Αναφορά Εργασίας στην Τεχνολογία Λογισμικού

Όνομα: Κωνσταντίνος Καφτεράνης

AM: inf2021090

Σεπτέμβριος 2024

Περιεχόμενα

1 Εισαγωγή

Για την εργασία του μαθήματος Τεχνολογία Λογισμικού, δημιουργήθηκε μια διαδικτυακή εφαρμογή μηχανικής μάθησης για την ανάλυση δεδομένων. Η υλοποίηση πραγματοποιήθηκε με τη γλώσσα προγραμματισμού Python χρησιμοποιώντας τη βιβλιοθήκη Streamlit καθώς και άλλες βιβλιοθήκες μηχανικής μάθησης όπως Scikit-learn, Pandas και Matplotlib. Η εφαρμογή ενσωματώνει διάφορους αλγορίθμους μηχανικής μάθησης για κατηγοριοποίηση και ομαδοποίηση, ενώ προσφέρει δυνατότητες οπτικοποίησης των δεδομένων μέσω γραφημάτων και διαγραμμάτων. Η εφαρμογή παρέχει μια φιλική προς τον χρήστη διεπαφή για τη φόρτωση δεδομένων, την επεξεργασία τους και την εξαγωγή πολύτιμων πληροφοριών μέσω στατιστικής ανάλυσης και οπτικοποίησης.

2 Μεθοδολογία Υλοποίησης - ΚλυκλοςΖωής Λογισμικού

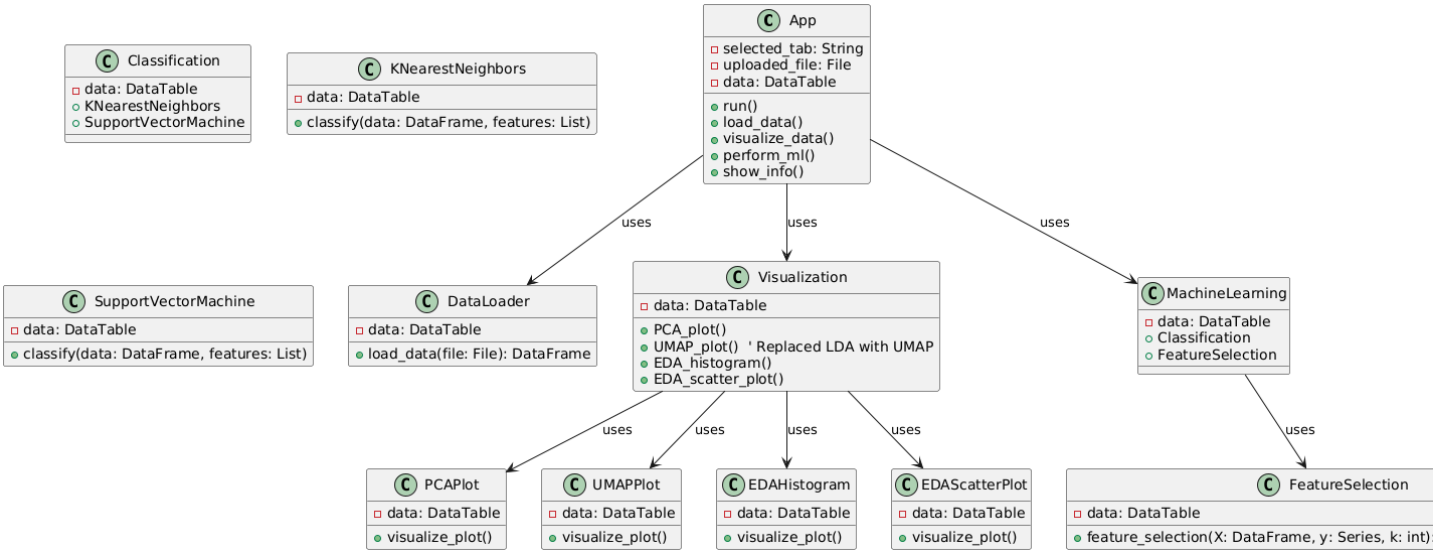
2.1 Μοντέλο Επαναληπτικής Ανάπτυξης (Iterative Model)

Το Μοντέλο Επαναληπτικής Ανάπτυξης είναι μια ευέλικτη προσέγγιση ανάπτυξης λογισμικού που βασίζεται στη συνεχή βελτίωση του λογισμικού μέσα από επαναλαμβανόμενες φάσεις ανάπτυξης.

- Ξεκινάει με τη συλλογή των απαιτήσεων και την ανάλυση των αναγκών του έργου.
- Ο κώδικας γράφεται σε μικρές, διαχειρίσιμες μονάδες, με κάθε μονάδα να δοκιμάζεται ξεχωριστά, δίνοντας έμφαση στη διόρθωση λαθών πριν προχωρήσουμε στην επόμενη ενότητα.
- Η ανάπτυξη γίνεται τμηματικά, με την ολοκλήρωση κάθε μονάδας να συμβάλλει στη συνολική ανάπτυξη του συστήματος.
- Οι τελικές διορθώσεις και βελτιώσεις γίνονται μόνο όταν το κύριο σύστημα είναι σταθερό και λειτουργικό.

Αυτή η προσέγγιση επιτρέπει την γρήγορη διόρθωση λαθών και την καλύτερη διαχείριση των απαιτήσεων που μπορεί να αλλάζουν κατά την ανάπτυξη της εφαρμογής.

3 Δομή Εφαρμογής



Σχήμα 1: Το UML διάγραμμα.

3.1 App

3.1.1 display_data

- Η συνάρτηση display_data εμφανίζει τον πίνακα δεδομένων στην εφαρμογή.
- Συγκεκριμένα, χρησιμοποιεί την st.write για να προβάλλει τα δεδομένα στην πλευρική γραμμή της εφαρμογής.

3.1.2 split_data

- Η συνάρτηση split_data διαχωρίζει τα δεδομένα σε χαρακτηριστικά (X) και στόχο (y).
- Τα χαρακτηριστικά περιλαμβάνουν όλες τις στήλες εκτός από την τελευταία, ενώ η τελευταία στήλη θεωρείται η μεταβλητή στόχος.
- Εμφανίζει τα χαρακτηριστικά και τη μεταβλητή στόχο στην εφαρμογή χρησιμοποιώντας την st.write και επιστρέφει τα X και y.

3.1.3 main

- Η συνάρτηση main είναι η κύρια συνάρτηση της εφαρμογής και διαχειρίζεται την κύρια ροή της εφαρμογής.
- Διαχειρίζεται την πλοήγηση μέσω της sidebar της εφαρμογής και καλεί άλλες συναρτήσεις ανάλογα με την επιλεγμένη ενότητα.
- Η ενότητα Upload Data φορτώνει και εμφανίζει δεδομένα από αρχεία CSV ή Excel.
- Η ενότητα Data Visualization επιτρέπει την απεικόνιση των δεδομένων με διάφορους τύπους γραφημάτων, όπως PCA και UMAP.
- Η ενότητα Machine Learning εκτελεί αλγορίθμους μηχανικής μάθησης, όπως KNN και SVM για ταξινόμηση
- Η ενώτητα Feature Selection έ ή έ έ μό ά μύ όμ
- Η ενότητα Information καλεί τη συνάρτηση display_info για να εμφανίσει πληροφορίες σχετικά με την εφαρμογή.

3.2 ML

3.2.1 Classification

Η κατηγοριοποίηση είναι η διαδικασία της ανάθεσης μιας ετικέτας σε κάθε δείγμα βάσει των χαρακτηριστικών του. Μερικές από τις χρησιμοποιούμενες μεθόδους περιλαμβάνουν:

- K-Nearest Neighbors (KNN): Η συνάρτηση 'knn' υλοποιεί τον αλγόριθμο K-Nearest Neighbors (KNN) για την κατηγοριοποίηση των δεδομένων. Ο αλγόριθμος βασίζεται στους πιο κοντινούς γείτονες ενός δείγματος για να προσδιορίσει την κλάση του.
- Support Vector Machine (SVM): Η συνάρτηση 'svm' χρησιμοποιεί τον αλγόριθμο Support Vector Machine για την κατηγοριοποίηση των δεδομένων. Ο αλγόριθμος λειτουργεί με την εύρεση του υπερ-επίπεδου που διαχωρίζει τα δεδομένα σε διαφορετικές κατηγορίες, με στόχο τη μέγιστη απόσταση μεταξύ των κατηγοριών. Το Σ"Μ μπορεί να επεξεργαστεί και μη γραμμικά δεδομένα μέσω της χρήσης του πυρήνα (κερνελ), επιτρέποντας τη μετατροπή του προβλήματος σε υψηλότερη διάσταση για καλύτερη διαχωριστικότητα.

3.2.2 Visualization

Η οπτικοποίηση των δεδομένων είναι σημαντική για την κατανόηση της δομής και των σχέσεων που υπάρχουν στα δεδομένα. Μερικές από τις διαθέσιμες μεθόδους περιλαμβάνουν:

- PCA (Principal Component Analysis): Η συνάρτηση pca_plot χρησιμοποιεί την PCA για τη μείωση των διαστάσεων των δεδομένων και την απεικόνισή τους σε έναν δισδιάστατο χώρο, διατηρώντας όσο το δυνατόν περισσότερη πληροφορία.
- UMAP (Uniform Manifold Approximation and Projection): Η συνάρτηση umap_plot εφαρμόζει την UMAP για την απεικόνιση των δεδομένων σε δύο διαστάσεις, διατηρώντας τη δομή των δεδομένων και την εγγύτητα των δειγμάτων στον αρχικό πολυδιάστατο χώρο.
- Correlation Heatmap: Η συνάρτηση correlation_heatmap δημιουργεί έναν θερμικό χάρτη που εμφανίζει τις συσχετίσεις μεταξύ των αριθμητικών χαρακτηριστικών ενός συνόλου δεδομένων.
- Box Plot: Η συνάρτηση box_plot δημιουργεί ένα box plot για να εμφανίσει την κατανομή των αριθμητικών δεδομένων και να εντοπίσει πιθανές ακραίες τιμές.
- Pair Plot: Η συνάρτηση pair_plot απεικονίζει ζεύγη μεταβλητών για να εξετάσει τις σχέσεις τους μεταξύ τους.
- Distribution Plot: Η συνάρτηση distribution_plot εμφανίζει την κατανομή των χαρακτηριστικών σε ένα σύνολο δεδομένων, συνδυάζοντας ιστογράμματα και καμπύλες πυκνότητας.

3.2.3 Feature Selection

Feature Selection: Η συνάρτηση feature_selection επιλέγει τα καλύτερα χαρακτηριστικά με βάση το στατιστικό ANOVA F. Χρησιμοποιεί τη μέθοδο SelectKBest από τη βιβλιοθήκη scikit-learn με τη συνάρτηση f_classif για να επιλέξει τα k καλύτερα χαρακτηριστικά από τα δεδομένα εισόδου. Επιστρέφει τα επιλεγμένα χαρακτηριστικά καθώς και τους δείκτες τους.

3.3 Utils

3.3.1 Data Loading

Η συνάρτηση load_data φορτώνει δεδομένα από διάφορους τύπους αρχείων και ελέγχει αν είναι CSV ή Excel. Επιστρέφει τα δεδομένα ή ένα μήνυμα σφάλματος.

3.3.2 Algorithm Comparison

Η συνάρτηση compare_algorithms συγκρίνει τις αποδόσεις των αλγορίθμων K-Nearest Neighbors (KNN) και Support Vector Machine (SVM) με βάση τα μετρικά accuracy, F1 score, ROC-AUC. Η σύγκριση γίνεται αρχικά στα δεδομένα χωρίς μείωση διαστάσεων και, εάν παρέχονται, και στα δεδομένα που έχουν υποστεί επιλογή χαρακτηριστικών.

Η συνάρτηση υπολογίζει ποιο μοντέλο αποδίδει καλύτερα για κάθε μετρικό (π.χ. ακρίβεια, Φ1 σcορε, PO͂-AΥ͂), συγκρίνοντας τα αποτελέσματα των δύο μοντέλων και επισημαίνοντας ποιο από τα δύο έχει την καλύτερη απόδοση. Αντίστοιχα, συγκρίνεται και η απόδοση κάθε αλγορίθμου πριν και μετά την εφαρμογή επιλογής χαρακτηριστικών.

Οι βασικές ενότητες της σύγκρισης περιλαμβάνουν:

- Σύγκριση των αποδόσεων του KNN και του SVM στα πρωτότυπα δεδομένα.
- Σύγκριση των αποδόσεων του KNN και του SVM στα δεδομένα με μειωμένα χαρακτηριστικά, αν υπάρχουν.
- Σύγκριση της απόδοσης του KNN πριν και μετά τη μείωση των χαρακτηριστικών.
- Σύγκριση της απόδοσης του SVM πριν και μετά τη μείωση των χαρακτηριστικών.

Για κάθε σύγκριση, η συνάρτηση ελέγχει ποιος αλγόριθμος αποδίδει καλύτερα ανά μετρικό και εμφανίζει το αποτέλεσμα με τη βοήθεια της βιβλιοθήκης Streamlit, τονίζοντας ποιος αλγόριθμος υπερέχει.

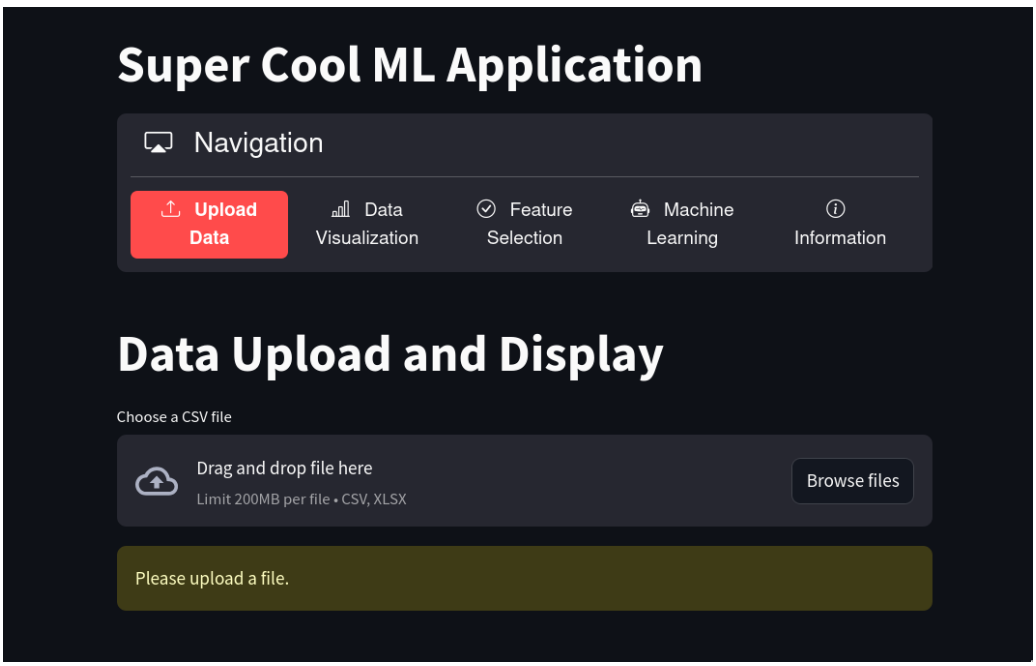
3.3.3 App Information

Πληροφορίες Εφαρμογής

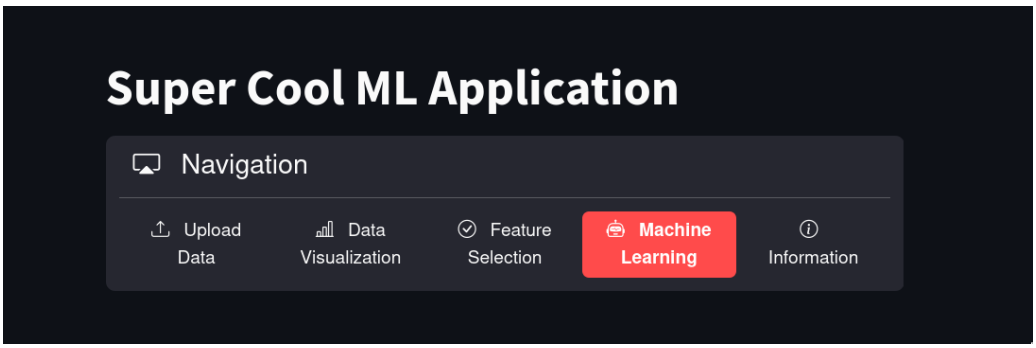
4 Παρουσίαση Εφαρμογής

4.0.1 Navigation

Παρακάτω εξηγείται πώς χρησιμοποιούμε την εφαρμογή. Κατά την είσοδο στην εφαρμογή, ο χρήστης έρχεται σε επαφή με το UI, το οποίο περιλαμβάνει ένα ωιδγετ για ανέβασμα αρχείων, καθώς και ένα οριζόντιο μενού ,το οποίο δημιουργείται έυκολα με την ββλιοθήκη streamlit-option-menu που χρησιμοποιεί για να περιηγηθεί στην εφαρμογή.



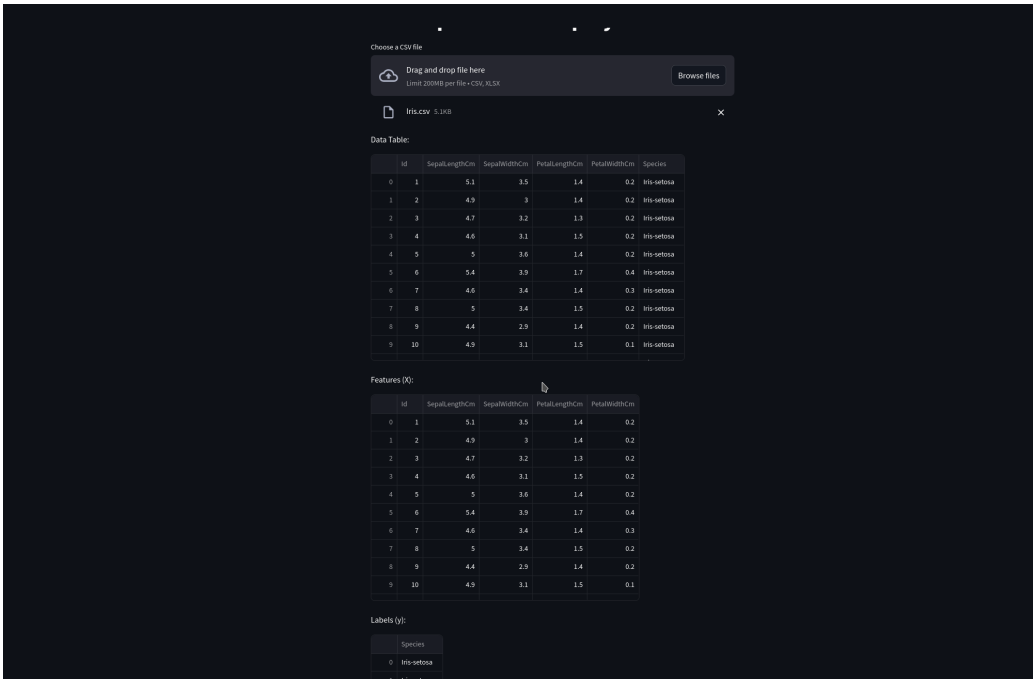
Σχήμα 2: Οθόνη Έναρξης



Σχήμα 3: Μενού

4.0.2 Φόρτωση Δεδομένων

Τα αρχεία δεδομένων πρέπει να είναι της μορφής Excel ή CSV, αλλιώς η εφαρμογή δεν θα επιτρέψει το ανέβασμά τους και θα ζητήσει από τον χρήστη να τα ανεβάσει ξανά.



Σχήμα 4: Φόρτωση Δεδομένων

Αφού το αρχείο ανέβει, εμφανίζονται τα δεδομένα χωρισμένα σε χαρακτηριστικά (features) και ετικέτες.

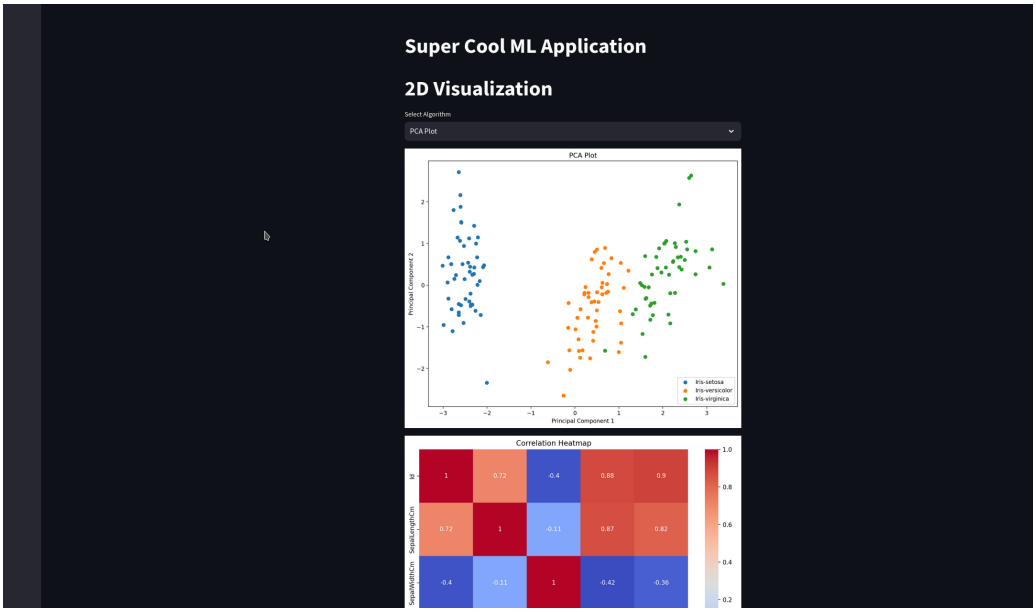
Στο μενού, ο χρήστης μπορεί να επιλέξει την επεξεργασία που θα εφαρμόσει στα δεδομένα: οπτικοποίηση, εφαρμογή μηχανικής μάθησης ή τη σελίδα με τις οδηγίες.

Αν επιστρέψει στο ανέβασμα αρχείων, τότε τα δεδομένα χάνονται και ζητείται να ανέβει ένα καινούργιο αρχείο.

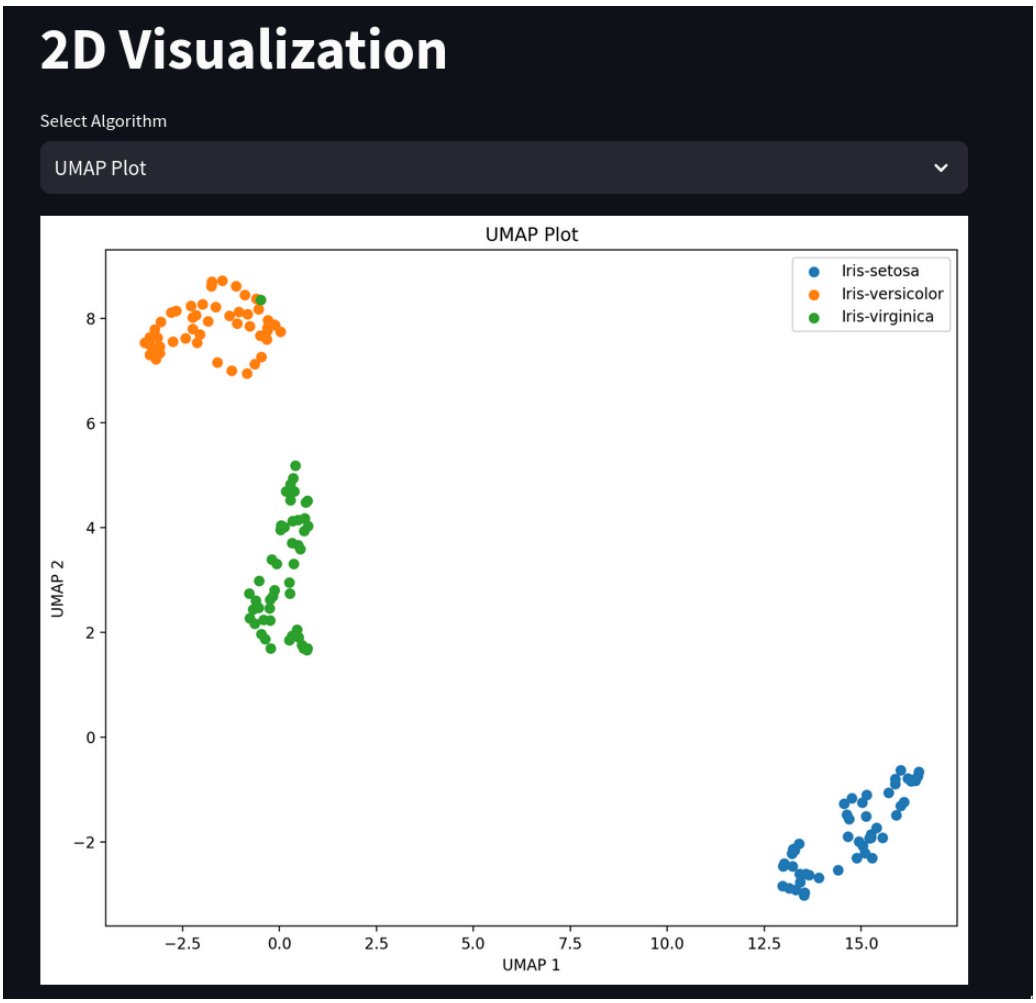
4.0.3 Οπτικοποίση

Επιλέγοντας οπτικοποίηση, ο χρήστης έχει δύο δυνατότητες. Από ένα dropdown menu επιλέγει είτε τον αλγόριθμο PCA ή UMAP και να δημιουργήσει ένα διάγραμμα, καθώς επίσης και μερικά

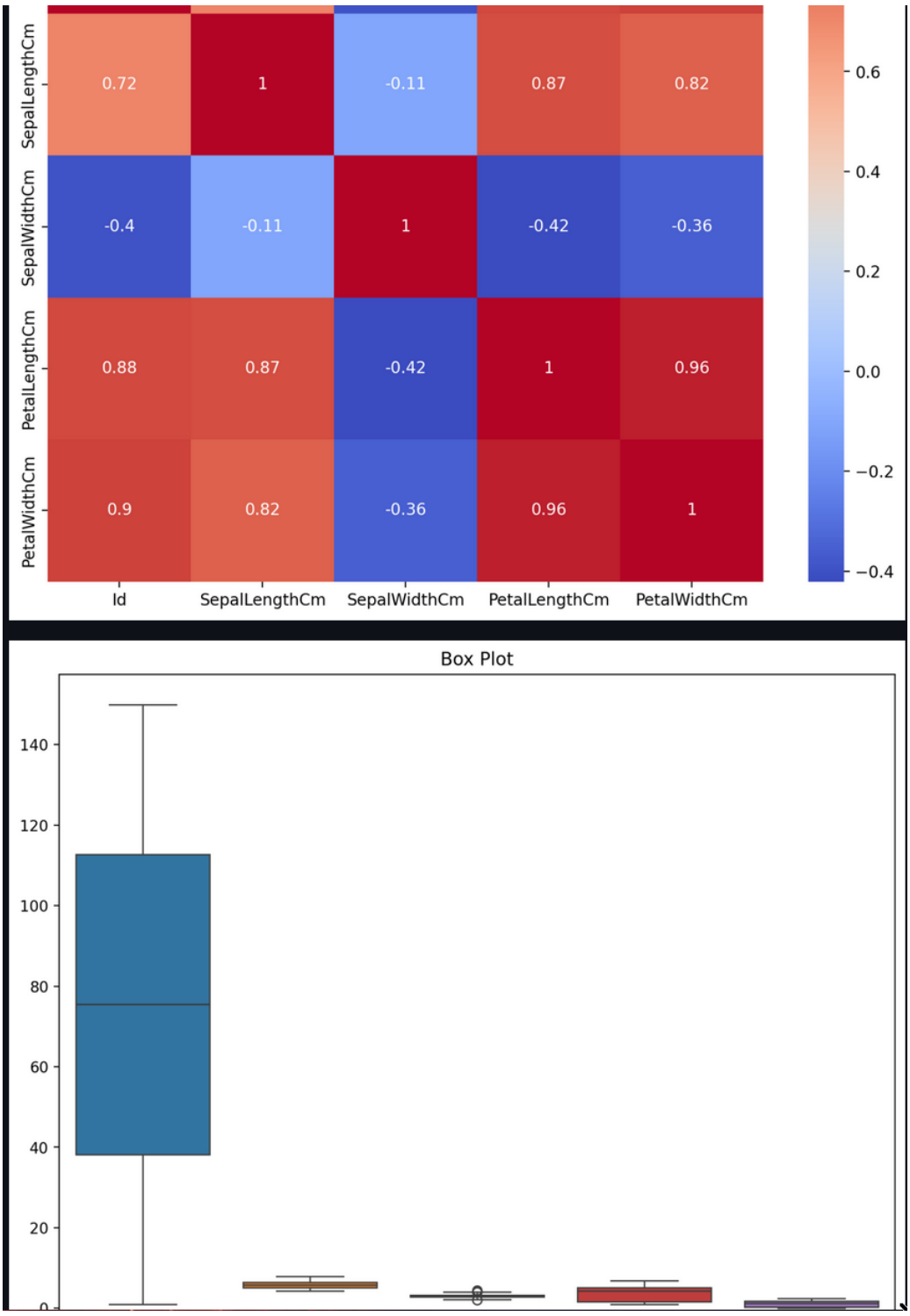
διαγράμματα επεξηγηματικών δεδομένων, τα οποία θα εμφανιστούν κάθετα κάτω από το διάγραμμα του αλγορίθμου οπτικοποίησης. Τα διαγράμματα αυτά είναι:



Σχήμα 5: Οπτικοποίηση Δεδομένων



Σχήμα 6: Ο αλγόριθμος Umap

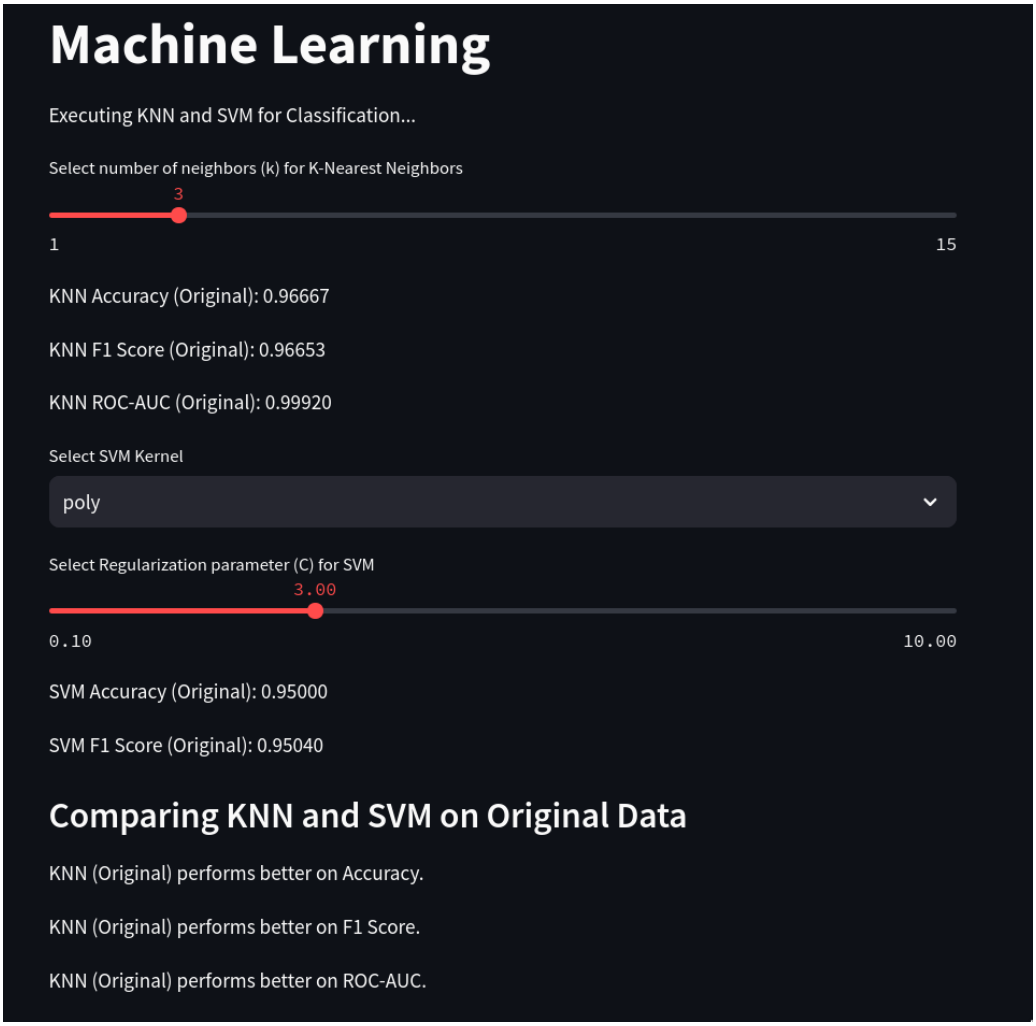


Σχήμα 7: EDA

4.0.4 Classification

Όσον αφορά τη μηχανική μάθηση, ο χρήστης μπορεί να επιλέξει ανάμεσα σε δύο προβλήματα, classification και clustering, τσεκάροντας το αντίστοιχο κυκλάκι. Στην περίπτωση της κατηγοριοποίησης, ο χρήστης επιλέγει μεταξύ των αλγορίθμων k-nearest neighbors (KNN) και support vector machine (SVM). Ο χρήστης χρησιμοποιεί μια μπάρα για να επιλέξει την τιμή του παραμέτρου k για το KNN και οι δύο αλγόριθμοι εφαρμόζονται στα δεδομένα. Τα αποτελέσματα

αξιολογούνται με βάση τους δείκτες accuracy, ROC-AUC, και F1 score.

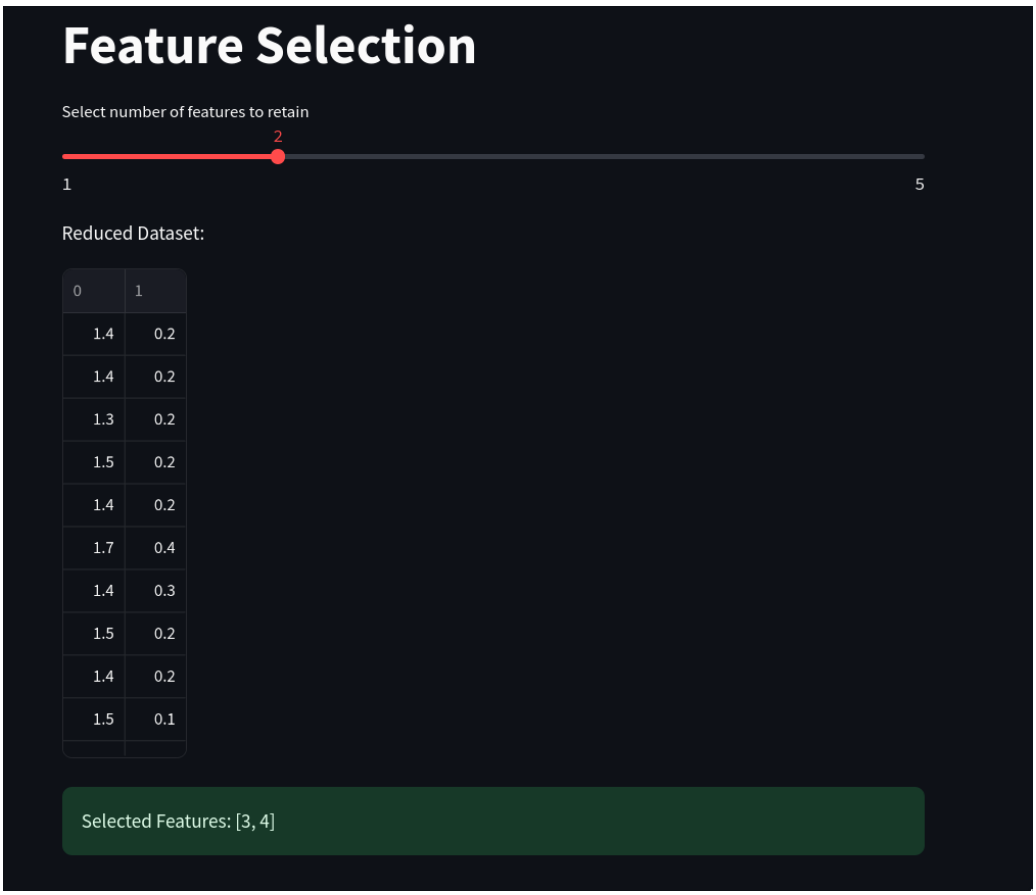


Σχήμα 8: KNN and SVM

Τα αποτελέσματα συγκρίνονται, και ο αλγόριθμος που αποδίδει καλύτερα στους περισσότερους δείκτες αναδεικνύεται ως η καλύτερη επιλογή. Συχνά, το SVM έχει καλύτερη απόδοση από το KNN για δεδομένα υψηλών διαστάσεων, ειδικά όταν το KNN υπερπροσαρμόζεται (overfits). Εάν τα αποτελέσματα είναι παρόμοια, εμφανίζεται μήνυμα που προειδοποιεί τον χρήστη να εξετάσει περαιτέρω ποιος αλγόριθμος είναι κατάλληλος για το συγκεκριμένο σύνολο δεδομένων.

4.0.5 Feature Selection

Η επιλογή χαρακτηριστικών είναι μια κρίσιμη διαδικασία στη μηχανική μάθηση, καθώς επιτρέπει τη μείωση της διάστασης του προβλήματος, βελτιώνοντας την αποδοτικότητα και την απόδοση των μοντέλων. Η συνάρτηση feature_selection εφαρμόζει την τεχνική ANOVA F-statistic για να επιλέξει τα πιο σημαντικά χαρακτηριστικά. Πιο συγκεκριμένα, η συνάρτηση χρησιμοποιεί τη μέθοδο SelectKBest από τη βιβλιοθήκη scikit-learn.



Σχήμα 9: Επιλογή Χαρακτηριστικών

Σχήμα 10: Σύγκριση Αλγορύθμων

4.0.6 Comparison Between Results

Η σύγκριση μεταξύ των αλγορίθμων K-Nearest Neighbors (KNN) και Support Vector Machine (SVM) παρουσιάζεται δυναμικά στην εφαρμογή μέσω της πλατφόρμας . Οι μετρικές Accuracy, F1 Score, και ROC-AUC απεικονίζονται με απλό και κατανοητό τρόπο για την ανάλυση των αποτελεσμάτων τόσο στα αρχικά δεδομένα όσο και στα μειωμένα δεδομένα.

Comparing KNN and SVM on Original Data Στην αρχική απεικόνιση, παρουσιάζονται οι επιδόσεις των δύο αλγορίθμων στα αρχικά δεδομένα. Για κάθε μετρική (Accuracy, F1 Score, ROC-AUC), η εφαρμογή εμφανίζει μήνυμα που αναδεικνύει τον αλγόριθμο που υπερτερεί. Για παράδειγμα, αν το SVM υπερτερεί στην Accuracy, εμφανίζεται το μήνυμα: "SVM (Original) performs better on Accuracy". Αυτή η διαδικασία επαναλαμβάνεται για όλες τις μετρικές.

Comparing KNN and SVM on Reduced Data Αν ο χρήστης έχει επιλέξει να μειώσει τα χαρακτηριστικά, η εφαρμογή παρουσιάζει τη σύγκριση των δύο αλγορίθμων στα μειωμένα δεδομένα. Οι μετρικές Accuracy, F1 Score, ROC-AUC εμφανίζονται ξανά, και αναφέρεται ο αλγόριθμος που έχει την καλύτερη απόδοση, π.χ.: "Both KNN (Reduced) and SVM (Reduced) perform equally on Accuracy" ή "SVM (Reduced) performs better on F1 Score".

Original vs Reduced Data Comparison for Each Algorithm Τέλος, η εφαρμογή εμφανίζει τη σύγκριση της απόδοσης του ίδιου αλγορίθμου μεταξύ των αρχικών και μειωμένων δεδομένων. Για παράδειγμα, εάν ο KNN έχει καλύτερη απόδοση στα αρχικά δεδομένα σε σύγκριση με τα μειωμένα, εμφανίζεται το μήνυμα: "KNN (Original) performs better on Accuracy". Αν δεν

υπάρχουν δεδομένα για κάποιες μετρικές, η εφαρμογή ενημερώνει τον χρήστη με το κατάλληλο μήνυμα, π.χ.: "No ROC-AUC available".

5 Εκτέλεση με Docker

- Η εικόνα Python 3.9 Slim χρησιμοποιείται ως βάση.
- Working directory to `/app`.
- Αντιγράφει το αρχείο requirements.txt στο κοντέινερ.
- Κατεβάζει τα πακέτα από requirements.txt caching.
- Αντιγράφει τον φάκελο src/ στο κοντέινερ.
- Exposes port 8501 for the app.
- Η εντολή CMD τρέχει την εφαρμογή Streamlit στην πόρτα 8501, ώστε να είναι προσβάσιμη.

Για την ανάπτυξη και διανομή της εφαρμογής:

- Εκτελέστε την εντολή `docker build -t my-streamlit-app .` για να δημιουργήσετε την εικόνα.
- Εκτελέστε την εντολή `docker run -p 8501:8501 my-streamlit-app` για να τρέξετε την εφαρμογή στη διεύθυνση `http://localhost:8501`.

6 Σύνδεσμοι Κώδικα,Αναφοράς & Διαγράμματος

- Project GitHub Repository
- GitHub Profile
- Source Code
- Report Raw LaTeX File
- Report PDF
- Dockerfile
- UML Diagram
- UML Diagram Code