Ιόνιο Πανεπιστήμιο

Τεχνολογία Λογισμικού

Ομάδα:
Κύπρος Ανδρέου inf2021002
Ελένη Γιαννούχου inf2021041
Παναγιώτης Κλεάνθους inf2021005

Καθηγητής: Βραχάτης Άρης

Αναφορά εργασίας.

Τμήμα Πληροφορικής

Github Link

Abstract

Η παρακάτω αναφορά, περιγραφή την εργασία μας για το μάθημα ≪Τεχνολογίες Λογισμικού≫ του ’ΣΤ εξαμήνου, όπου στην οποία χρησιμοποιήσαμε τους αλγορίθμους μηχανικής μάθησης K-means και Decision tree για την ανάλυση ενός συνόλου δεδομένων από πάσχοντες και μη πάσχοντες καρδιακών παθήσεων. Στην συνέχεια περιγράφεται η διαδικασία με την οποία εργαστήκαμε όπως και τα αποτελέσματα που προέκυψαν απο τους αλγορίθμους.

Contents

List of Figures

Chapter 1

Εισαγωγή

1.1 Στόχος

Στόχος μας σε αυτή την εργασία, ήταν να δημιουργήσουμε μια διαδικτυακή εφαρμογή, όπου ο χρήστης μπορεί να ανεβάσει ένα σύνολο δεδομένων της μορφής (SxF)
- Γραμμές: Αντιπροσωπεύουν τα S δείγματα που αποτελούν το σύνολο δεδομένων.
- Στήλες: Καταγράφουν τα F χαρακτηριστικά που περιγράφουν κάθε δείγμα.
- Μεταβλητή Εξόδου: Η στήλη F+1, η οποία προστίθεται στο τέλος, περιέχει την
ετικέτα (label) για κάθε δείγμα.
Στην συνέχεια, ο χρήστης μπορεί να αναλύσει τα δεδομένα μέσω των αλγορίθμων K-Means και Decision Tree και να συγκρίνει τα αποτελέσματα τους.

1.1.1 Σύνολο δεδομένων

Το σύνολο δεδομένων το οποίο επιλέξαμε για να κάνουμε τις δικές μας δοκιμές αποτελείται από άτομα τα οποία παράσχουν από καρδιοπαθείς και άτομα τα οποία είναι υγιείς. Το σύνολο δεδομένων μας το κατεβάσαμε μέσω της παρακάτω ιστοσελίδας: (Heart Disease dataset)

1.1.2 Μεθοδολογία

Για να δημιουργήσουμε την εφαρμογή μας επιλέξαμε τη ν γλώσσα προγραμματισμού Python λόγο του ότι είμαστε ποιο εξοικειωμένοι με αυτή. Στην συνέχεια, αφού υλοποιήσαμε τους αλγορίθμους ξεχωριστά, με την βοήθεια της βιβλιοθήκης streamlit δημιουργήσαμε το διαδικτυακό γραφικό περιβάλλον της εφαρμογής μας. Έπειτα, μετά την ολοκλήρωση του κώδικα μας προχωρήσαμε στο Dockerization της εφαρμογής.

1.2 Ο ρόλος του κάθε μέλους

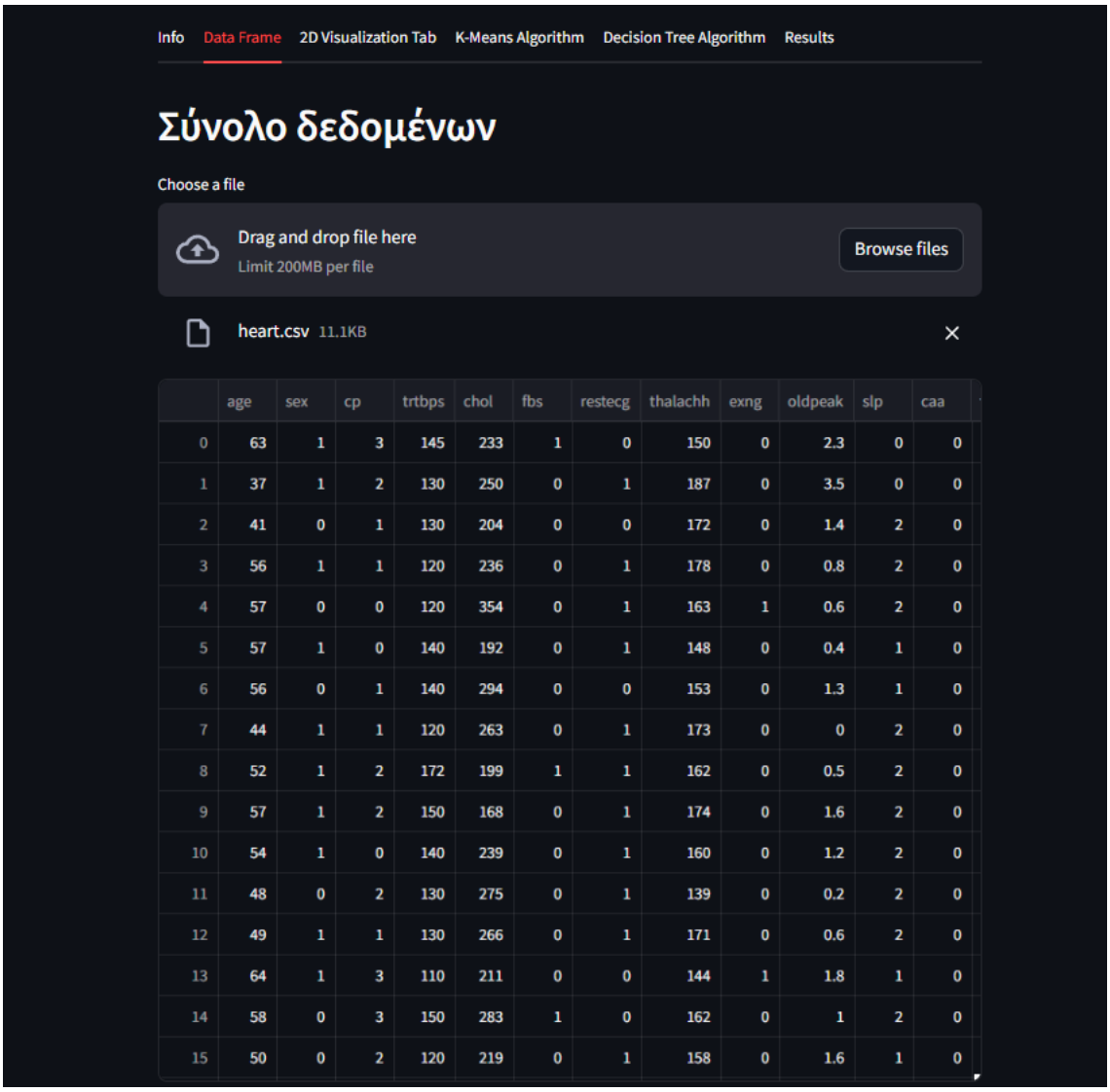
Αρχικά η Ελένη Γιαννόυχου, ασχολήθηκε με την υλοποίηση του αλγορίθμου K-Means, και τη οπτικοποίηση των αποτελεσμάτων του όπως και την οπτικοποίηση του συνόλου δεδομένων με ιστογράμματα και pair plots. Στην συνέχεια, ο Κύπρος Ανδρέου, ασχολήθηκε με την υλοποίησή του αλγορίθμου Decision Tree και την οπτικοποίηση του. Έπειτα, οπτικοποίησε τα δεδομένα με τους αλγορίθμους PCA και TSNE. Τέλος ο Παναγιώτης Κλεάνθους, συνένωσε τα κομμάτια

του κώδικα δημιουργώντας την διαδικτυακή εφαρμογή και προχορόντας στο dockerization της εφαρμογής.

1.2.1 Ο κώδικας της εφαρμογής

Ο κώδικας της εφαρμογής αποτελείται από τις παρακάτω συναρτήσεις:
-KMeans_Algorithm
-DecisionTree_Algorithm
-plot_data_pca
-plot_data_tsne
-plot_histograms
-plot_pair_plots
-main
Η συνάρτηση KMeans_Algorithm παίρνει σαν όρισμα το σύνολο δεδομένων και τον αριθμό των cluster. Εν συνεχεία, δημιουργεί ένα αντίγραφο του συνόλου δεδομένων και "αφαιρεί" την τελευταία στήλη.Τέλος, χωρίζει τα δεδομένα σε train και test, εκπαιδεύει τον αλγόριθμο και οπτικοποιεί τα αποτελέσματα.
Η συνάρτηση DecisionTree_Algorithm, παίρνει σαν όρισμα το σύνολο δεδομένων.Εν συνεχεία, δημιουργεί ένα αντίγραφο του συνόλου δεδομένων,"αφαιρεί" την τελευταία στήλη από τα δείγματά και τα χωρίζει σε χαρακτηριστικά και στόχο. Τέλος, χωρίζει τα δεδομένα σε train και test, εκπαιδεύει τον αλγόριθμο και οπτικοποιεί τα αποτελέσματα.
Η συναρτήσεις plot_data_pca, plot_data_tsne, plot_histograms, plot_pair_plots, παίρνουν σαν όρισμα το σύνολο δεδομένων και το οπτικοποιούν με διαφορετικούς τρόπους.
Στην κύρια συνάρτηση main, αρχικά δημιουργεί τα tabs περιήγησης, όπου στο πρώτο tab(Info), αναγράφονται κάποιες πληροφορίες συνοπτικά για την εφαρμογή και τον τρόπο χρήσης της. Στο δεύτερο tab(Dataframe), ο χρήστης μπορεί να ναναιβάσει το σύνολο δεδομένων του και να δεί τα σχετικά διαγράμματα που εμφανίζονται. Στο τρίτο tab(2D Visualizations) ο χρήστης μπορέι να μελατίσει επίσης τα σχετικά διαγράμματα με τους αλγορίθμους PCA και TSNE. Στο τέταρτο tab(K-Means Algorithm) Η εφαρμογή δίνει την δυνατότητα στον χρήστη να επιλέξει τον αριθμό τον cluster που θέλει και έπειτα να προχωρήσει στην ανάλυση των δεδομένων του με την βοήθεια του αλγορίθμου, K-means. Στο πέμπτο tab(Decision Tree Algorithm) τα δεδομένα αναλύωνται μέσω του αλγορίθμου Decision Tree και παρουσιάζονται τα αποτελέσματα. Τέλος, στο τελευταίο tab ο αναγράφονται τα αποτελέσματα από το σύνολο δεδομένων μας όπως και τα συμπεράσματα μας σχετικά με τους δύο αλγορίθμους.

Figure 1.1: *Information tab*



	age	sex	cp	trtbps	chol	fbs	restecg	thalachh	exng	oldpeak	slp	caa
0	63	1	3	145	233	1	0	150	0	2.3	0	0
1	37	1	2	130	250	0	1	187	0	3.5	0	0
2	41	0	1	130	204	0	0	172	0	1.4	2	0
3	56	1	1	120	236	0	1	178	0	0.8	2	0
4	57	0	0	120	354	0	1	163	1	0.6	2	0
5	57	1	0	140	192	0	1	148	0	0.4	1	0
6	56	0	1	140	294	0	0	153	0	1.3	1	0
7	44	1	1	120	263	0	1	173	0	0	2	0
8	52	1	2	172	199	1	1	162	0	0.5	2	0
9	57	1	2	150	168	0	1	174	0	1.6	2	0
10	54	1	0	140	239	0	1	160	0	1.2	2	0
11	48	0	2	130	275	0	1	139	0	0.2	2	0
12	49	1	1	130	266	0	1	171	0	0.6	2	0
13	64	1	3	110	211	0	0	144	1	1.8	1	0
14	58	0	3	150	283	1	0	162	0	1	2	0
15	50	0	2	120	219	0	1	158	0	1.6	1	0

Figure 1.2: *Dataframe tab*

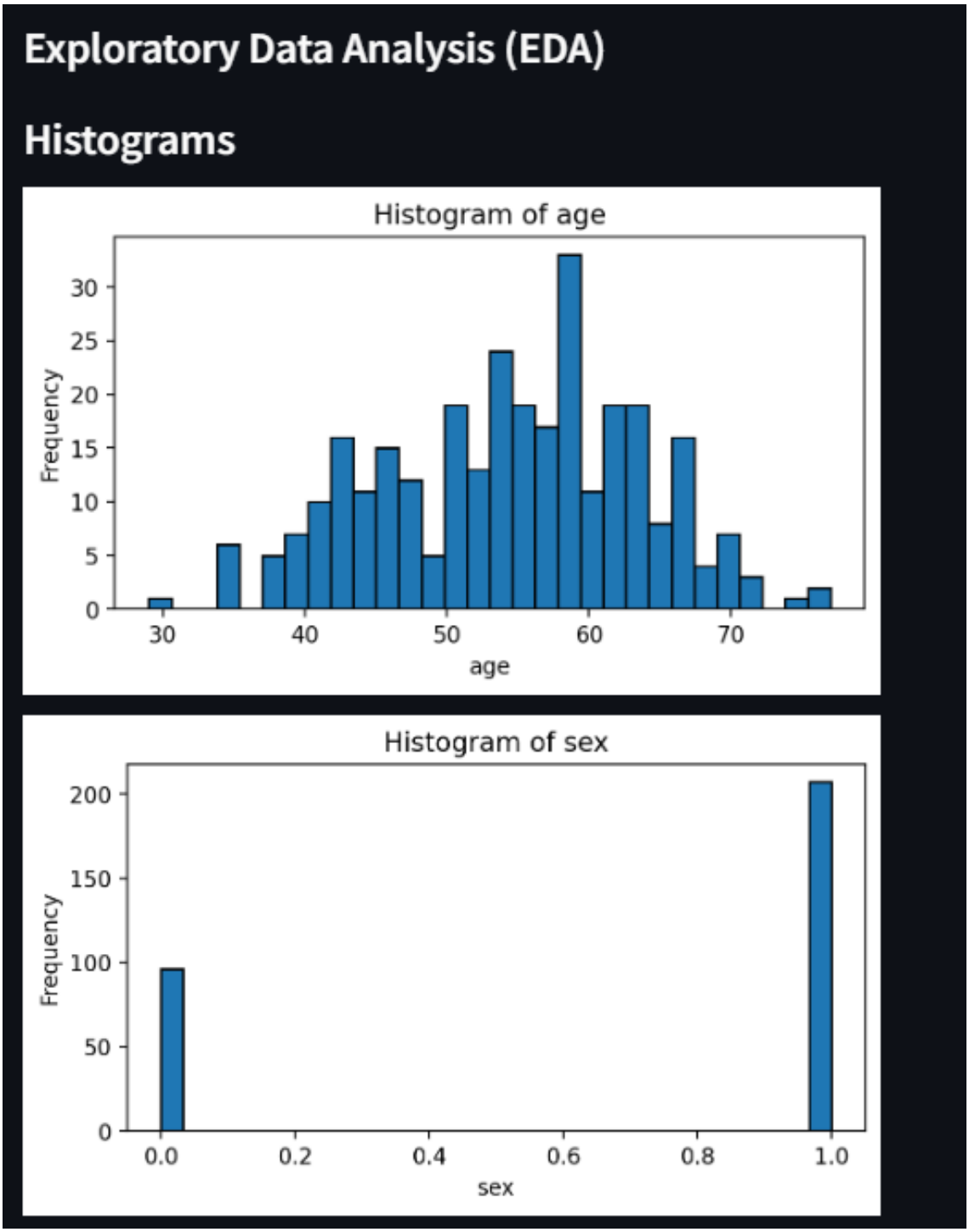


Figure 1.3: *Dataframe tab*

Figure 1.4: *Dataframe tab*

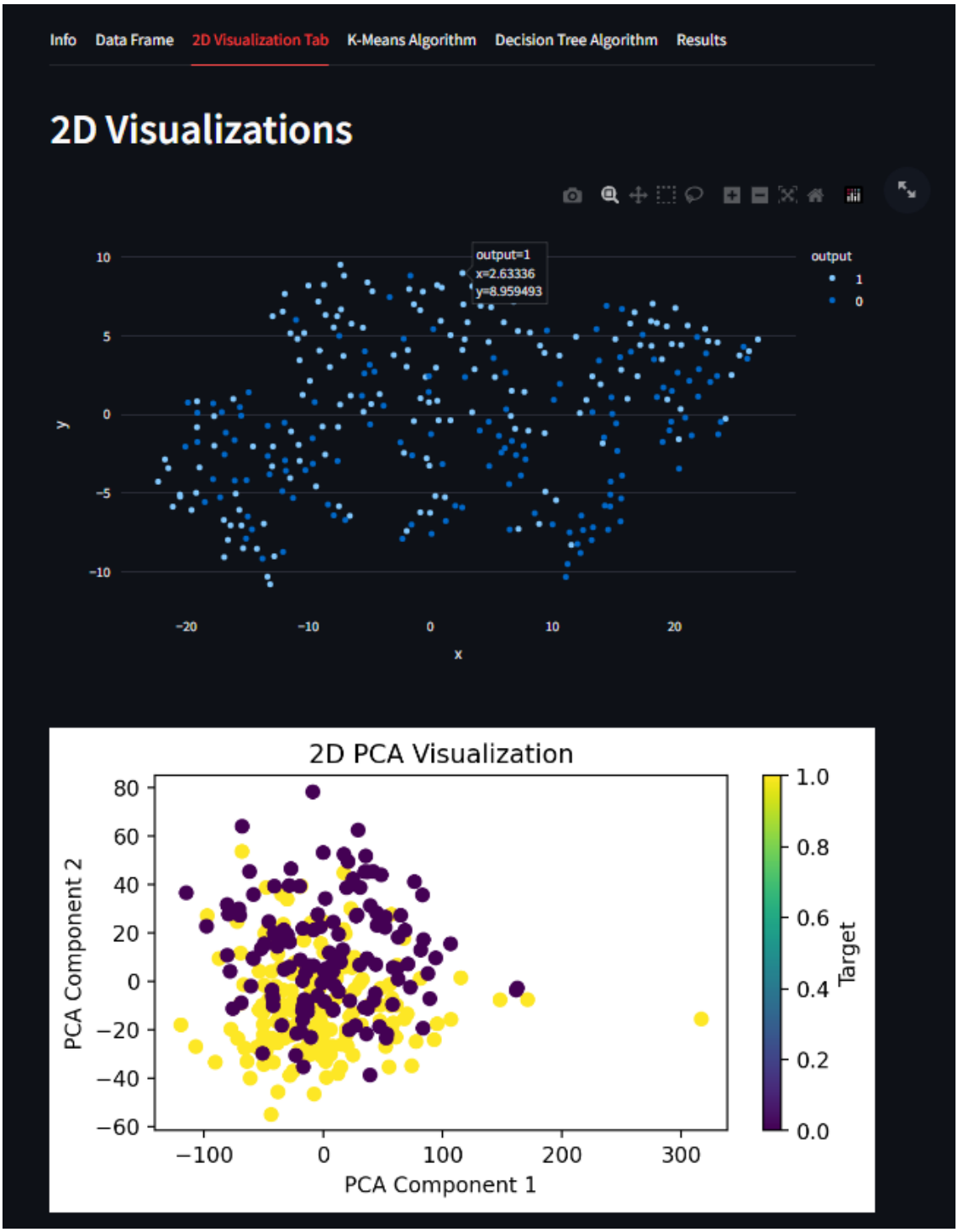


Figure 1.5: *2D Visualizations tab*

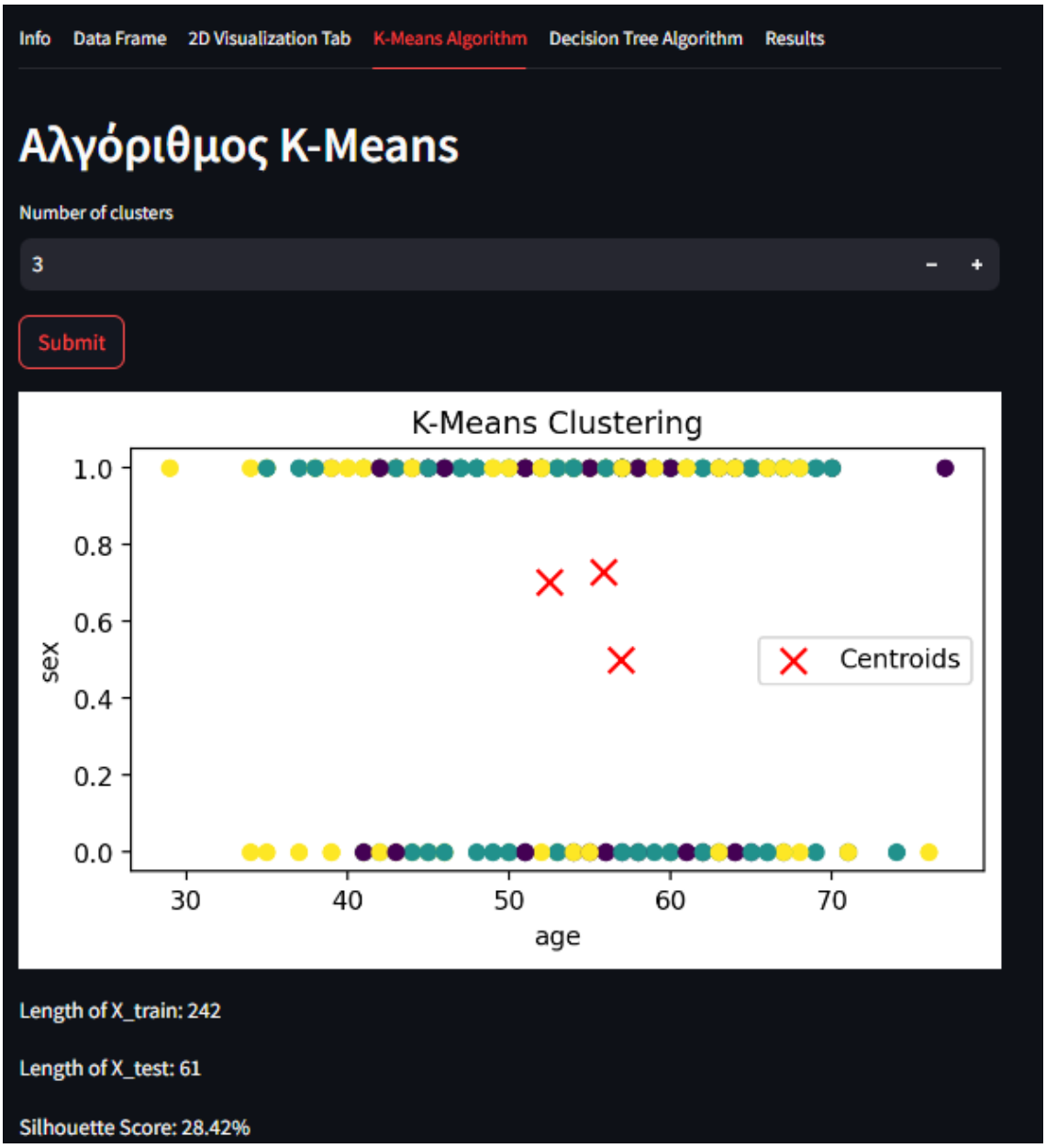


Figure 1.6: *K-means Algorithm tab*

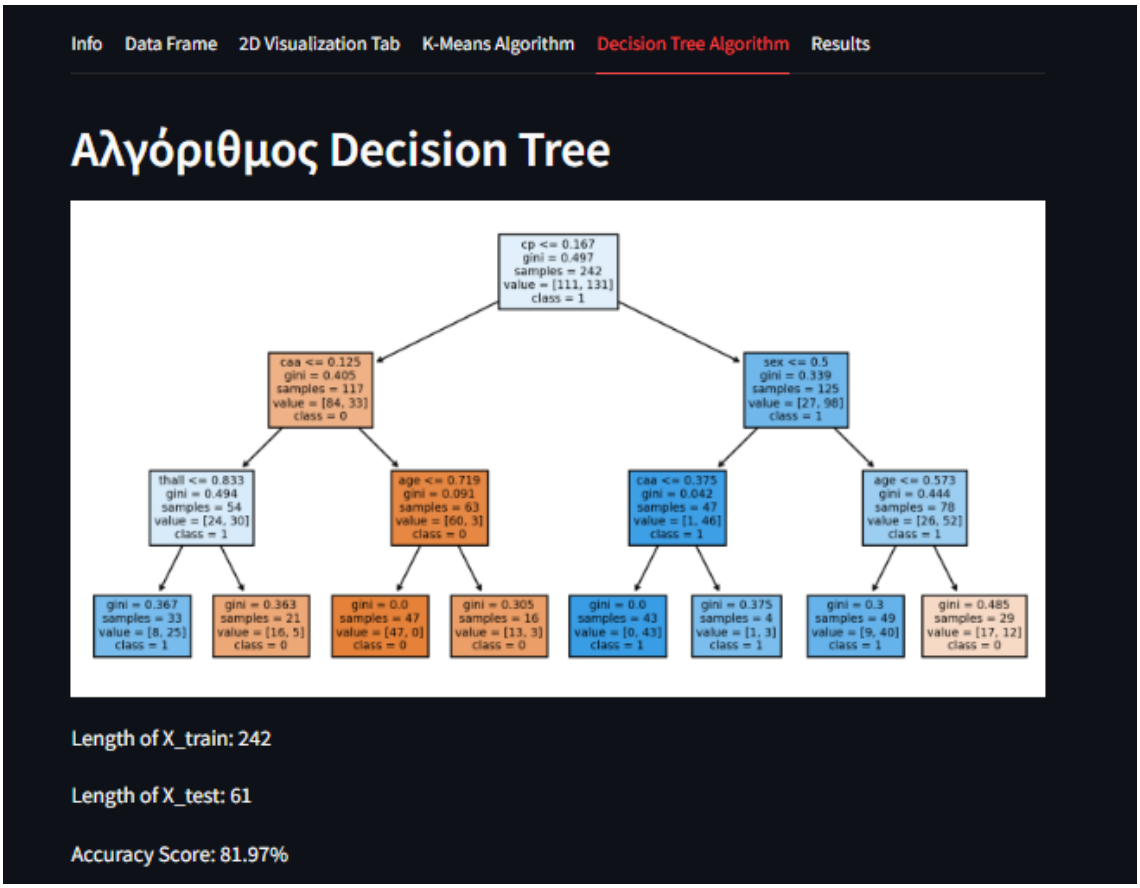


Figure 1.7: *Decision Tree Algorithm tab*

Figure 1.8: *Results tab*

Chapter 2

Αποτελέσματα και σύγκριση

2.1 Πώς λειτουργεί ο αλγόριθμος Decision tree;

Ο αλγόριθμος decision tree αρχικά δημιουργεί ένα μοντέλο που προβλέπει την τιμή στόχο με βάση τους κανόνες από τα χαρακτηριστικά εισόδου του συνόλου δεδομένων.Στην συνέχεια, χωρίζει το σύνολο δεδομένων σε μικρότερα υποσύνολα με βάση τα χαρακτηριστικά, δημιουργώντας ένα δέντρο αποφάσεων.Σε κάθε κόμβο του δέντρου λαμβάνει απόφαση για το πώς να χωρίσει τα δεδομένα χρησιμοποιώντας κριτήρια όπως η εντροπία ή το Gini index.
Χρήσιμος για:

1. Κατηγοριοποίηση και πρόβλεψη με δεδομένα με ετικέτες,
2. Κατανόηση των σχέσεων μεταξύ των μεταβλητών.

2.2 Πώς λειτουργεί ο αλγόριθμος K-means;

Αντίθετα, ο αλγόριμός K-means, χωρίζει τα δεδομένα σε Κ ομάδες, όπου κάθε ομάδα έχει ένα κέντρο. Ο K-means βασίζεται σε μετρήσεις αποστάσεων, συνήθως την ευκλείδεια απόσταση, για να αντιστοιχίσει κάθε δείγμα από το σύνολο δεδομένων στο πλησιέστερο κέντρο. Έτσι, σχηματίζονται οι ομάδες των δεδομένων με βάση τα κοινά χαρακτηριστικά τους.
Χρησιμος για:

1. Ομαδοποίηση χωρίς ετικέτες,
2. Ανακάλυψη ομοιοτήτων και μοτίβων.

2.3 Αποτελέσματα με βάση το σύνολο δεδομένων μας

Για το σύνολο δεδομένων που χρησιμοποιήσαμε, ο αλγόριθμος K-means είχε ποσοστό επιτυχίας περίπου 28% ενώ ο αλγόριθμος Decision tree είχε ποσοστό επιτυχίας περίπου 90%. Για την εκπαίδευση των μοντέλων, χρησιμοποιήθηκαν 242 δείγματα και για τον έλεγχο 61 δείγματα.

2.4 Συμπεράσματα

Με βάση τα αποτελέσματά, ο αλγόριθμος Decision tree είναι ακριβέστερος για την ανάλυση των δεδομένων. Ο λόγος που είναι ποιο αποτελεσματικός, είναι γιατί το σύνολο δεδομένων μας περιείχε δεδομένα από άτομα που έπασχαν από καρδιακά προβλήματά και από μη πάσχων άτομα. Έτσι, ο αλγόριθμος, με την βοήθεια των ετικετών,για την κατανόηση των μοτίβων με τα χαρακτηριστικά των ασθενών μπορεί να καταλήξει σε συμπέρασμα για το αν κάποιος πάσχει από καρδιακά προβλήματα ή όχι. Συμπερασματικά, ο αλγόριθμος Decision tree, είναι καλύτερος για την πρόβλεψη με την χρήση τέτοιου είδους συνόλων δεδομένων με ξεκάθαρα μοτίβα και ετικέτες σε αντίθεσή με τον K-means που δεν χρησιμοποιεί τις ετικέτες για την εκπαίδευση του.

Chapter 3

UML

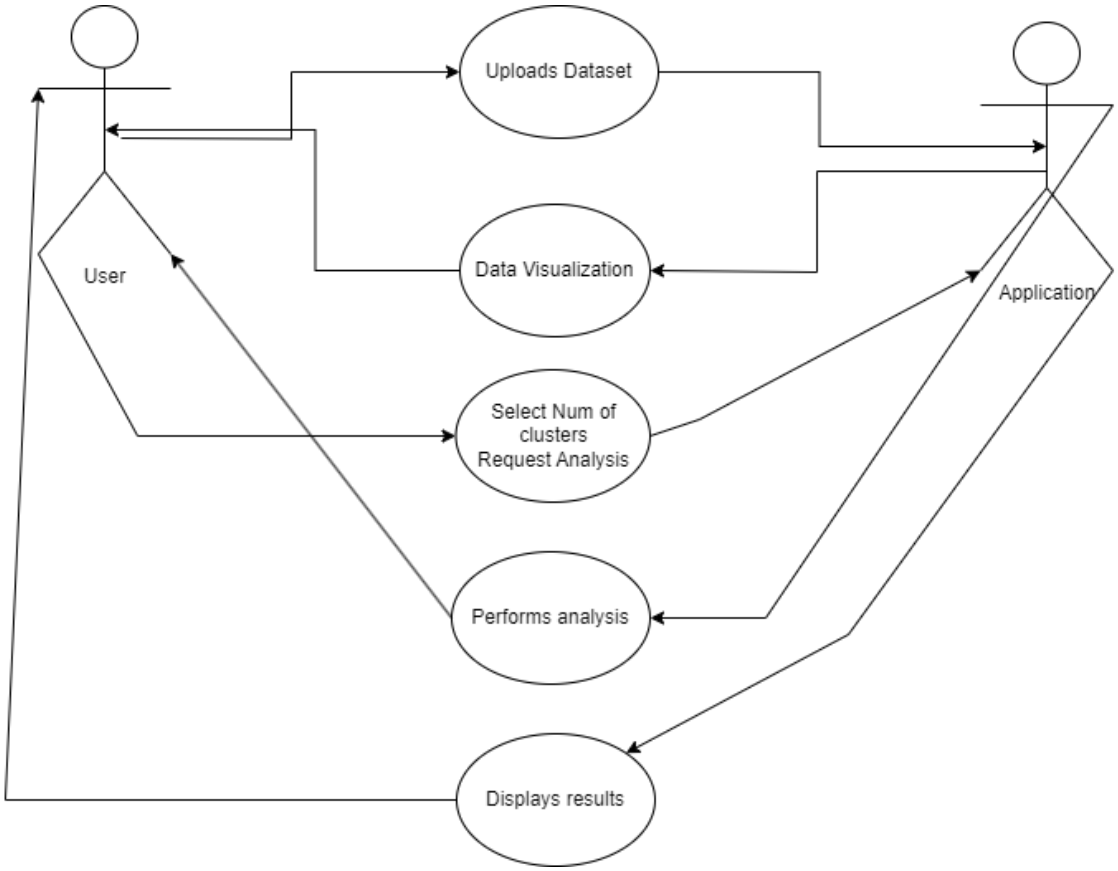


Figure 3.1: *UML diagram*

3.1 Επεξήγηση του διαγράμματος

- Ο χρήστης ανεβάζει το σύνολο δεδομένων
- Η εφαρμογή αναπαριστά τα δεδομένα

- Ο χρήστης επιλέγει τον αριθμό τον clusters και ζητάει ανάλυση των δεδομένων.
- Η εφαρμογή αναλύει τα δεδομένα
- Η εφαρμογή επιστρέφει τα αποτελέσματα στον χρήστη

Chapter 4

Κύκλος ζωής προγράμματος(Waterfall)

4.1 Επεξήγηση του διαγράμματος Καταρράκτη

1)Απαιτήσεις:
Με τον όρο απαιτήσεις εννοούμε την διαδικασία συγκέντρωσης και ανάλυσης των απαιτήσεων του συστήματος-εφαρμογής μας. Για να επιτευχθεί αυτό αρχικά πρέπει να διεξαχθούν συναντήσεις με τους ενδιαφερόμενους για να καταγραφούν οι απαιτήσεις τους. Στην συνέχεια πρέπει να καθοριστούν οι λειτουργίες των απαιτήσεων της εφαρμογής και τέλος Ο καθορισμό των λειτουργικών απαιτήσεων όπως η απόδοση, η ασφάλεια και η ευχρηστία του συστήματος.
2)Σχεδιασμός του συστήματος:
Σε αυτό το στάδιο, επιτυγχάνεται ο σχεδιασμός της εφαρμογής και των επιμέρους τμημάτων της. Αυτό σημαίνει ότι σχεδιάζεται το γραφικό περιβάλλον και γίνεται ο καθορισμός των εργαλείων όπως και ο σχεδιασμό των UML διαγραμμάτων για την απεικόνιση της λειτουργικότητας και της ροής του προγράμματος.
3)Υλοποίηση:
Στο στάδιο αυτό, γίνεται η ανάπτυξη του γραφικού περιβάλλοντος και υλοποιούνται οι λειτουργικότητες του.
4)'Ελεγχος:
Στο στάδιο αυτό, διεξάγονται διάφοροι έλεγχοι για κάθε λειτουργία του προγράμματος.Επίσης, διεξάγονται ολοκληρωμένοι έλεγχοι για την διασφάλισή της ομαλής λειτουργικότητάς και συνεργασίας όλων των τμημάτων της εφαρμογής μεταξύ τους. Τέλος, γίνονται δοκιμές από τους χρήστες για να διασφαλιστεί ότι το σύστημα καλύπτει τις ανάγκες τους.
5)Εγκατάσταση:
Σε αυτό το στάδιο η εφαρμογή εγκαθίσταται είτε στο cloud είτε στις συσκευές των χρηστών, και διασφαλίζεται ότι οι χρήστες μπορούν να έχουν πρόσβαση στην εφαρμογή.
6)Συντήρηση:
Αυτό είναι το τελευταίο βήμα και ένα από τα σημαντικότερα, για τον λόγο ότι σε αυτό το βήμα, παρακολουθούνται οι αναφορές και υπάρχει ανατροφοδότηση από τους χρήστες, για την επιδιόρθωση σφαλμάτων ή για υλοποίηση βελτιώσεων. Τέλος, για την διασφάλιση της ομαλής λειτουργικότητας και την ασφαλή της εφαρμογής γίνονται τακτικές ενημερώσεις των βιβλιοθηκών και των εργαλείων.

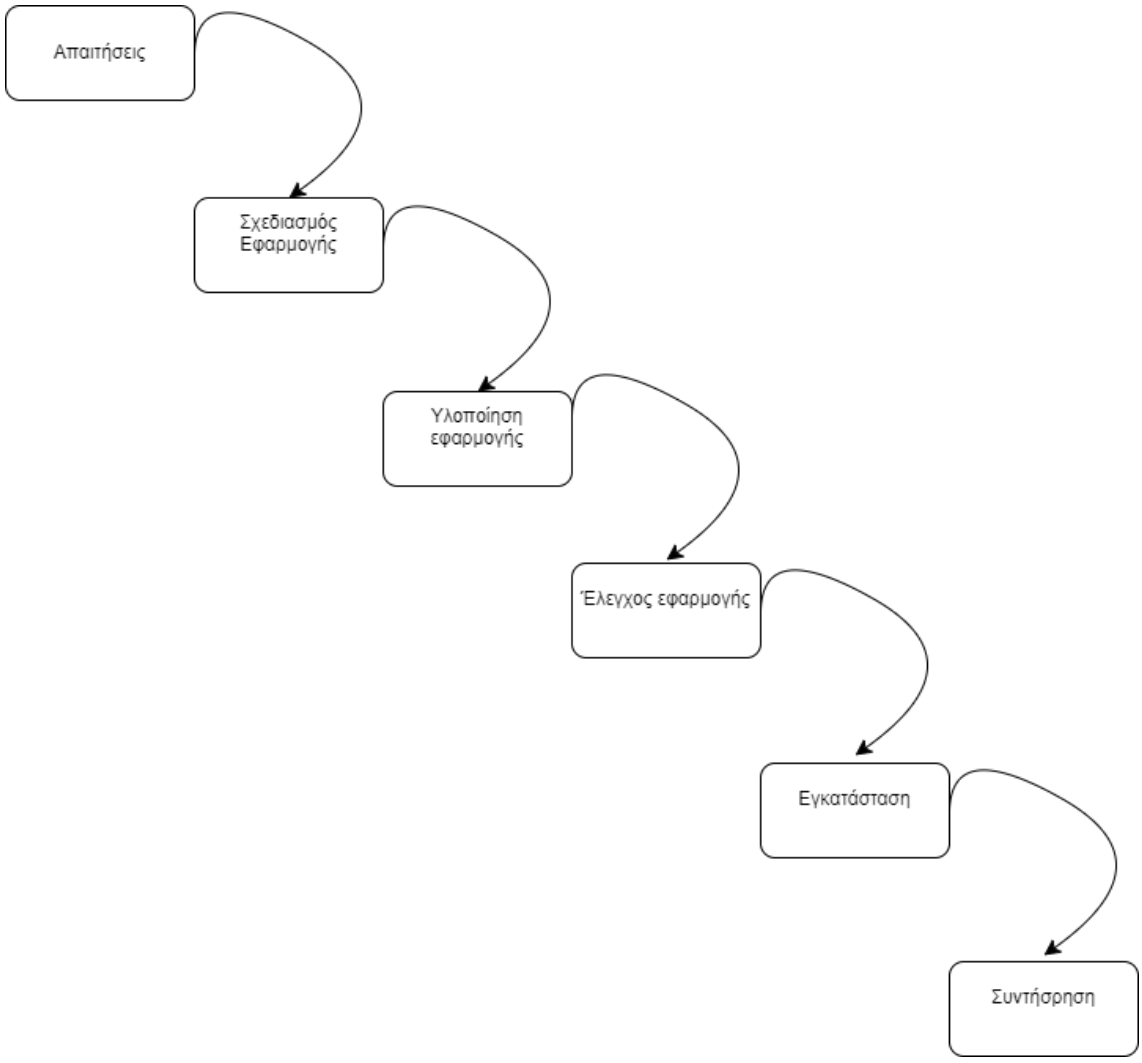


Figure 4.1: *Waterfall model diagram*