προηγούμενο παράδειγμα κώδικα. Στο Σχήμα 4.6 βλέπουμε τον κώδικα του middleware.

```
import jwt, { TokenExpiredError } from "jsonwebtoken";
import "dotenv/config";

const auth = (req: any, res: any, next: any) => {
  const authHeader = req.headers.authorization;
  if (!authHeader || !authHeader.startsWith("Bearer")) {
    return res.status(403).send("A token is required for authentication");
  }
  try {
    jwt.verify(
      authHeader.split(" ")[1],
      process.env.JWT_KEY ?? "jwt_key",
      function (err: any, decoded: any) {
        if (decoded) {
          req.body.decoded = decoded;
        }
      }
    );
  } catch (err) {
    if (err instanceof TokenExpiredError) {
      return res.status(401).send("jwt expired");
    } else {
      return res.status(401).send("Invalid Token");
    }
  }
  return next();
};

export default auth;
```

Σχήμα 4.6: Κώδικας auth middleware

Στο Σχήμα 4.7 φαίνονται οι παράμετροι που έρχονται απο το query του url οπως (req.query.filename) και στη συνέχεια βλέπουμε το email του user που έρχεται σαν body param απο τον middleware (req.body.decoded.email)

```
export const fetchPrivateDataset: RequestHandler = async (req, res) => {
  const filename = req.query.filename as string;
  const chunkIndex = parseInt(req.query.chunkIndex as string, 10) || 0; // Get the chunk index from the query parameter
  const chunkSize = 3; // Number of pages per chunk
  const pageSize = parseInt(req.query.pageSize as string, 10) || 10; // Get the page size from the query parameter

  if (req.body.decoded && req.body.decoded.email) {
    const user = await User.findByPk(req.body.decoded.email);
    if (user) {
      const filePath = path.join(
        __dirname,
        "..",
        "..",
        `/datasets/private/${user.apiKey}`,
        filename
      );
```

Σχήμα 4.7: Κώδικας παραμέτρων fetchPrivateDataset

Στο παρόν endpoint αξίζει να αναφερθεί ο κώδικας για τη τμηματοποίηση του συνόλου δε-

δομένων (Σχήμα 4.8). Η τμηματοποίηση γίνεται σε chunks. Στη δική μας περίπτωση το chunk (σύνολο σελίδων που επιστρέφεται) είναι το 3. Για την τμηματοποίηση ο client χρειάζεται να στείλει το chunkIndex (ο δείκτης της τριάδας σελίδων των οποίων τα δεδομένα επιθυμεί να δει) και το pageSize (απο πόσες γραμμές αποτελείται η σελίδα).

```
fileStream
  .on("data", (chunk) => {
    rowData.push(chunk);
  })
  .on("end", () => {
    const data = rowData.join("").split("\n");
    const header = data[0].split(",");

    const content = data.slice(1);

    const first1000Rows = content.slice(1, 1000);
    const numericalHeaders = header.filter((_, columnIndex) => {
      return first1000Rows.every(
        (row) => !isNaN(Number(row.split(",")[columnIndex]))
      );
    });
    const totalRows = content.length;
    const totalPages = Math.ceil(totalRows / pageSize);

    const startIndex = chunkIndex * chunkSize * pageSize;
    const endIndex = Math.min(
      startIndex + chunkSize * pageSize,
      totalRows
    );

    const paginatedContent = content.slice(startIndex, endIndex);

    const paginatedData = {
      header,
      numericalHeaders,
      content: paginatedContent,
      totalPages,
      totalChunks: Math.ceil(totalPages / chunkSize),
    };

    res.json(paginatedData);
  })
```

Σχήμα 4.8: Κώδικας pagination

Προσδιορισμός eps και minPts method

Όπως αναφέραμε στα προηγούμενα κεφάλαια για να εκτελεστεί ο αλγόριθμος DBSCAN σαν απαραίτητη προυπόθεση έχουμε τον καθορισμό των παραμέτρων eps και minPts. Η μέθοδος του προσδιορισμού eps και minPts υπάρχει για να δώσει μια πρόταση στον χρήστη για την τιμή eps με βάση την τιμη K (minPts). Ο χρήστης θα πρέπει να δώσει την τιμή K μετά απο μελέτη του συνόλου δεδομένων.

```
export const findEpsilon: RequestHandler = (req, res) => {
  const tempImageFilename = `temp_${Date.now()}-${Math.random()}.png`;
  const dataset_name = req.query.dataset_name as string;
  const k = req.query.k as string;
  const columns = req.query.columns as string;

  const tempImageFilePath = path.join(tempFolder, tempImageFilename);
  const args: string[] = [
    path.join(csvFolder, dataset_name),
    tempImageFilePath,
    k,
    columns,
  ];
```

Σχήμα 4.9: Κώδικας μεταβλητών της μεθόδου findEpsilon

Όπως φαίνεται στο Σχήμα 4.9, αρχικά δημιουργείται μια μεταβλητή tempImageFilename που περιέχει ένα μοναδικό όνομα αρχείου εικόνας, το οποίο περιλαμβάνει την τρέχουσα χρονική σήμανση και ένα τυχαίο αριθμητικό τμήμα. Ανακτούνται παράμετροι από το αίτημα HTTP, όπως το dataset_name, το k και το columns, με τη χρήση της req.query. Έπειτα δημιουργείται η μεταβλητή tempImageFilePath, η οποία αντιπροσωπεύει την πλήρη διαδρομή του προσωρινού αρχείου εικόνας. Τέλος δημιουργείται ένας πίνακας args που περιέχει τις παραμέτρους που θα περαστούν σε μια άλλη λειτουργία ή εργαλείο. Ο πίνακας περιλαμβάνει τη διαδρομή του αρχείου CSV, τη διαδρομή του προσωρινού αρχείου εικόνας, την παράμετρο k και την παράμετρο columns.

Στο επόμενο σχημα 4.10 περιγράφεται η εκτέλεση του python script. Αρχικά, χρησιμοποιείται η μέθοδος spawn για να ξεκινήσει μια διεργασία Python. Αυτό γίνεται με την εκτέλεση της εντολής “python3” και των παραμέτρων που παρέχονται. Οι παράμετροι περιλαμβάνουν το μονοπάτι προς το Python script “eps_calculator.py” και τα περαιτέρω ορίσματα (args) που έχουν προηγουμένως οριστεί. Έπειτα, χρησιμοποιείται μια μεταβλητή epsilon για να αποθηκευθεί η τιμή που παράγεται από τη διεργασία Python. Στη συνέχεια, προστίθεται ένας ακροατής για την stdout της διεργασίας Python, ώστε να διαχειρίζεται τα δεδομένα που εκτυπώνονται από το Python script. Όταν ληφθούν δεδομένα, αυτά μετατρέπονται σε συμβολοσειρά, διαγράφονται περιττά κενά χαρακτήρες (trim), και στη συνέχεια χωρίζονται σε γραμμές χρησιμοποιώντας τον χαρακτήρα νέας γραμμής (
n). Η πρώτη γραμμή από αυτές τις γραμμές αποθηκεύεται στη μεταβλητή epsilon. Αν η διεργασία ολοκληρωθεί με κωδικό επιστροφής 0, τότε σημαίνει ότι η εκτέλεση ολοκληρώθηκε επι-

τυχώς. Σε αυτήν την περίπτωση, αποστέλλεται μια απάντηση προς τον πελάτη (client) που περιέχει την τιμή της epsilon και το όνομα του προσωρινού αρχείου εικόνας. Εάν η διεργασία ολοκληρωθεί με οποιονδήποτε άλλο κωδικό επιστροφής, τότε επιστρέφεται μια απάντηση με κωδικό κατάστασης 500 που υποδηλώνει ότι παρουσιάστηκε κάποιο σφάλμα κατά την εκτέλεση του Python script

```
const pythonProcess = spawn("python3", [
  path.join(pythonScriptsFolder, "eps_calculator.py"),
  ...args,
]);

let epsilon: string;
pythonProcess.stdout.on("data", (data) => {
  epsilon = data.toString().trim().split("\n")[0];
});

// Wait for the Python script to complete
pythonProcess.on("close", (code) => {
  if (code === 0) {
    res.send({ epsilon, plotImage: tempImageFilename });
  } else {
    res.status(500).json({ error: "An error occurred on python script" });
  }
});
```

Σχήμα 4.10: Κώδικας εκτέλεσης findEpsilon script

Ας δούμε πως βρίσκουμε την τιμή eps. Ακολουθεί η επεξήγηση του σχήματος 4.11.
Η μεταβλητή dataSetFilePath αποθηκεύει τη διαδρομή προς το αρχείο εισόδου (dataset) το οποίο πρόκειται να φορτωθεί και να επεξεργαστεί.
Η μεταβλητή firstPlotFileName αποθηκεύει τη διαδρομή στην οποία θα αποθηκευτεί ο πρώτος γράφος (plot) που προκύπτει από την επεξεργασία.
Η μεταβλητή k περιέχει τον αριθμό των πλησιέστερων γειτόνων που θα ληφθούν υπόψιν κατά την επεξεργασία.
Η μεταβλητή selected_columns αποθηκεύει τη λίστα των στηλών που θα ληφθούν υπόψιν από το dataset, όπως ορίζεται στα ορίσματα που περάστηκαν κατά την κλήση του script.
Μέσω της βιβλιοθήκης pandas και της μεθόδου read_csv, το αρχείο dataset φορτώνεται και αποθηκεύεται στη μεταβλητή dataSet. Χρησιμοποιώντας την μεταβλητή selected_columns, επιλέγονται και αποθηκεύονται στη μεταβλητή data οι στήλες που έχουν καθοριστεί για επεξεργασία από το dataset.

```
dataSetFilePath = sys.argv[1]  # Path to the input dataset
firstPlotFileName = sys.argv[2]  # Path to save the first plot
k = int(sys.argv[3])  # The number of nearest neighbors to consider
selected_columns = sys.argv[4].split(',')  # The list of columns to be considered in the input dataset

# Load the data
dataSet = pd.read_csv(dataSetFilePath)  # Reading the input dataset
data = dataSet[selected_columns]  # Extracting the selected columns
```

Σχήμα 4.11: Κώδικας eps python script params input

Στο σχήμα 4.12 βλέπουμε την δημιουργία ενός μοντέλου Nearest Neighbors για μια συλλογή δεδομένων. Η μεταβλητή neighb δημιουργεί ένα αντικείμενο του μοντέλου NearestNeighbors με τον αριθμό των γειτόνων που θα χρησιμοποιηθούν να είναι k + 1. Το k + 1 επιλέγεται γιατί το μοντέλο θα βρει τους κοντινότερους γείτονες, συμπεριλαμβανομένου του εαυτού του κάθε σημείου. Η μεταβλητή nbrs εκπαιδεύει το μοντέλο NearestNeighbors χρησιμοποιώντας τα δε-

```
neighb = NearestNeighbors(n_neighbors=k+1)  # Initializing the NearestNeighbors model with k+1 neighbors
nbrs = neighb.fit(data)  # Fitting the model on the data
distances, indices = nbrs.kneighbors(data)  # Computing the k-neighbors for each point
distances = np.sort(distances[:, -1], axis=0)  # Sorting the distances
```

Σχήμα 4.12: Κώδικας eps python script neighbors

δομένα data που έχουν επιλεχθεί και φορτωθεί προηγουμένως. Έπειτα χρησιμοποιώντας τη μέθοδο .kneighbors(data), υπολογίζονται οι k πλησιέστεροι γείτονες για κάθε σημείο των δεδομένων. Τα αποτελέσματα αποθηκεύονται στις μεταβλητές distances και indices, όπου distances περιέχει τις αποστάσεις προς τους γείτονες και indices περιέχει τις αντίστοιχες δείκτες των γειτόνων. Τέλος, ο πίνακας αποστάσεων distances ταξινομείται σε αύξουσα σειρά, ενώ διατηρείται μόνο η τελευταία στήλη του πίνακα. Αυτό ουσιαστικά αντιπροσωπεύει τις αποστάσεις προς τον k-οστό πλησιέστερο γείτονα για κάθε σημείο στο σύνολο δεδομένων.

Στο σχήμα 4.13 φαίνεται ο κώδικας αναζήτησης της βέλτιστης τιμής eps. Η μεταβλητή differences δημιουργείται με τη χρήση της συνάρτησης np.diff, υπολογίζει τις διαφορές μεταξύ διαδοχικών αποστάσεων που έχουν αποθηκευτεί στη μεταβλητή distances. Έπειτα με τη χρήση της συνάρτησης np.diff δημιουργείται η μεταβλητή second_derivative για τον υπολογισμό της δεύτερης παραγώγου (διαφορά διαφορών) του πίνακα differences. Η μεταβλητή optimal_index υπολογίζεται με τη χρήση της συνάρτησης np.argmax, η οποία επιστρέφει τον δείκτη του μεγαλύτερου στοιχείου σε έναν πίνακα. Εδώ, εφαρμόζεται στον πίνακα second_derivative για να βρεθεί ο δείκτης της μέγιστης τιμής. Προστίθεται 1 στο αποτέλεσμα, καθώς οι δείκτες του πίνακα ξεκινούν από το 0 ενώ η θέση της πρώτης διαφοράς είναι 1. Τέλος η μεταβλητή optimal_eps αποθηκεύει την βέλτιστη τιμή ε (epsilon), η οποία αντιστοιχεί στην απόσταση στον πίνακα distances που βρίσκεται στη θέση optimal_index και εκτυπώνεται στην οθόνη.

```
differences = np.diff(distances)  # Computing the differences between consecutive distances
second_derivative = np.diff(differences)  # Computing the second derivative of the distances
optimal_index = np.argmax(second_derivative) + 1  # Finding the index of the maximum value in the second_derivative array
optimal_eps = distances[optimal_index]  # Finding the optimal epsilon value (the distance corresponding to the optimal_index)

print(optimal_eps)  # Printing the optimal epsilon value
```

Σχήμα 4.13: Κώδικας eps python script results print

DBSCAN

Η διαδικασία εκτέλεσης του DBSCAN script γίνεται με τον ίδιο τροπο που αναλύσαμε στην προηγούμενη ενότητα οποτε ας αναλύσουμε το ίδιο το python script για την εφαρμογή του αλγορίθμου DBSCAN.

```
dataSetFilePath = sys.argv[1]
generatedDatasetFilePath = sys.argv[2]
epsilon = float(sys.argv[3])
min_samples = int(sys.argv[4])
columns = sys.argv[5].split(',')
```

Σχήμα 4.14: Κώδικας dbscan python script params input

Ο παραπάνω κώδικας αναλαμβάνει την παραμετροποίηση και διαχείριση δεδομένων σε έναν αλγόριθμο συσταδοποίησης που βασίζεται στην πυκνότητα, γνωστός ως DBSCAN (Density-Based Spatial Clustering of Applications with Noise). Συγκεκριμένα:

- Η μεταβλητή dataSetFilePath αποθηκεύει τη διαδρομή προς το αρχείο εισόδου (dataset), το οποίο θα χρησιμοποιηθεί για την εκτέλεση του αλγορίθμου DBSCAN.
- Η μεταβλητή dataSetFilePath αποθηκεύει τη διαδρομή προς το αρχείο εισόδου (dataset), το οποίο θα χρησιμοποιηθεί για την εκτέλεση του αλγορίθμου DBSCAN.
- Η μεταβλητή generatedDatasetFilePath αποθηκεύει τη διαδρομή προς το αρχείο εξόδου, όπου τα δεδομένα θα αποθηκευτούν μετά την εκτέλεση του αλγορίθμου.
- Η μεταβλητή epsilon καθορίζει τη μέγιστη απόσταση μεταξύ δύο δειγμάτων, ώστε να θεωρούνται στη γειτνίαση το ένα με το άλλο. Αυτό είναι το κριτήριο της πλησιέστερης γειτονιάς που χρησιμοποιεί ο αλγόριθμος.
- Η μεταβλητή min_samples καθορίζει το ελάχιστο αριθμό δειγμάτων σε μια γειτονιά, ώστε ένα σημείο να θεωρείται πυρήνας. Αυτός ο πυρήνας πρέπει να έχει τουλάχιστον min_samples γείτονες εντός της ακτίνας

Στο σχήμα 4.15 βλέπουμε τις βασικές ενέργειες του αλγορίθμου dbscan. Η μεταβλητή dataset χρησιμοποιεί τη βιβλιοθήκη pandas για να διαβάσει το αρχείο dataset που δόθηκε στην διαδρομή dataSetFilePath. Μέσω της μεταβλητής data, εξάγονται όλες οι στήλες του αρχικού dataset, εκτός από την τελευταία.

```
dataset = pd.read_csv(dataSetFilePath)  # Reading the input dataset
data = dataset.iloc[:, 0:-1]  # Extracting all columns except the last one
dbscan = DBSCAN(eps=epsilon, min_samples=min_samples)  # Initializing the DBSCAN model
dbscan.fit(data)  # Fitting the model on the data
# Adding a 'cluster' column to the dataset with the labels assigned to each data point
dataset['cluster'] = dbscan.labels_.tolist()
 # Saving the modified dataset with the 'cluster' column to a new file
dataset.to_csv(generatedDatasetFilePath)
```

Σχήμα 4.15: Κώδικας dbscan python script

Ο κώδικας εκτελεί τα εξής βήματα:

- Δημιουργεί ένα αντικείμενο του αλγορίθμου DBSCAN με τη χρήση των παραμέτρων epsilon και min_samples που έχουν δοθεί, προκειμένου να προσαρμόσει το μοντέλο στα δεδομένα. Η μεταβλητή dbscan αποθηκεύει αυτό το αντικείμενο.
- Χρησιμοποιώντας τη μέθοδο .fit(data), εκτελείται η εκπαίδευση του μοντέλου DBSCAN στα δεδομένα που εξήχθησαν (χωρίς την τελευταία στήλη).
- Προσθέτεται μια νέα στήλη με όνομα 'cluster' στο αρχικό dataset, η οποία περιέχει τις ετικέτες που αντιστοιχούν σε κάθε σημείο συστάδας. Οι ετικέτες αποθηκεύονται στο πίνακα labels_ που παρέχει το μοντέλο DBSCAN.
- Τα τροποποιημένα δεδομένα του dataset αποθηκεύονται σε ένα νέο αρχείο, το οποίο έχει τη διαδρομή που έχει δοθεί στην μεταβλητή generatedDatasetFilePath. Αυτό το νέο αρχείο περιλαμβάνει τις ετικέτες συστάδας που έχουν ανατεθεί σε κάθε σημείο.

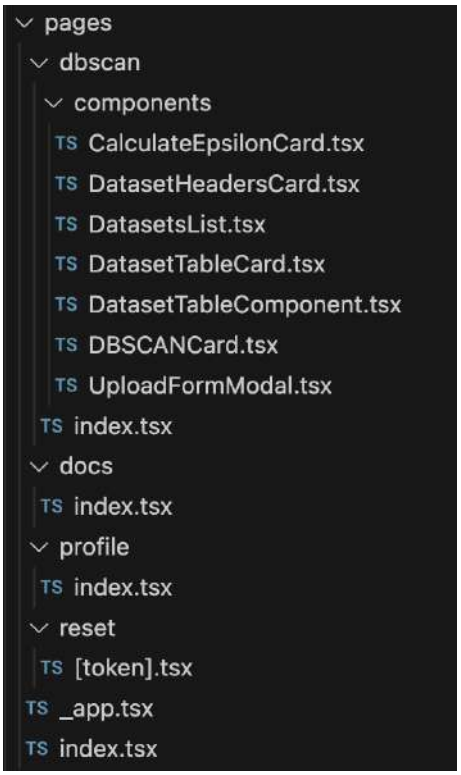
4.4 Υλοποίηση του Client

Για την υλοποίηση του πελάτη (client), αξιοποιήθηκαν σύχρονες τεχνολογίες, συμπεριλαμβανομένων των Next.js, Javascript, HTML, CSS και του Tailwind CSS.

Κατά την ανάπτυξη, το πλαίσιο της Next.js χρησιμοποιήθηκε για τη δημιουργία του project. Έτσι, η δομή του project ακολουθεί τα πρότυπα του συγκεκριμένου πλαισίου. Ωστόσο, τα HTML tags παραμένουν αναπόσπαστο μέρος της ανάπτυξης, καθώς χρησιμεύουν για την απεικόνιση γραφικών στοιχείων στον φυλλομετρητή.

Η δομή ενός Next.js project ακολουθεί έναν οργανωμένο και συνεκτικό τρόπο, παρέχοντας ένα πλαίσιο για την ανάπτυξη δυναμικών ιστοσελίδων και εφαρμογών. Η δομή αυτή βοηθάει στην οργάνωση του κώδικα, την διαχείριστη διάφορων λειτουργιών και τη διατήρηση μιας σαφής εικόνα του project.
Ας ρίξουμε μια ματιά στην δομή του project:

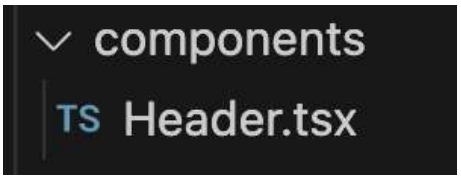
- Φάκελος pages: Ο πυρήνας του Next.js project βρίσκεται εδώ. Κάθε αρχείο JavaScript (ή TypeScript) που δημιουργείται σε αυτόν το φάκελο αποτελεί μια σελίδα της εφαρμογής. Το όνομα του αρχείου καθορίζει το URL μονοπάτι της σελίδας.



Σχήμα 4.16: Δομή Client: Pages

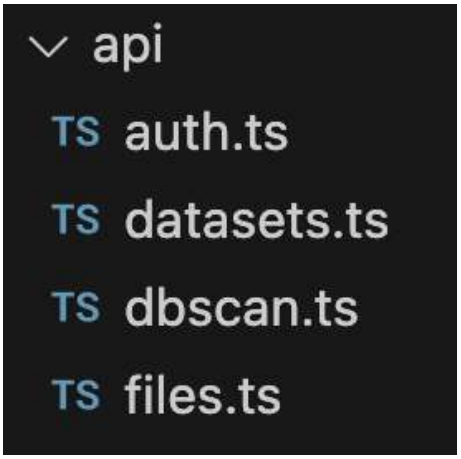
- Φάκελος public: Εδώ τοποθετούνται στατικά αρχεία όπως εικόνες, αρχεία CSS και άλλα που δεν απαιτούν επεξεργασία από το Next.js.

- Αυτός ο φάκελος περιέχει επαναχρησιμοποιήσιμα React (ή React-like) στοιχεία που μπορούν να χρησιμοποιηθούν σε διάφορες σελίδες. Αυτό βελτιώνει την οργάνωση και τη συντηρησιμότητα του κώδικα.



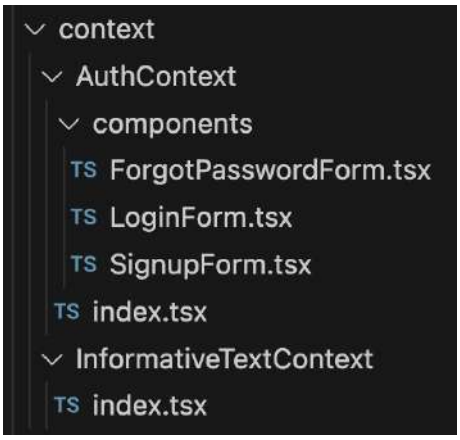
Σχήμα 4.17: Δομή Client: Components

- Φάκελος styles: Εδώ μπορείτε να τοποθετήσετε τα αρχεία CSS, SASS ή άλλες εναλλακτικές διαμορφώσεις στυλ.
- Αρχείο package.json: Το αρχείο που περιέχει τις πληροφορίες για το project, τις εξαρτήσεις, τα scripts και άλλες ρυθμίσεις.
- Φάκελος node_modules: Εδώ αποθηκεύονται οι εξαρτήσεις του project που εγκαθίστανται μέσω του npm.
- Αρχείο next.config.js: Αν τοποθετηθεί, αυτό το αρχείο παρέχει προηγμένες ρυθμίσεις του Next.js.
- Άλλα αρχεία και φακέλους:
  APIs Structure: Στον φακελο αυτό βρίσκονται όλες οι κλήσεις των διαφόρων endpoints που χρησιμοποιεί η ιστοσελίδα μας



Σχήμα 4.18: Δομή Client: APIs

Δημιουργία Global components με τη χρήση context. Το “context” στο React είναι ένας μηχανισμός για την κοινοποίηση δεδομένων μεταξύ συστατικών, επιτρέποντας πρόσβαση σε δεδομένα χωρίς τη συνεχή χρήση props. Βοηθά στην απλούστευση του κώδικα και τη βελτίωση της διαχείρισης κατάστασης (state).



Σχήμα 4.19: Δομή Client: Context

Για τη διαμόρφωση των γραφικών στοιχείων, επιλέχθηκε η βιβλιοθήκη Tailwind CSS. Αυτή η βιβλιοθήκη μας παρέχει τη δυνατότητα να ορίσουμε τα CSS styles εύκολα και γρήγορα, όπως παρατηρούμε στο παρακάτω παράδειγμα. Η χρήση του Tailwind CSS επιτρέπει τη δημιουργία ορισμένων στυλ μέσα στον κώδικα του στοιχείου μας, με απλό και κατανοητό τρόπο.

```
<div className="flex flex-col items-center min-h-screen bg-fixed bg-center bg-cover custom-background-img relative">
  <div className="flex-col items-center justify-center overflow-auto mt-[77px] w-full max-h-[calc(100vh-77px)] pb-[150px] pt-[123px]">
```

Σχήμα 4.20: Παράδειγμα χρήσης Tailwind CSS

Για να αποστείλει αιτήσεις HTTP ο client προς τον server χρησιμοποιείται η βιβλιοθήκη axios.

Αρχικά ο clinet δημιουργεί μια αίτηση HTTP, συνήθως με τη μέθοδο axios.get(), axios.post() κ.λπ., καθορίζοντας το URL του ενός endpoint στον διακομιστή. Στέλνει την αίτηση προς τον διακομιστή χρησιμοποιώντας τη μέθοδο Axios που αντιστοιχεί στον τύπο της αίτησης (GET, POST, κλπ.). Ο διακομιστής επεξεργάζεται την αίτηση και αποστέλλει πίσω την ανταπόκρισή του, περιέχοντας τα απαιτούμενα δεδομένα. Μόλις η ανταπόκριση ληφθεί από τον client, μπορεί να γίνει επεξεργασία των δεδομένων, όπως η απόδοση τους σε συστατικά της εφαρμογής για εμφάνιση στο χρήστη. Οπως φαίνεται στο Σχήμα 4.21.

```
export const fetchPublicDatasets = async (): Promise<string[] | null> => {
  try {
    const response: AxiosResponse<string[]> = await axios.get(
      "http://localhost:8081/api/fetchPublicDatasets"
    );
    return response.data;
  } catch (error: any) {
    return null;
  }
};
```

Σχήμα 4.21: Παράδειγμα axios request

Οι αιτήσεις προς τον διακομιστή μπορούν να περιλαμβάνουν παραμέτρους, κεφαλίδες και άλλες πληροφορίες, ανάλογα με τις απαιτήσεις της συγκεκριμένης αίτησης. Η Axios παρέχει ευκολία και ευελιξία στη διαδικασία αποστολής και λήψης δεδομένων μεταξύ client και server, βοηθώντας στην αποτελεσματική ανάπτυξη και λειτουργία των εφαρμογών Next.js.

Διαδικασία αυθεντικοποίησης χρήστη.
Οταν ένας χρήστης συνδεθεί στο σύστημα επιστρέφεται στην απάντηση του server ένα authorization token. Στη συνέχεια χρησιμοποιείται η βιλβιοθήκη Cookies για να κάνει set στο session του browser το συγκεκριμενο token. Αυτό εξυπηρετεί τον χρήστη όντας αποθηκευμένο στα Cookies του browser ώστε την επόμενη φορα μέσα στο χρονικό διάστημα που θα είναι εγκυρο το token, να μην χρειαστεί να ξανά ακολουθήσει την διαδικασία σύνδεσης.

```
const response = await loginUser({ email, password });
if (response && typeof response !== "string") {
  if (response.user && response.accessToken) {
    Cookies.set("accessToken", response?.accessToken);
    setUser(response.user);
    setAccessToken(response.accessToken);
    setIsOpen(!isOpen);
    router.push("/");
    return true;
```

Σχήμα 4.22: Αποθήκευση authorization token στα Cookies

4.5 Github repository

Το GitHub repository (ή απλώς "repo") αντιπροσωπεύει ένα αποθετήριο ηλεκτρονικών αρχείων που φιλοξενείται στην πλατφόρμα GitHub. Αυτό το αποθετήριο μπορεί να περιλαμβάνει κώδικα, έγγραφα, εικόνες, αρχεία δεδομένων και άλλα είδη πληροφοριών. Το GitHub αποτελεί ένα ευρέως διαδεδομένο εργαλείο συνεργασίας για προγραμματιστές και άλλους επαγγελματίες, παρέχοντας μια πλατφόρμα για τη διαχείριση και την κοινή χρήση κώδικα και έργων.

Ένα GitHub repository προσφέρει τα εξής χαρακτηριστικά:

- Διαχείριση Κώδικα: Το repo είναι ένας κεντρικός τόπος για την αποθήκευση, την παρακολούθηση και την επεξεργασία κώδικα. Οι προγραμματιστές μπορούν να κάνουν αναζήτηση στο ιστορικό αλλαγών, να δημιουργούν νέα υποκαταλόγους και να συγχωνεύουν τις αλλαγές.
- Συνεργασία: Πολλοί χρήστες μπορούν να συνεισφέρουν στο repo, κάνοντας αλλαγές, προσθέτοντας νέο κώδικα ή διορθώνοντας σφάλματα. Οι συνεισφορές γίνονται μέσω διαδικασιών όπως οι "pull requests."
- Ιστορικό Αλλαγών: Κάθε αλλαγή στον κώδικα αποθηκεύεται, παρέχοντας μια λεπτομερή ανασκόπηση της εξέλιξης του έργου. Αυτό είναι χρήσιμο για την ανίχνευση σφαλμάτων, την παρακολούθηση προόδου και την επαναφορά σε προηγούμενες εκδόσεις.
- Σχολιασμός: Οι χρήστες μπορούν να σχολιάζουν τον κώδικα, τις συνεισφορές ή τα issues, βοηθώντας έτσι στην ανταλλαγή απόψεων και στην επίλυση προβλημάτων.
- Εκδόσεις: Το GitHub παρέχει ένα σύστημα για τη διαχείριση εκδόσεων του κώδικα, επιτρέποντας τη δημιουργία σταθερών και ασταθών εκδόσεων.
- Διαχείριση Προβλημάτων: Οι χρήστες μπορούν να δημιουργούν "issues" για τα προβλήματα που αντιμετωπίζουν, τις προτάσεις βελτίωσης και τις ιδέες, παρέχοντας ένα τρόπο για τη διαχείριση και την παρακολούθηση των ανοιχτών θεμάτων.

Τα GitHub repositories αποτελούν απαραίτητα εργαλεία για τη συνεργασία, τη διαμοιρασμό γνώσης και τη διαχείριση έργων σε ποικίλους τομείς, από τον ανοικτό κώδικα έως την επιχειρηματική ανάπτυξη και την ακαδημαϊκή έρευνα.

Το παρόν project βρίσκεται στην σελίδα https://github.com/johnkeik/DBSCAN-Project

Κεφάλαιο 5

Παρουσιάση του AutoDBSCAN

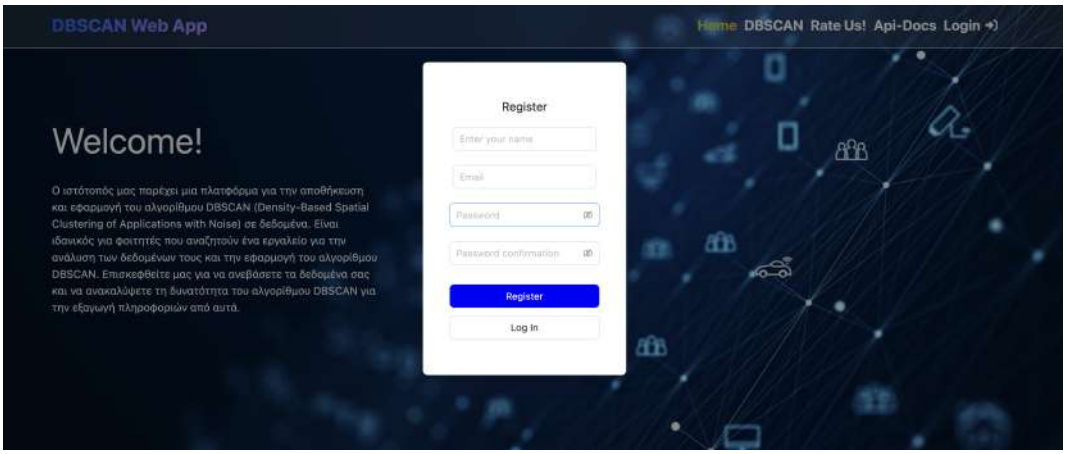
5.1 Αρχική σελίδα

Στο σχήμα 5.1 φαίνεται η αρχική σελίδα της εφαρμογής. Εδώ, ο χρήστης μπορεί να δει μια περιγραγή της εφαρμογής.

Σχήμα 5.1: Αρχική σελίδα

5.2 Εγγραφή νέου χρήστη

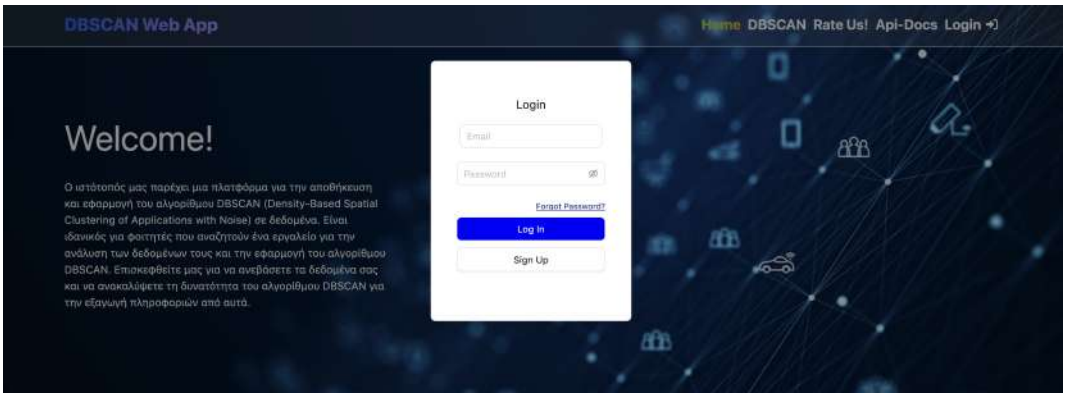
Στο σχήμα 5.2 φαίνεται η σελίδα εγγραφής χρήστη. Στην σελίδα αυτή, ο χρήστης μπορεί να δημιουργήσει ένα νέο λογαριασμό στην εφαρμογή. Οι βασικές πληροφορίες που απαιτούνται για την εγγραφή περιλαμβάνουν όνομα χρήστη, ηλεκτρονική διεύθυνση και κωδικό πρόσβασης.



Σχήμα 5.2: Δημιουργία λογαριασμού

5.3 Σύνδεση χρήστη στο σύστημα

Στο σχήμα 5.3 φαίνεται η σελίδα εισόδου. Οι υπάρχοντες χρήστες μπορούν να συνδεθούν στο σύστημα χρησιμοποιώντας την ηλεκτρονική τους διεύθυνση και τον κωδικό πρόσβασης.



Σχήμα 5.3: Σύνδεση χρήστη στο σύστημα

5.4 Επεξεργασία προσωπικών στοιχείων και κωδικού πρόσβασης

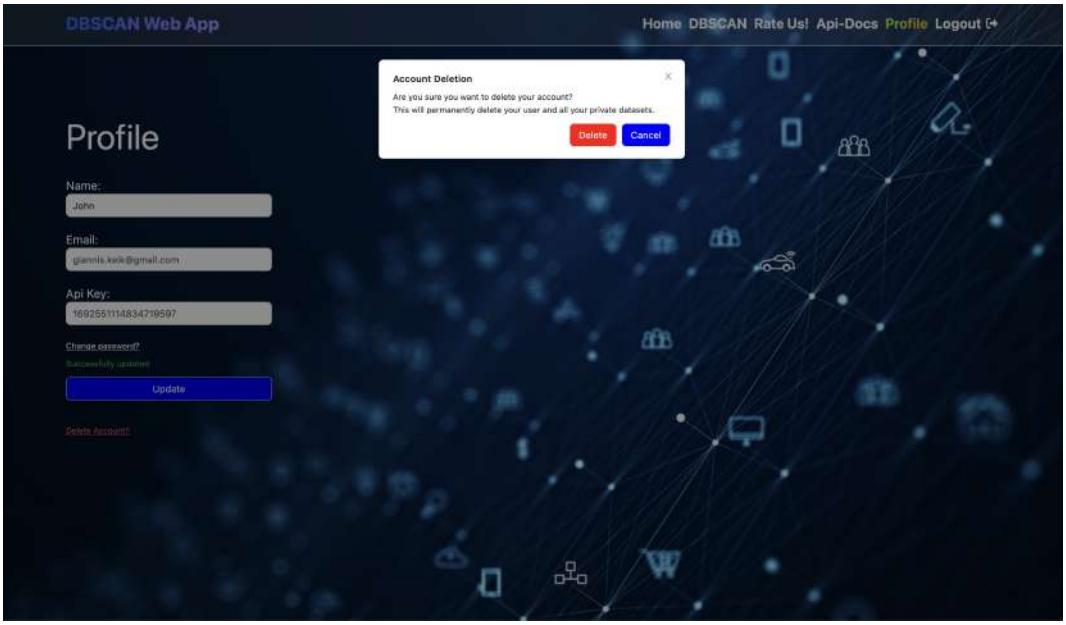
Στο σχήμα 5.4 φαίνεται η σελίδα του χρήστη. Εδω μπορεί να ενημερώσει ή να αλλάξει τα προσωπικά του στοιχεία, όπως το όνομα, την ηλεκτρονική διεύθυνση ή τον κωδικό πρόσβασης (Σχήμα 5.5), για να διατηρήσει το προφίλ του ενημερωμένο.

Σχήμα 5.4: Επεξεργασία προσωπικών στοιχείων

Σχήμα 5.5: Αλλαγή κωδικού πρόσβασης

5.5 Διαγραφή Λογαριασμού

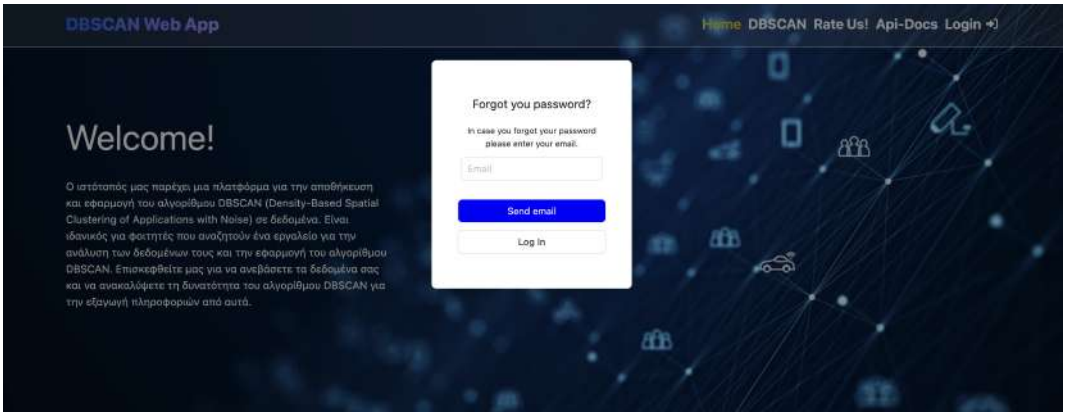
Στο σχήμα 5.6 φαίνεται η περίπτωση που ο χρήστης επιθυμεί να διαγράψει τον λογαριασμό του.



Σχήμα 5.6: Διαγραφή Λογαριασμού

5.6 Ανάκτηση κωδικού πρόσβασης

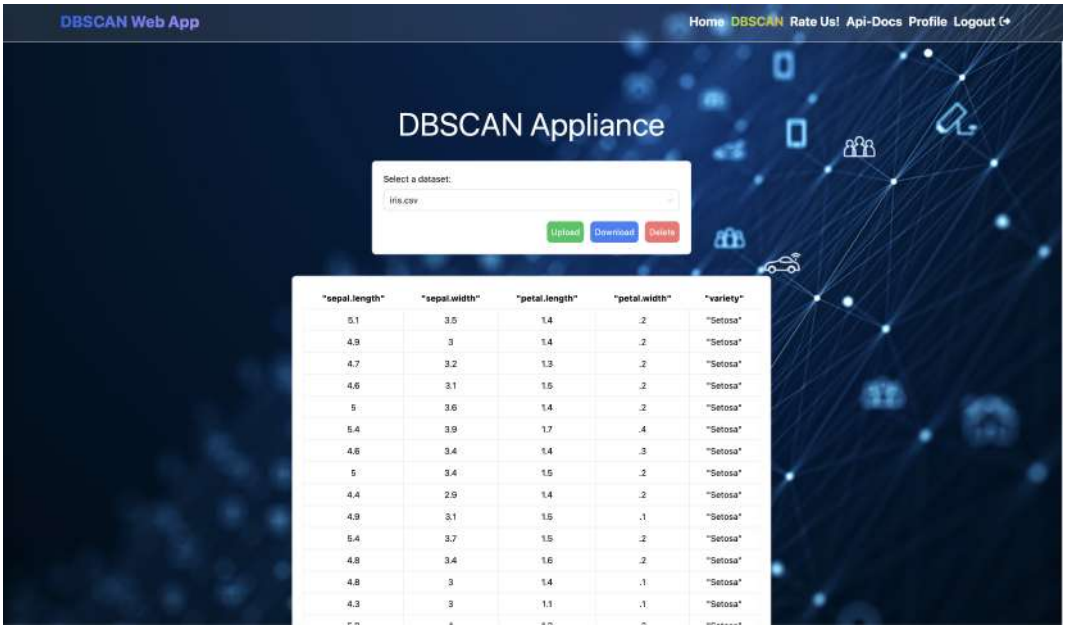
Στο σχήμα 5.7 φαίνεται η φόρμα που πρέπει να συμπληρώσει ένας χρήστης σε περίπτωση που ξεχάσει τον κωδικό του.



Σχήμα 5.7: Ανάκτηση κωδικού πρόσβασης

5.7 Σελίδα DBSCAN

Στο σχήμα 5.8 φαίνεται η σελίδα DBSCAN. Εδώ ο χρήστης μπορεί να επξεργαστεί τα συνολα δεδομένων τους, να ανεβάσει καινούρια και να εφαρμόσει τον αλγόριθμο DBSCAN. Για τις επιμέρους λειτουργίες της σελίδας θα αναφερθούμε στις επόμενες ενότητες

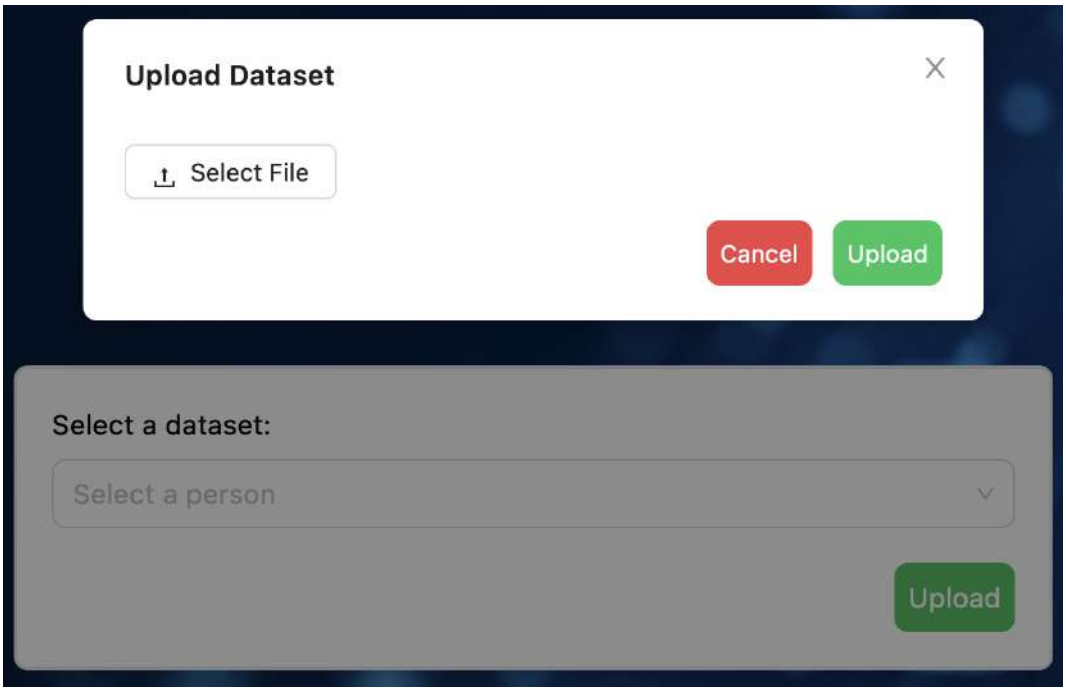


Σχήμα 5.8: Σελίδα DBSCAN

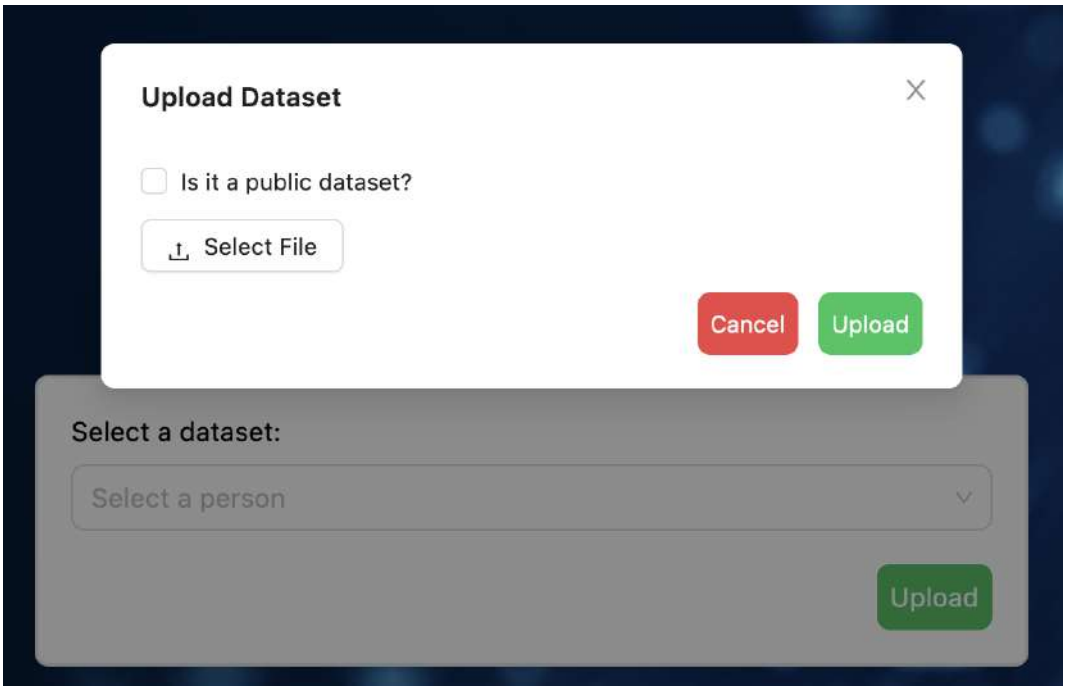
Σχήμα 5.9: Λίστα συνόλων δεδομένων

5.8 Ανέβασμα αρχείου

Ο χρήστης μπορεί να ανεβάσει αρχεία δεδομένων που επιθυμεί να αναλύσει μέσω της εφαρμογής. Στο σχήμα 5.10 φαίνεται η φόρμα που βλέπει ένας απλός χρήστης ενω στο 5.11 φαίνεται η φόρμα που βλέπει ένας χρήστης με πρόσβαση στη δημοσίευση δημοσίων αρχείων.



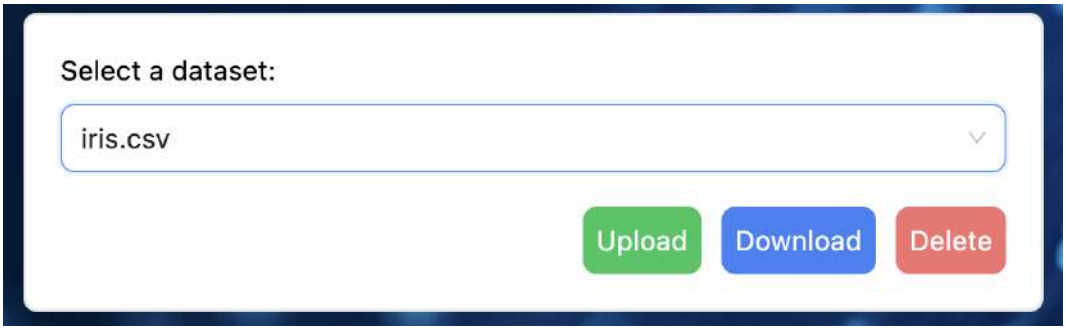
Σχήμα 5.10: Ανέβασμα ιδιωτικού αρχείου



Σχήμα 5.11: Ανέβασμα δημόσιου αρχείου

5.9 Διαγραφή αρχείου

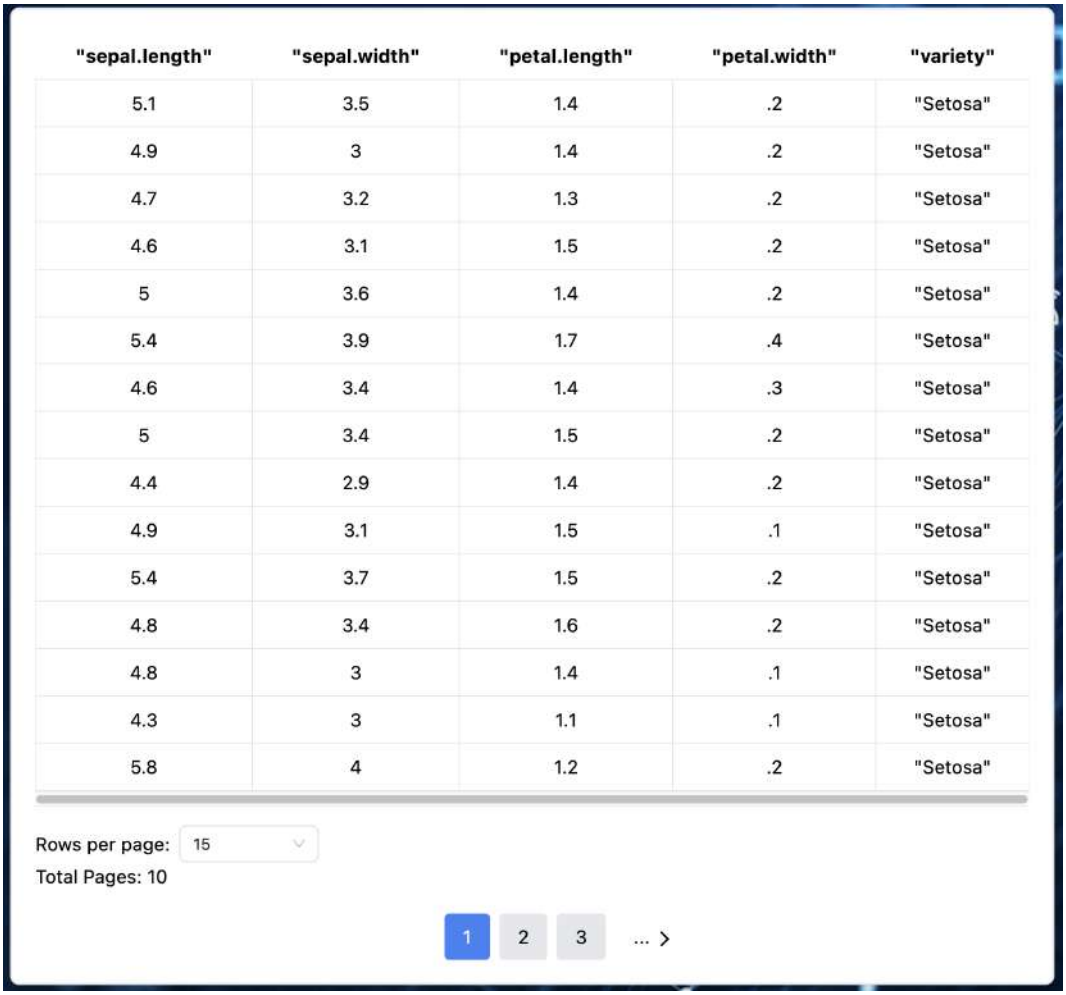
Εάν ένα αρχείο δεν είναι πλέον απαραίτητο, ο χρήστης μπορεί να το διαγράψει από το σύστημα όπως φαίνεται στο σχήμα 5.12.



Σχήμα 5.12: Διαγραφή συνόλου

5.10 Ανάγνωση αρχείου

Στο σχήμα 5.13 φαίνεται η προεπισκόπηση ενός συνόλου δεδομένων. Εδω ο χρήστης μπορεί να δει τα περιεχόμενα του συνόλου ώστε να επιλέξει στο επόμενο βήμα σωστά τις στήλες που περιέχουν μόνο αριθμούς.

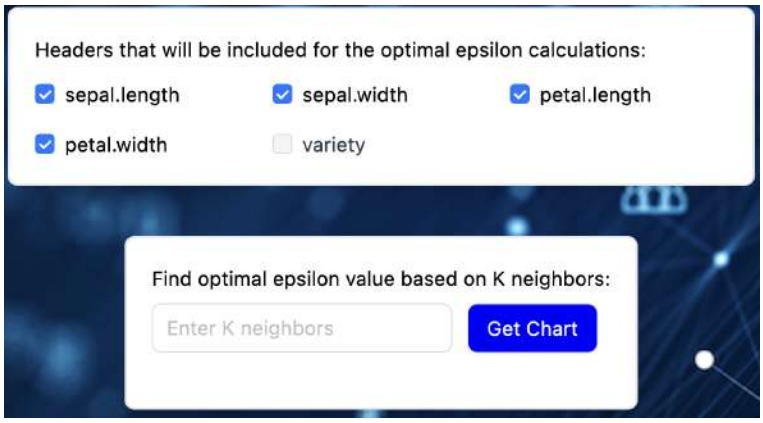


"sepal.length"	"sepal.width"	"petal.length"	"petal.width"	"variety"
5.1	3.5	1.4	.2	"Setosa"
4.9	3	1.4	.2	"Setosa"
4.7	3.2	1.3	.2	"Setosa"
4.6	3.1	1.5	.2	"Setosa"
5	3.6	1.4	.2	"Setosa"
5.4	3.9	1.7	.4	"Setosa"
4.6	3.4	1.4	.3	"Setosa"
5	3.4	1.5	.2	"Setosa"
4.4	2.9	1.4	.2	"Setosa"
4.9	3.1	1.5	.1	"Setosa"
5.4	3.7	1.5	.2	"Setosa"
4.8	3.4	1.6	.2	"Setosa"
4.8	3	1.4	.1	"Setosa"
4.3	3	1.1	.1	"Setosa"
5.8	4	1.2	.2	"Setosa"

Σχήμα 5.13: Προεπισκόπηση συνόλου δεδομένων

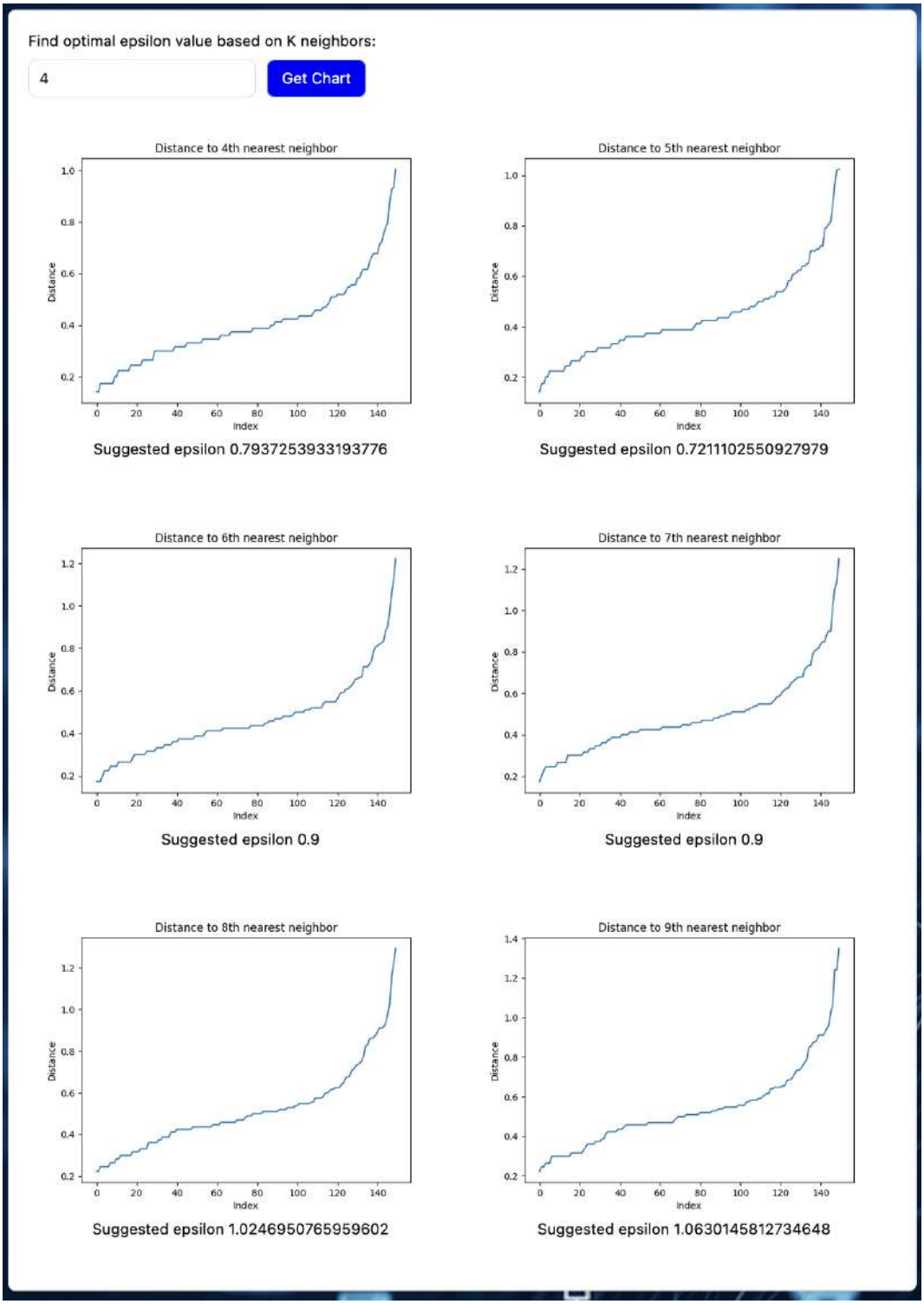
5.11 Μέθοδος προσδιορισμού eps

Στο σχήμα 5.14 φαίνεται η φόρμα που πρέπει να συμπληρώσει ο χρήστης ώστε να χρησιμοποιήσει την μέθοδο προσδιορισμού eps. Βλέπουμε οτι υπάρχει μία λίστα με τις κεφαλίδες των στηλών του συνόλου, απο τις οποίες ειναι προεπιλεγμένες μόνο αυτές που περιέχουν αριθμητικά δεδομένα. Ο χρήστης μπορεί να επιλέξει ποιες στήλες θα λάβουν μέρος στην διαδικασία. Έπειτα πρέπει να επιλέξει τον αριθμό γειτόνων που επιθυμεί ώστε να τρέξει η μέθοδος και να του προτείνει τιμές για το eps για την τιμή K που έδωσε καθώς και για τις 5 επόμενες.



Σχήμα 5.14: Μέθοδος προσδιορισμού eps

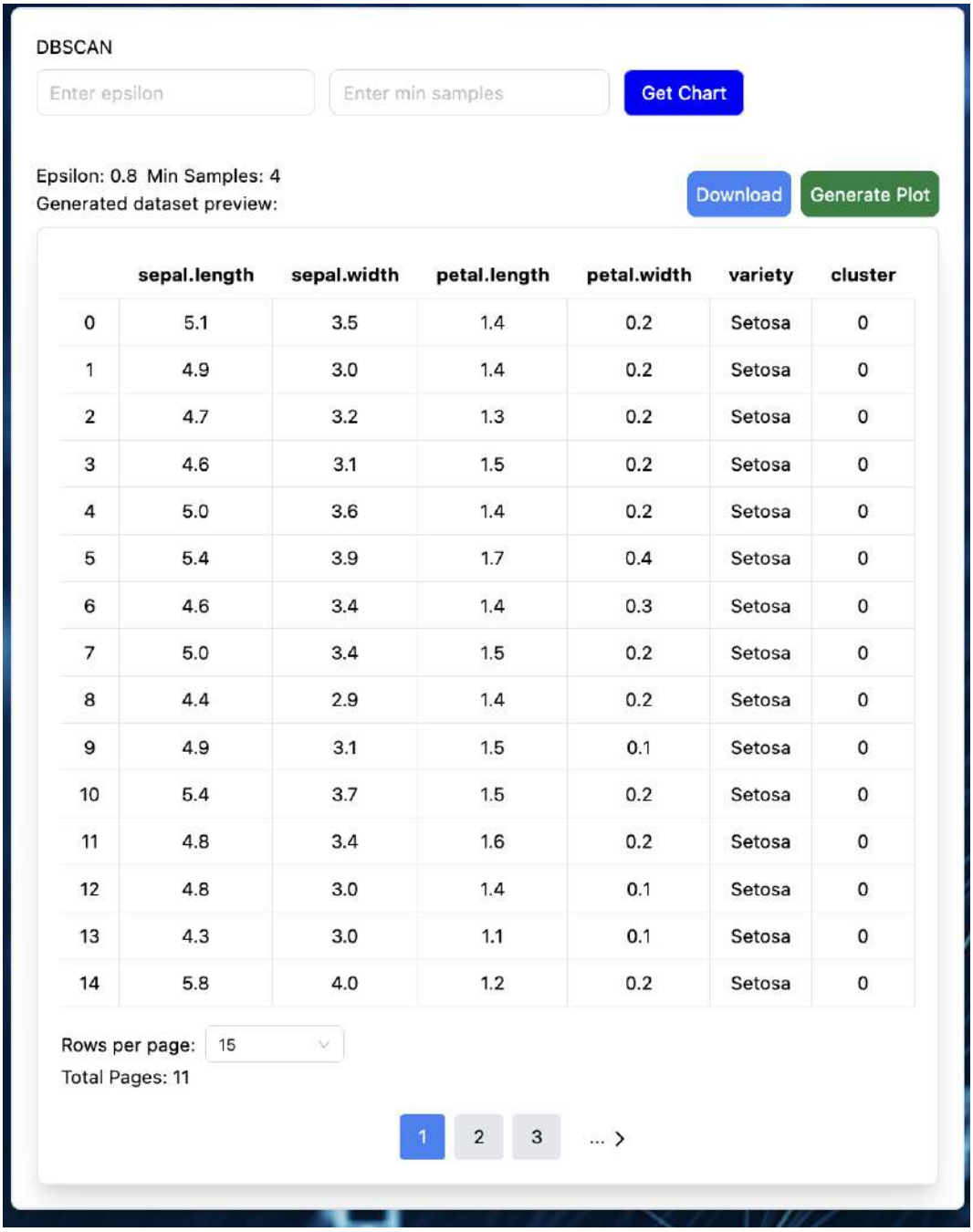
Στο σχήμα 5.15 βλέπουμε τα αποτελέσματα της μεθόδου. Επειδή οι τιμές των αποτελεσμάτων ειναι απλώς μια πρόταση δίνεται η δυνατότητα στον χρήστη να μπορεί να πατήσει σε οποιαδήποτε εικόνα για να την μεγενθύνει ωστε να μπορεί να υπολογίσει και μόνος του την βέλτιστη τιμη για το eps με βάση την αύξηση της καπυλότητας.



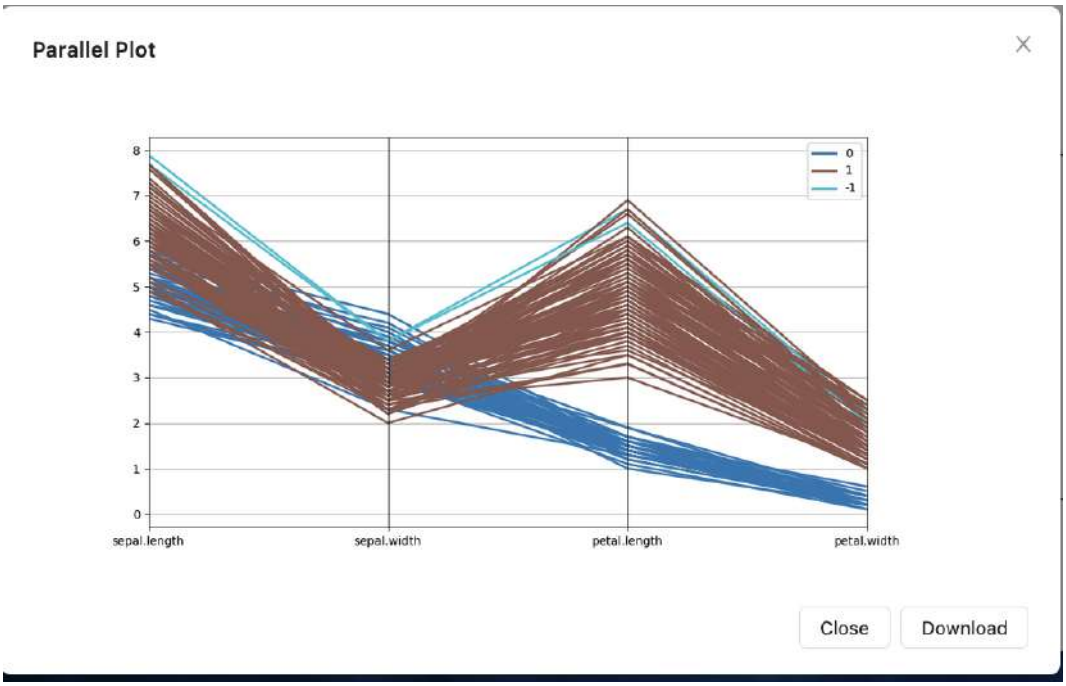
Σχήμα 5.15: Αποτελέσματα μεθόδου προσδιορισμού eps

5.12 Συσταδοποίηση DBSCAN

Εδώ, ο χρήστης μπορεί να εισάγει τις παραμέτρους που επιθυμεί και να εφαρμόσει τον αλγόριθμο DBSCAN για τα δεδομένα του. Στο σχήμα 5.16 βλέπουμε την προεπισκόπηση ενος συσταδοποιημένου συνόλου για τις τιμές $eps = 0.8$ και $minPts = 4$.
Στο σχήμα 5.17 φαίνεται το γράφημα το οποίο υπολογίζεται οταν ο χρήστης πατήση το κουμπί 'Generate Plot'



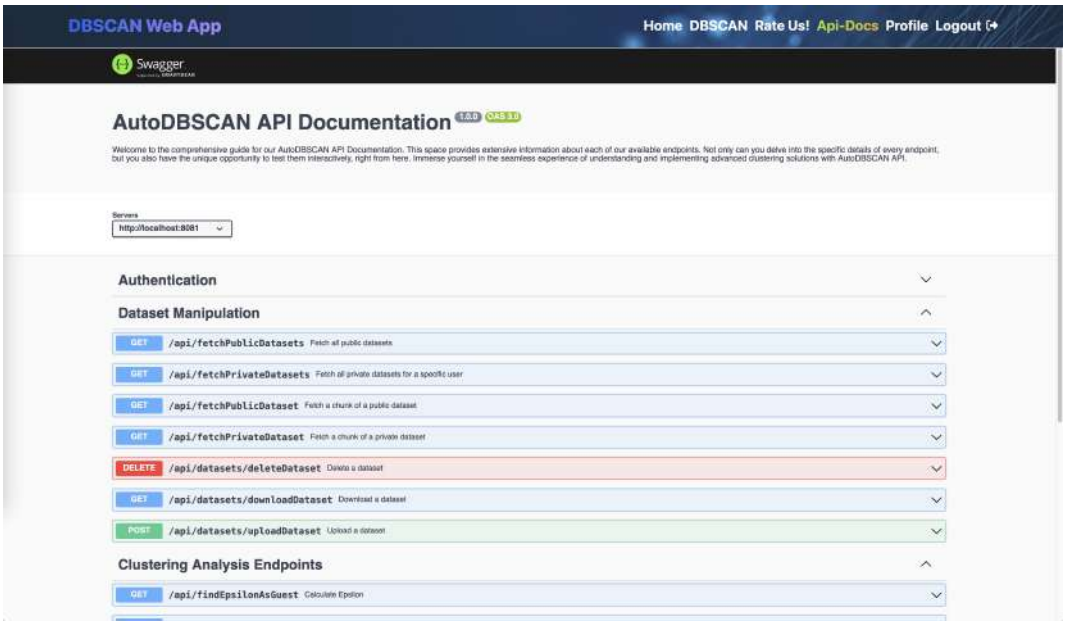
Σχήμα 5.16: Προεπισκόπιση συνόλου αποτελέσματος συσταδοποίησης



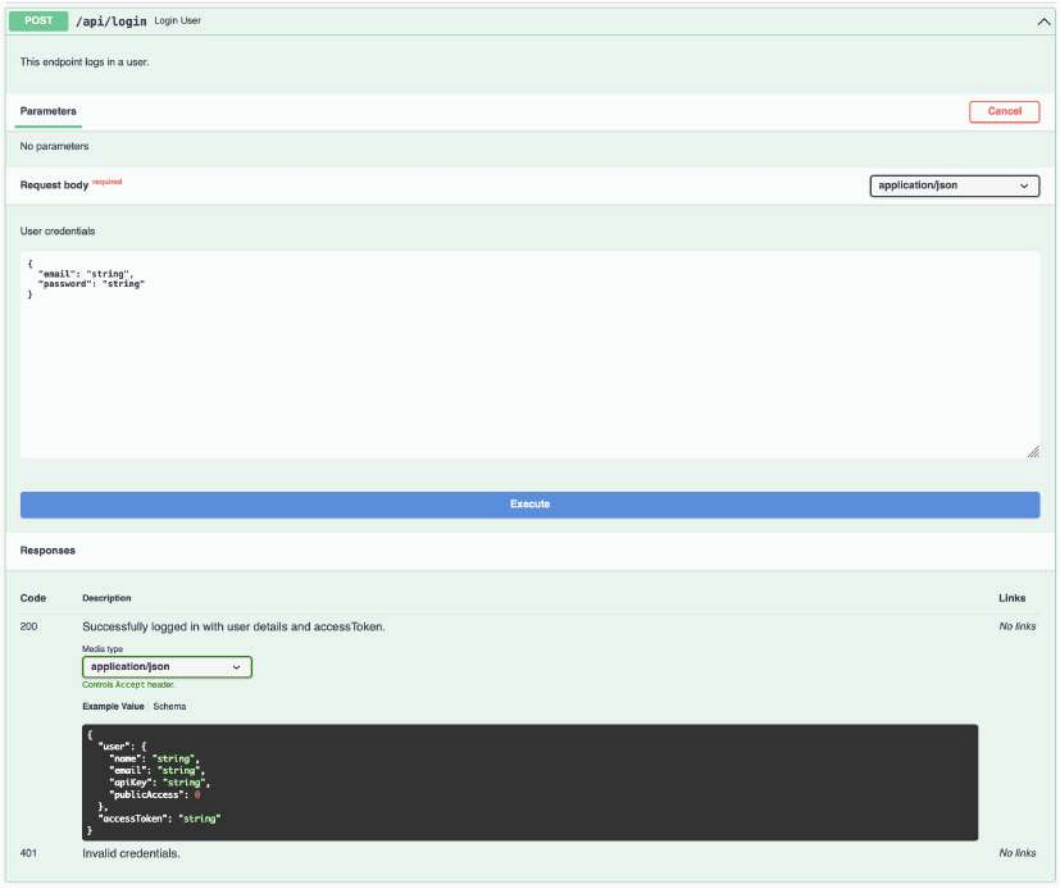
Σχήμα 5.17: Γράφημα συσταδοποίησης

5.13 Public API

Η εφαρμογή προσφέρει ένα Public API όπου οι προγραμματιστές μπορούν να ανακτήσουν, να αναλύσουν ή να στείλουν δεδομένα στην εφαρμογή. Η προβολή και η επεξήγηση των API's γίνεται σε μια εξατομικευμένη σελίδα που δημιουργήθηκε γι αυτόν τον σκοπό. Στο σχήμα 5.18 φαίνεται η σελίδα με την λίστα των διαθέσιμων API's.
Επιλέγοντας μία αναφορά γίνεται επέκταση των πληροφοριών και μπορούμε να δούμε τις απαιτήσεις/ προυποθέσεις που χρειάζεται ενα endpoint για να λειτουργήσει.
Ένα καλο χαρακτηριστικό της τεχνολογίας που επιλέχθηκε για την προβολή των API's είναι η δυνατότητα να δοκιμαστεί το endpoint ζωντανά μέσω της ιστοσελίδας (Σχήμα 5.19)



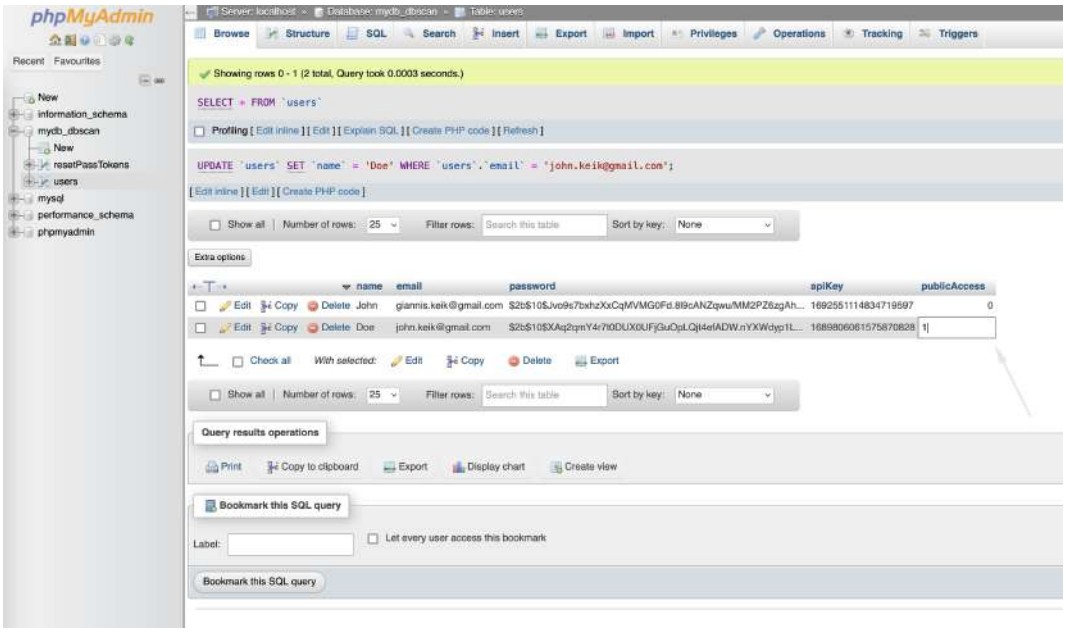
Σχήμα 5.18: Σελίδα προβολής API



Σχήμα 5.19: Παράδειγμα API

5.14 Διαχείρηση χρηστών

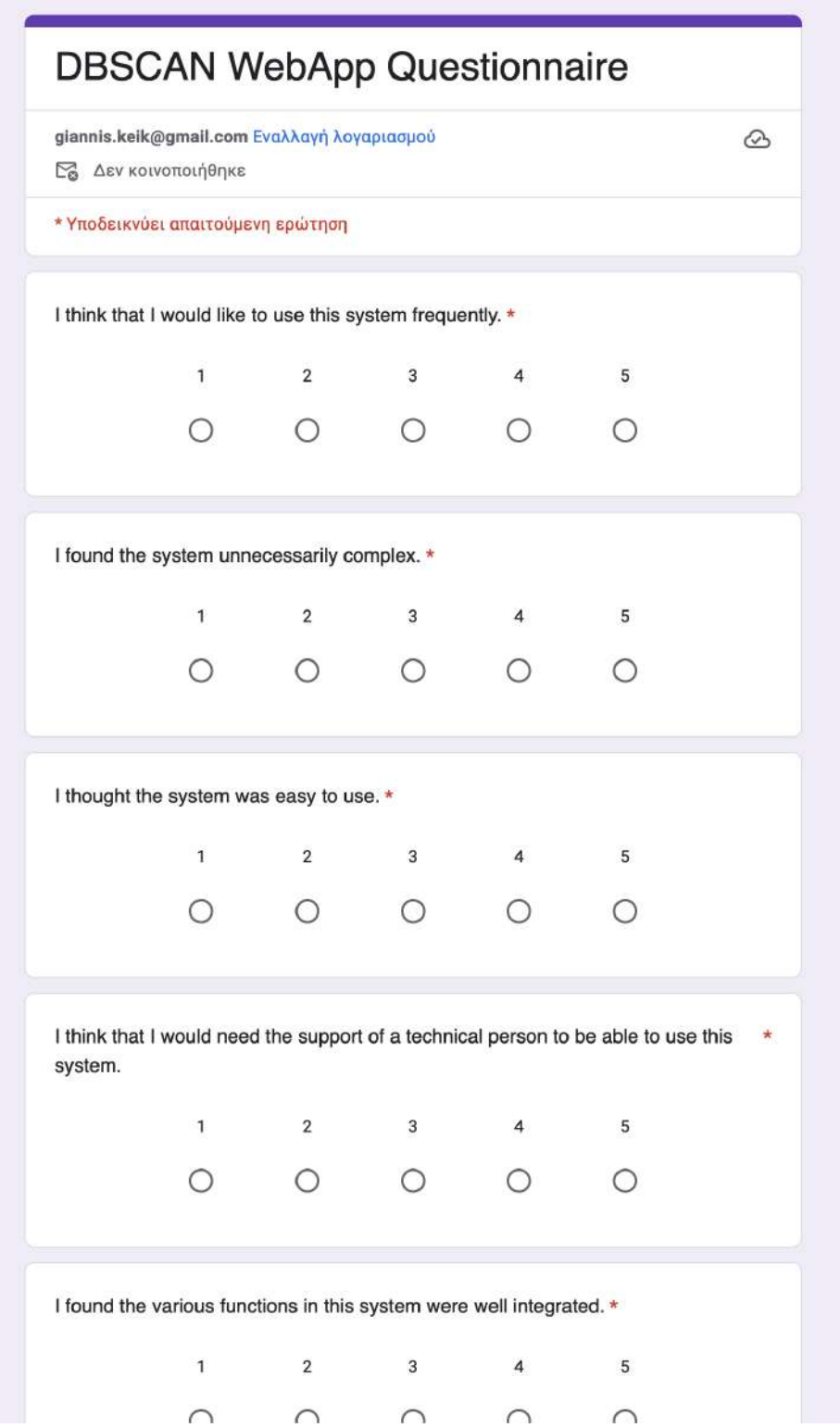
Στο Σχήμα 5.20 φάινεται η σελίδα με την οποία οι διαχειριστές της σελιδα μπορούν να επεξεργαστούν το ποιος χρήστης έχει δικαίωμα να δημοσιεύει δημόσια σύνολα δεδομένων.



Σχήμα 5.20: Διαχείρηση δικαιωμάτων χρηστών

5.15 Αξιολόγηση Εμπειρίας Χρήσης

Πατώντας το 'Rate Us' απο την γραμμή πλοήγησης της σελίδας ο χρήστης μεταφέρεται σε ενα Google Form το οποιο περιέχει 10 ερωτήσεις σύμφωνα με το System Usability Scale (SUS) για την αξιολόγηση εμπειρίας χρήστη



Σχήμα 5.21: Ερωτηματολόγιο Εμπειρίας Χρήστη

Κεφάλαιο 6

Αξιολόγηση του AutoDBSCAN

6.1 Αξιολόγηση της Εμπειρίας χρήστη μέσω SUS

Το System Usability Scale (SUS) είναι ένα απλό, αλλά αποτελεσματικό εργαλείο για τη μέτρηση της χρηστικότητας ενός συστήματος, προϊόντος ή υπηρεσίας. Δημιουργήθηκε το 1986 από τον John Brooke και, από τότε, έχει γίνει ένα από τα πιο δημοφιλή και ευρέως αποδεκτά εργαλεία για την αξιολόγηση της χρηστικότητας.

Το SUS αποτελείται από 10 ερωτήσεις με πέντε επιλογές απάντησης που καλύπτουν διάφορες πτυχές της χρηστικότητας, όπως η ικανοποίηση των χρηστών, η ευκολία χρήσης και η λειτουργικότητα του συστήματος. Οι χρήστες απαντούν στις ερωτήσεις βαθμολογώντας την εμπειρία τους με το σύστημα σε μια κλίμακα από 1 (Απόλυτα Διαφωνώ) έως 5 (Απόλυτα Συμφωνώ).

Η βαθμολογία του SUS υπολογίζεται με έναν συγκεκριμένο τρόπο. Στις ζυγές ερωτήσεις (1, 3, 5, 7, 9) αφαιρείται 1 από την απάντηση του χρήστη, ενώ στις μονές ερωτήσεις (2, 4, 6, 8, 10) αφαιρείται η απάντηση του χρήστη από το 5. Το άθροισμα των αποτελεσμάτων πολλαπλασιάζεται με το 2.5 για να δώσει μια βαθμολογία από 0 έως 100.

Παρόλο που η βαθμολογία κυμαίνεται από 0 έως 100, δεν πρέπει να ερμηνεύεται ως ποσοστό. Γενικά, μια βαθμολογία SUS πάνω από 68 θεωρείται πάνω από τον μέσο όρο και δείχνει καλή χρηστικότητα, ενώ μια βαθμολογία κάτω από 68 υποδηλώνει ότι υπάρχουν περιθώρια βελτίωσης.

Η χρηστικότητα είναι κρίσιμη για την επιτυχία ενός προϊόντος ή υπηρεσίας. Ένα σύστημα που είναι δύσκολο στη χρήση ή δεν ανταποκρίνεται στις ανάγκες των χρηστών μπορεί να οδηγήσει σε μειωμένη υιοθέτηση, αυξημένες καταγγελίες από τους χρήστες και, τελικά, σε αποτυχία του προϊόντος. Το SUS παρέχει μια γρήγορη και αποτελεσματική μέθοδο για την κατανόηση της χρηστικότητας ενός συστήματος και την ταυτοποίηση περιθωρίων βελτίωσης.

Στο παρακάτω γράφημα βλέπουμε ενα δείγμα 20 απαντήσεων στην φόρμα της ιστοσελίδας μας. Για λόγους απλούστευσης του πίνακα αναφέρονται οι ερωτήσεις με τη σειρά που φαίνονται στον πίνακα:

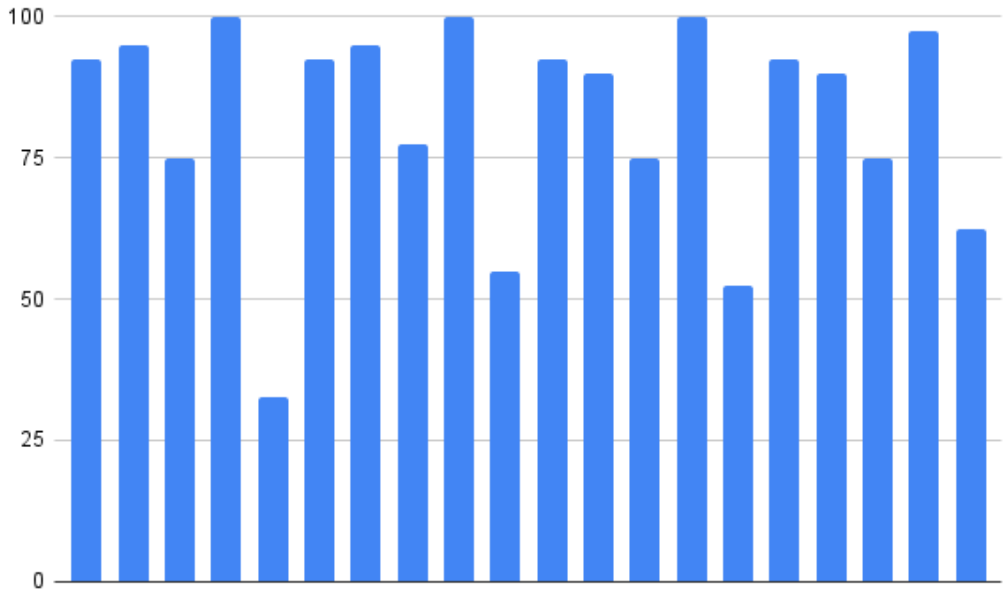
1. I think that I would like to use this system frequently.
2. I found the system unnecessarily complex.
3. I thought the system was easy to use.
4. I think that I would need the support of a technical person to be able to use this system.
5. I found the various functions in this system were well integrated.
6. I thought there was too much inconsistency in this system.
7. I would imagine that most people would learn to use this system very quickly.
8. I found the system very cumbersome to use.
9. I felt very confident using the system.
10. I needed to learn a lot of things before I could get going with this system.

Πίνακας 6.1: Results from the SUS questionnaire

Χρονική σήμανση	Questions										SUM
09/08/2023	3	1	4	1	4	1	5	1	5	1	90
09/08/2023	4	1	5	2	5	1	5	1	5	2	92.5
11/08/2023	5	1	4	1	4	1	5	1	5	1	95
11/08/2023	4	1	5	2	5	1	5	1	5	2	92.5
12/08/2023	3	2	5	1	4	2	4	3	4	2	75
12/08/2023	4	1	5	1	5	1	5	1	5	1	97.5
13/08/2023	4	3	5	2	3	2	4	4	4	4	62.5
13/08/2023	3	2	5	1	4	2	4	3	4	2	75
15/08/2023	5	1	5	1	5	1	5	1	5	1	100
17/08/2023	3	3	2	2	1	4	2	4	2	4	32.5
19/08/2023	4	1	5	2	5	1	5	1	5	2	92.5
21/08/2023	5	1	4	1	4	1	5	1	5	1	95
21/08/2023	3	2	5	1	4	2	4	3	4	1	77.5
23/08/2023	5	1	5	1	5	1	5	1	5	1	100
24/08/2023	4	3	3	2	3	3	4	4	4	4	55
25/08/2023	4	1	5	2	5	1	5	1	5	2	92.5
25/08/2023	5	1	4	1	4	1	3	1	5	1	90
25/08/2023	3	2	5	1	4	2	4	3	4	2	75
26/08/2023	5	1	5	1	5	1	5	1	5	1	100
26/08/2023	4	3	3	2	4	4	4	4	3	4	52.5

Βλέπουμε οτι κατα μέσο όρο η εφαρμογή αξιολογείται με περίπου 83.375 για την εμπειρία χρήσης.

Μια γραφική αναπαράσταση του Πίνακα 6.1 φαίνεται στο Γράφημα 6.1.



Σχήμα 6.1: Γράφημα Αποτελεσμάτων Ερωτηματολογίου

Κεφάλαιο 7

Συμπεράσματα και Μελλοντικές επεκτάσεις

7.1 Συμπεράσματα

Στην επιστήμη δεδομένων, η συσταδοποίηση αναφέρεται στην οργάνωση των δεδομένων σε ομάδες με βάση την ομοιότητά τους, χωρίς τη χρήση προηγούμενων ετικετών. Ο αλγόριθμος DBSCAN είναι ένας πολύ δημοφιλής αλγόριθμος συσταδοποίησης που λαμβάνει υπόψη την πυκνότητα των σημείων στο χώρο. Αντίθετα με πολλούς άλλους αλγόριθμους, ο DBSCAN δεν απαιτεί τον καθορισμό του αριθμού των ομάδων εκ των προτέρων, πράγμα που τον καθιστά ιδιαίτερα ελκυστικό για πολλές εφαρμογές.

Μέσα από έρευνά, διαπιστώθηκε ότι υπάρχει μια σημαντική έλλειψη σε προσβάσιμες και δωρεάν διαδικτυακές εφαρμογές που να υποστηρίζουν τον DBSCAN για την ευρύτερη επιστημονική κοινότητα. Ως αποτέλεσμα, αναπτύχθηκε μια web εφαρμογή που δίνει τη δυνατότητα στους χρήστες να ανεβάζουν τα δεδομένα τους, να προβάλλουν την καμπύλη ελαχιστοποίησης του K-dist graph για την εκτίμηση των καλύτερων παραμέτρων εισόδου και, τέλος, να εκτελούν συσταδοποίηση με βάση τον DBSCAN. Η εφαρμογή επίσης παρέχει τη δυνατότητα download των αποτελεσμάτων, περιλαμβάνοντας το γράφημα και την αντιστοίχιση των σημείων στις αντίστοιχες συστάδες.

7.2 Μελλοντικές επεκτάσεις

Ασύγχρονη εφαρμογή αλγορίθμου και ειδοποίηση χρήστη
Στο πλαίσιο της συσταδοποίησης, όταν ένας χρήστης επιλέγει να επεξεργαστεί ένα εκτεταμένο σύνολο δεδομένων - π.χ., 1.000.000 εγγραφές - είναι φυσικό να υπολογίζει σημαντικούς χρόνους αναμονής για τη λήψη των αποτελεσμάτων, είτε πρόκειται για την εφαρμογή της μεθόδου προσδιορισμού eps, είτε για την ίδια την διαδικασία της συσταδοποίησης. Για να βελτιώσουμε την εμπειρία του χρήστη και να διευκολύνουμε τη διαδικασία, μία προσθήκη στην εφαρμογή θα μπορούσε να είναι η αυτόματη αποθήκευση των αποτελεσμάτων στον διακομιστή όταν αυτά

χρειάζονται περισσότερο χρόνο για να παραχθούν. Καθώς ολοκληρώνεται η επεξεργασία, ο χρήστης θα λαμβάνει ειδοποίηση στο ηλεκτρονικό του ταχυδρομείο, πληροφορώντας τον ότι η ανάλυση ολοκληρώθηκε και τα αποτελέσματα είναι προσβάσιμα. Έτσι, ο χρήστης μπορεί να επιστρέψει στην ιστοσελίδα όποτε τον βολεύει για να προβάλλει και να αναλύσει τα δεδομένα του.

Συγκριτική ανάλυση

Στην πρόταση "Συγκριτική Ανάλυση" προτείνεται η εισαγωγή ενός χαρακτηριστικού που θα επιτρέπει στο χρήστη να συγκρίνει τα αποτελέσματα διαφόρων εκτελέσεων του DBSCAN με διαφορετικές παραμέτρους. Αυτό είναι ζωτικής σημασίας για την κατανόηση του πώς διαφορετικές παράμετροι επηρεάζουν τα αποτελέσματα της συσταδοποίησης, διευκολύνοντας τον χρήστη στη διαδικασία απόφασης για τις πιο κατάλληλες τιμές. Κατά τη διάρκεια αυτής της διαδικασίας, μετά από κάθε εκτέλεση του DBSCAN, τα αποτελέσματα αποθηκεύονται, επιτρέποντας στον χρήστη να επιλέξει ποιες εκτελέσεις θέλει να συγκρίνει. Αυτές οι συγκρίσεις μπορούν να παρουσιαστούν σε σχηματικές απεικονίσεις, προσφέροντας μια οπτική επισκόπηση των διαφορών. Επιπλέον, το σύστημα μπορεί να παρέχει στατιστικές, οπτικοποιήσεις, καθώς και προτεινόμενες παρατηρήσεις για βελτιώσεις, καθιστώντας το εργαλείο ακόμη πιο πολύτιμο για τον τελικό χρήστη.

Προσαρμογέας για Άλλους Αλγορίθμους

Το χαρακτηριστικό "Προσαρμογέας για Άλλους Αλγορίθμους" αποτελεί μια προοδευτική επέκταση της πλατφόρμας, προσφέροντας τη δυνατότητα υποστήριξης πολλαπλών αλγορίθμων συσταδοποίησης. Αυτή η ευελιξία επιτρέπει στους χρήστες να διερευνούν και να επιλέγουν τον καταλληλότερο αλγόριθμο για το εκάστοτε σετ δεδομένων, δίνοντας τους την ευκαιρία να πραγματοποιήσουν άμεσες συγκρίσεις μεταξύ τους. Αυτός ο προσαρμογέας εμπλουτίζει το περιβάλλον της πλατφόρμας, καθιστώντας την μια ολοκληρωμένη λύση για ερευνητές και επαγγελματίες που αναζητούν ένα εργαλείο που να καλύπτει μια ευρεία γκάμα τεχνικών συσταδοποίησης.

Οδηγός Χρήσης και Εκπαίδευση

Ο "Οδηγός Χρήσης και Εκπαίδευση" παρέχει μια σειρά από βίντεο tutorials και λεπτομερείς οδηγίες που σχεδιάστηκαν για να καθοδηγήσουν τους νέους χρήστες μέσα στην εφαρμογή. Αντιλαμβανόμενοι τη δυσκολία που μπορεί να συναντήσει κάποιος όταν αντιμετωπίζει ένα νέο λογισμικό, η εν λόγω προσέγγιση στοχεύει στην απλοποίηση της μάθησης και στην ενθάρρυνση της αυτονομίας των χρηστών. Μέσω αυτών των εκπαιδευτικών υλικών, οι χρήστες έχουν τη δυνατότητα να εκμεταλλευτούν πλήρως τις δυνατότητες της εφαρμογής, ενισχύοντας την παραγωγικότητά τους και την αποτελεσματικότητα των αναλύσεων τους.

Βιβλιογραφία

[1] M. Ester, H. P. Kriegel, J. Sander, and X. Xu, "A density-based algorithm for discovering clusters in large spatial databases with noise," in *Proceedings of the 2nd International Conference on Knowledge Discovery and Data Mining*, pp. 226–231, AAAI Press, 1996.

[2] R. Harvey, "Explaining clustering: A conceptual framework," *Quality & Quantity*, vol. 24, no. 2, pp. 149–168, 1990.

[3] J. McQueen, "Some methods for classification and analysis of multivariate observations," *Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability*, vol. 1, pp. 281–297, 1967.

[4] A. K. Jain, M. N. Murty, and P. J. Flynn, "Data clustering: a review," *ACM computing surveys (CSUR)*, vol. 31, no. 3, pp. 264–323, 1999.

[5] E. Schubert, J. Sander, M. Ester, H. P. Kriegel, and X. Xu, "Dbscan revisited, revisited: why and how you should (still) use dbscan," *ACM Transactions on Database Systems (TODS)*, vol. 42, no. 3, p. 19, 2017.

[6] U. von Luxburg, "A tutorial on spectral clustering," *Statistics and computing*, vol. 17, no. 4, pp. 395–416, 2007.

[7] J. Shi and J. Malik, "Normalized cuts and image segmentation," *IEEE Transactions on pattern analysis and machine intelligence*, vol. 22, no. 8, pp. 888–905, 2000.

[8] C. Fraley and A. E. Raftery, "Model-based clustering, discriminant analysis, and density estimation," *Journal of the American Statistical Association*, vol. 97, no. 458, pp. 611–631, 2002.

[9] G. J. McLachlan and D. Peel, *Finite mixture models*. John Wiley & Sons, 2000.

[10] F. Hutter, L. Kotthoff, and J. Vanschoren, *Automated Machine Learning: Methods, Systems, Challenges*. Springer, 2019.

[11] H. He and D. Wu, "Automl: A survey of the state-of-the-art," *Knowledge and Information Systems*, vol. 61, no. 1, pp. 1–42, 2019.

[12] J. Han, J. Pei, and M. Kamber, *Data Mining: Concepts and Techniques*. Elsevier, 2011.

[13] A. K. Jain, "Data clustering: 50 years beyond k-means," *Pattern recognition letters*, vol. 31, no. 8, pp. 651–666, 2010.

[14] J. Sander, M. Ester, H. P. Kriegel, and X. Xu, "Density-based clustering in spatial databases: The algorithm gdbscan and its applications," *Data Mining and Knowledge Discovery*, vol. 2, no. 2, pp. 169–194, 1998.

[15] H.-P. Kriegel, P. Kröger, J. Sander, and A. Zimek, "Density-based clustering," *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, vol. 1, no. 3, pp. 231–240, 2011.

[16] K. Beyer, J. Goldstein, R. Ramakrishnan, and U. Shaft, "When is "nearest neighbor" meaningful?," in *International conference on database theory*, pp. 217–235, Springer, 1999.

[17] R. J. Campello, D. Moulavi, A. Zimek, and J. Sander, "Density-based clustering based on hierarchical density estimates," *Pacific-Asia conference on knowledge discovery and data mining*, pp. 160–172, 2013.

[18] M. Ankerst, M. M. Breunig, H.-P. Kriegel, and J. Sander, "Optics: ordering points to identify the clustering structure," in *ACM Sigmod record*, vol. 28, pp. 49–60, ACM, 1999.

[19] R. J. G. B. Campello, D. Moulavi, and J. Sander, "Hierarchical density estimates for data clustering, visualization, and outlier detection," *ACM Transactions on Knowledge Discovery from Data*, vol. 10, pp. 5:1–5:51, 2015.

[20] S. Kisilevich and F. Mansmann, "A grid based dbscan for parallel clustering analysis," in *Advances in Databases and Information Systems*, pp. 20–34, Springer, 2010.

[21] J. Zhang, J. Wang, D. Liu, X. Wang, and X. Wang, "Incorporated dbscan clustering for weighted gene co-expression network analysis," *BMC Bioinformatics*, vol. 20, no. 1, p. 20, 2019.

[22] "Nodejs documentation." `https://nodejs.org/docs/latest-v20.x/api/`. Accessed: Sep 5, 2023.

[23] "Understanding node.js event-driven architecture." `https://www.freecodecamp.org/news/understanding-node-js-event-driven-architecture-223292fcbc2d/`. Accessed: Aug 18, 2023.

[24] G. van Rossum, "Python tutorial," *Python Tutorial*, 1991. Accessed: Aug 16, 2023.

[25] M. Lutz, *Learning Python*. O'Reilly Media, 2013.

[26] "Scikit learn documentation." `https://scikit-learn.org/stable/`. Accessed: Aug 14, 2023.

[27] "Express library documentation." `https://expressjs.com/en/5x/api.html`. Accessed: Sep 11, 2023.

[28] "Jwt introduction." `https://jwt.io/introduction/`. Accessed: Aug 29, 2023.

[29] "Swagger documentation." `https://swagger.io/docs/`. Accessed: Aug 18, 2023.

[30] "Apache friends documentation." https://www.apachefriends.org/docs/. Accessed: Aug 15, 2023.

[31] C. Gackenheimer, *Introduction to React*. USA: Apress, 1st ed., 2015.

[32] "Postman." https://www.postman.com/. Accessed: Aug 15, 2023.

[33] "Tailwind css documentation." https://tailwindcss.com/docs/. Accessed: Aug 18, 2023.

[34] B. Schneier, "Description of a new variable-length key, 64-bit block cipher (blowfish)," *Fast software encryption*, vol. 809, pp. 191–204, 1993.

[35] R. S, "Er diagrams in dbms: Entities relationship diagram model," 2022. Accessed: Aug 16, 2023.