
ΠΑΝΕΠΙΣΤΗΜΙΟ
ΔΥΤΙΚΗΣ ΑΤΤΙΚΗΣ

Τμήμα Μηχανικών
Βιομηχανικής Σχεδίασης & Παραγωγής

Διπλωματική Εργασία

# Μηχανική Μάθηση στην Καρδιολογία

από
**Μαρία Γάκη**
ΑΜ: 18389192

Επιβλέπων Καθηγητής:
**ΓΡΗΓΟΡΗΣ ΝΙΚΟΛΑΟΥ**

Αθήνα, ΦΕΒΡΟΥΑΡΙΟΣ, 2023

Εξεταστική Επιτροπή:

| **ΟΝΟΜΑΤΕΠΩΝΥΜΟ** | **ΒΑΘΜΙΔΑ** | **ΨΗΦΙΑΚΗ ΥΠΟΓΡΑΦΗ** |
|---|---|---|
| ΝΙΚΟΛΑΟΥ Γ. | ΛΕΚΤΟΡΑΣ | GRIGORIOS NIKOLAOU<br>Digitally signed by GRIGORIOS NIKOLAOU<br>Date: 2023.03.23 12:13:34 +02'00' |
| ΒΑΣΙΛΕΙΑΔΟΥ Σ. | ΕΠΙΚΟΥΡΗ ΚΑΘΗΓΗΤΡΙΑ | Soultana Vasileiadou<br>Digitally signed by Soultana Vasileiadou<br>Date: 2023.03.23 15:31:37 +02'00' |
| ΔΡΟΣΟΣ Χ. | ΕΔΙΠ | Christos Drosos<br>Digitally signed by Christos Drosos<br>Date: 2023.03.27 22:01:43 +03'00' |

**ΔΗΛΩΣΗ ΣΥΓΓΡΑΦΕΑ ΠΤΥΧΙΑΚΗΣ/ΔΙΠΛΩΜΑΤΙΚΗΣ ΕΡΓΑΣΙΑΣ**

Ο/η κάτωθι υπογεγραμμένος/η Γάκη Μαρία του Νικολάου, με αριθμό μητρώου 18389192 φοιτητής/τρια του Πανεπιστημίου Δυτικής Αττικής της Σχολής Μηχανικών του Τμήματος Βιομηχανικής Σχεδίασης και Παραγωγής, δηλώνω υπεύθυνα ότι:

«Είμαι συγγραφέας αυτής της πτυχιακής/διπλωματικής εργασίας και ότι κάθε βοήθεια την οποία είχα για την προετοιμασία της είναι πλήρως αναγνωρισμένη και αναφέρεται στην εργασία. Επίσης, οι όποιες πηγές από τις οποίες έκανα χρήση δεδομένων, ιδεών ή λέξεων, είτε ακριβώς είτε παραφρασμένες, αναφέρονται στο σύνολό τους, με πλήρη αναφορά στους συγγραφείς, τον εκδοτικό οίκο ή το περιοδικό, συμπεριλαμβανομένων και των πηγών που ενδεχομένως χρησιμοποιήθηκαν από το διαδίκτυο. Επίσης, βεβαιώνω ότι αυτή η εργασία έχει συγγραφεί από μένα αποκλειστικά και αποτελεί προϊόν πνευματικής ιδιοκτησίας τόσο δικής μου, όσο και του Ιδρύματος.

Παράβαση της ανωτέρω ακαδημαϊκής μου ευθύνης αποτελεί ουσιώδη λόγο για την ανάκληση του πτυχίου μου».

Ο/Η Δηλών/ούσα

ΓΑΚΗ ΜΑΡΙΑ

# Περίληψη

Η μηχανική μάθηση αποτελεί έναν ερευνητικό τομέα ο οποίος αναπτύσσεται διαρκώς σύμφωνα με την τεχνολογική εξέλιξη και διεισδύει γρήγορα σε πολλούς τομείς της υγειονομικής περίθαλψης, τόσο στη διάγνωση και την πρόγνωση ασθενειών όσο και την εύρεση νέων φαρμάκων. Η μηχανική μάθηση προσφέρει μεθόδους, τεχνικές και εργαλεία που μπορούν να συμβάλουν στην επίλυση ποικίλων διαγνωστικών και προγνωστικών προβλημάτων στον τομέα της ιατρικής.

Στόχος της συγκεκριμένης εργασίας είναι η ανίχνευση της καρδιακής αρρυθμίας στον ασθενή μέσω μηχανικής μάθησης. Για να επιτευχθεί, αναπτύχθηκαν διάφορα μοντέλα μηχανικής μάθησης με επίβλεψη που χρησιμοποιούνται σε προβλήματα ταξινόμησης. Τα μοντέλα αξιολογήθηκαν βάσει της απόδοσής τους, ώστε να βρεθεί εκείνο που έχει τη βέλτιστη ακρίβεια. Χρησιμοποιήθηκε η βάση δεδομένων του Πανεπιστημίου Chapman και του Λαϊκού Νοσοκομείου Shaoxing, η οποία περιέχει τα αποτελέσματα των σημάτων που προήλθαν από ηλεκτροκαρδιογραφήματα, αφού έχει αφαιρεθεί ο θόρυβος των σημάτων.

## Λέξεις κλειδιά

Καρδιολογία, Καρδιακή Αρρυθμία , Ταξινόμηση, Ηλεκτροκαρδιογράφημα, Μηχανική Μάθηση

# Abstract

Machine learning is a research area, which is developing continuously in line with technological evolution and is rapidly influencing many areas of healthcare, including diagnosis and prognosis of diseases as well as developing new drugs. Machine learning offers methods, techniques and tools that can help solve a variety of diagnostic and prognostic problems in the field of Medicine.

The aim of this research is to detect cardiac arrhythmia in the patient through machine learning. To achieve this, several supervised machine learning models used in classification problems were developed. The models were evaluated based on their performance in order to find the one that has the optimal accuracy. The database of Chapman University and Shaoxing people's Hospital was used, which contains the results of signals that came from electrocardiograms, after denoising them.



## Key Words

Cardiology, Cardiac Arrhythmia, Classification, Electrocardiogram, Machine Learning

# Ευχαριστίες

Αρχικά, θα ήθελα να ευχαριστήσω τον επιβλέποντα καθηγητή μου , κύριο Νικολάου Γρηγόριο, για τη συνεχή υποστήριξη και καθοδήγηση για τη διεκπεραίωση της διπλωματικής μου εργασίας.

Επιπλέον, ευχαριστώ την οικογένειά μου, για την υποστήριξη και την ενθάρρυνση που μου έχουν προσφέρει κατά την διάρκεια των προπτυχιακών σπουδών μου.

# Πίνακας Περιεχομένων

Περίληψη ..... 4
Λέξεις κλειδιά ..... 4
Abstract ..... 5
Key Words ..... 5
Ευχαριστίες ..... 6
Πίνακας Περιεχομένων ..... 7
Κατάλογος Εικόνων ..... 9
Κατάλογος Διαγραμμάτων ..... 9
Εισαγωγή ..... 10
Κεφάλαιο 1 : Μηχανική μάθηση ..... 11
1.1 Εισαγωγή ..... 11
1.2 Μηχανική Μάθηση με επίβλεψη (Supervised) ..... 12
1.3 Μηχανική Μάθηση χωρίς επίβλεψη (Unsupervised) ..... 13
1.4 Ημιεπιβλεπόμενη Μηχανική Μάθηση (Semi-Supervised) ..... 13
1.5 Ενισχυτική Μάθηση (Reinforcement) ..... 14
1.6 Τεχνικές Ensemble ..... 15
Κεφάλαιο 2 : Καρδιολογία και καρδιαγγειακές αρρυθμίες ..... 16
2.1 Εισαγωγή ..... 16
2.2 Ανατομία καρδιάς ..... 17
2.3 Ηλεκτροκαρδιογράφημα ..... 18
2.4 Καρδιαγγειακές Αρρυθμίες ..... 20
Κεφάλαιο 3 : Μέσα υλοποίησης ..... 22
3.1 Αλγόριθμοι ..... 22
3.1.1 Δέντρα αποφάσεων ..... 22
3.1.2 Random Forest ..... 23
3.1.3 K-Γείτονες ..... 24
3.1.4 Extreme Gradient Boosting Tree ..... 25
3.1.5 LightGBM ..... 26
3.2 Περιβάλλον υλοποίησης ..... 27
3.2.1 Jupyter Notebook ..... 27

3.3 Βιβλιοθήκες ..... 28
3.3.1 Numpy ..... 28
3.3.2 Pandas ..... 28
3.3.3 Matplotlib ..... 28
3.3.4 Sklearn ..... 29
3.3.5 Xgboost ..... 29
3.3.6 Lightgbm ..... 29
3.3.7 Imblearn ..... 29
Κεφάλαιο 4: Υλοποίηση ..... 30
4.1 Βάση δεδομένων ..... 30
4.1.1 Δεδομένα και χαρακτηριστικά ..... 30
4.1.2 Επεξεργασία δεδομένων ..... 31
4.1.2.1 Διαχωρισμός Gender ..... 31
4.1.2.2 Διαχωρισμός Beat ..... 31
4.1.2.3 Ομαδοποίηση Rhythm ..... 32
4.2 Μεθοδολογία ..... 34
4.3 Αποτελέσματα και Συγκρίσεις Μοντέλων ..... 35
4.3.1 Αποτελέσματα Μοντέλων ..... 36
4.3.2 Επιλογή Μοντέλων ..... 41
4.3.3 Τελικά Μοντέλα ..... 43
Επίλογος ..... 45
Παράρτημα Α: Ακρωνύμια και συντομογραφίες ..... 46
Παράρτημα Β: Κώδικας Python εύρεσης καρδιακής αρρυθμίας ..... 47
Βιβλιογραφία ..... 59

# Κατάλογος Εικόνων

Εικόνα 1: Υποκατηγορίες Μηχανικής Μάθησης ..... 11
Εικόνα 2: Μηχανική Μάθηση με Επίβλεψη ..... 12
Εικόνα 3: Μηχανική Μάθηση χωρίς Επίβλεψη ..... 13
Εικόνα 4: Τεχνικές Ensemble ..... 15
Εικόνα 5: Ανατομία Καρδιάς ..... 17
Εικόνα 6: Χαρακτηριστικά Στοιχεία ΗΚΓ ..... 19
Εικόνα 7: Ταξινόμηση Πιθανότητας Βροχής Βάσει Καιρικών Χαρακτηριστικών ..... 22
Εικόνα 8: Σύγκριση Δέντρου Απόφασης και Random Forest ..... 23
Εικόνα 9: Ταξινόμηση με 3-κοντινότερους γείτονες ..... 24
Εικόνα 10: Ταξινόμηση με 3-κοντινότερους γείτονες ..... 25
Εικόνα 11: Σύγκριση Ανάπτυξης XGBoost και LightGBM ..... 27
Εικόνα 12: Ονομασίες και Ακρώνυμα Παθήσεων ..... 33
Εικόνα 13: 10- fold validation ..... 34
Εικόνα 14: Τυπολόγιο μετρικών επιδόσεων ..... 35
Εικόνα 15: Πίνακας σύγχυσης LightGBM ..... 36
Εικόνα 16: Αναφορά Ταξινόμησης LightGBM ..... 36
Εικόνα 17:Πίνακας σύγχυσης Δέντρου Αποφάσεων ..... 37
Εικόνα 18: Αναφορά Ταξινόμησης Δέντρου Αποφάσεων ..... 37
Εικόνα 19: Πίνακας σύγχυσης Random Forest ..... 38
Εικόνα 20: Αναφορά Ταξινόμησης Random Forest ..... 38
Εικόνα 21: Πίνακας σύγχυσης K-Γειτόνων ..... 39
Εικόνα 22: Αναφορά Ταξινόμησης K-Γειτόνων ..... 39
Εικόνα 23: Πίνακας σύγχυσης XGBoost ..... 40
Εικόνα 24: Αναφορά Ταξινόμησης XGBoost ..... 40

# Κατάλογος Διαγραμμάτων

Διάγραμμα 1: Κατανομή Αρχικών Κατηγοριών ..... 32
Διάγραμμα 2: Ομαδοποίηση Κατηγοριών ..... 33
Διάγραμμα 3: F1-Score και Recall Μοντέλων ..... 41
Διάγραμμα 4: Recall Μοντέλων ανά κατηγορία ..... 42
Διάγραμμα 5: Επιδόσεις Τελικών Μοντέλων ..... 43
Διάγραμμα 6: Recall Τελικών Μοντέλων ανά κατηγορία ..... 44

# Εισαγωγή

Η μηχανική μάθηση είναι ένας ταχύτατα αναπτυσσόμενος τομέας που έχει τη δυνατότητα να ενισχύσει και εξελίξει τον τρόπο διάγνωσης και θεραπείας των καρδιαγγειακών παθήσεων. Αποτελεί το πεδίο τεχνητής νοημοσύνης το οποίο επικεντρώνεται στην ανάπτυξη αλγορίθμων και μοντέλων που μπορούν να εκπαιδευτούν από μεγάλες ποσότητες δεδομένων, τα οποία ύστερα χρησιμοποιούνται για να κάνουν προβλέψεις ή να λαμβάνουν αποφάσεις.

Γενικότερα στην ιατρική, η μηχανική μάθηση μπορεί να χρησιμοποιηθεί για την ανάλυση μεγάλων ποσοτήτων ιατρικών δεδομένων, τον εντοπισμό παραγόντων κινδύνου και συσχετισμό τους με την εκάστοτε πάθηση, διευκολύνοντας τη διάγνωση και αντιμετώπισή της. Αξιοποιώντας τις κατάλληλες μεθόδους μηχανικής μάθησης, το ιατρικό προσωπικό μπορεί να λαμβάνει πιο ενημερωμένες αποφάσεις σχετικά με τη θεραπεία των ασθενών σε πιο σύντομο χρονικό διάστημα.

Συγκεκριμένα για τις καρδιαγγειακές παθήσεις, η μηχανική μάθηση μπορεί να χρησιμοποιηθεί για την ανάλυση δεδομένων ασθενών, όπως λόγου χάρη δημογραφικές πληροφορίες και ιατρικό ιστορικό, με σκοπό τον εντοπισμό ατόμων με υψηλό κίνδυνο εμφάνισης καρδιαγγειακών παθήσεων. Επιπλέον, μπορεί να βοηθήσει τους ιατρούς να λαμβάνουν αποφάσεις σχετικά με την επιλογή της κατάλληλης θεραπείας με βάση τα μοναδικά χαρακτηριστικά του ασθενούς.

Συνεπώς, οι αλγόριθμοι μηχανικής μάθησης μπορούν να συμβάλλουν στη βελτίωση των αποτελεσμάτων των καρδιαγγειακών παθήσεων, προσφέροντας τη δυνατότητα για έγκαιρη διάγνωση, εξατομικευμένα σχέδια θεραπείας ανάλογα τις ανάγκες του ασθενή και αποτελεσματικότερη παρακολούθηση και αντιμετώπιση των καρδιαγγειακών παθήσεων. Τα παραπάνω δύνανται να επεκταθούν γενικότερα και στον τομέα της υγειονομικής περίθαλψης.

Ως προς τη δομή, η παρούσα εργασία χωρίζεται σε πέντε διαφορετικά κεφάλαια. Τα δύο πρώτα κεφάλαιο αποτελούν το θεωρητικό υπόβαθρο. Το πρώτο κεφάλαιο πραγματεύεται τι είναι η μηχανική μάθηση, εξηγώντας βασικές έννοιες και υποκατηγορίες στις οποίες χωρίζεται. Στο δεύτερο κεφάλαιο αναλύονται βασικές αρχές και όροι της καρδιολογίας και πιο συγκεκριμένα οι καρδιαγγειακές αρρυθμίες.

Ύστερα, ακολουθεί το τρίτο κεφάλαιο που λειτουργεί ως γέφυρα μεταξύ του θεωρητικού και του πρακτικού μέρους. Στο κεφάλαιο αυτό γίνεται λόγος για τα μέσα υλοποίησης της εφαρμογής πρόβλεψης καρδιαγγειακής πάθησης, δηλαδή τους αλγορίθμους, τις βιβλιοθήκες και το περιβάλλον που χρησιμοποιήθηκαν.

Έπειτα, το τέταρτο και το πέμπτο κεφάλαιο απαρτίζουν το πρακτικό μέρος. Στο τέταρτο κεφάλαιο αναπτύσσεται η εφαρμογή. Αναλύεται η βάση δεδομένων που χρησιμοποιήθηκε, η επεξεργασία των δεδομένων και η μεθοδολογία που ακολουθήθηκε. Επιπλέον, στο κεφάλαιο αυτό παρουσιάζονται τα αποτελέσματα των μοντέλων, τα οποία συγκρίνονται ως προς τις επιδόσεις τους. Τέλος, στο πέμπτο κεφάλαιο διατυπώνονται τα τελικά συμπεράσματα που εξάχθηκαν από τα παραπάνω μοντέλα μηχανικής μάθησης, ολοκληρώνοντας την εργασία.

# Κεφάλαιο 1 : Μηχανική μάθηση

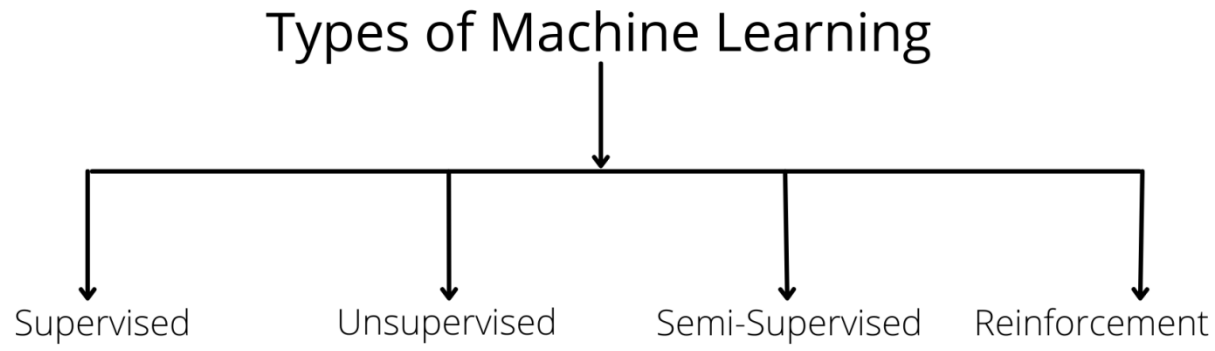
## 1.1 Εισαγωγή

Η μηχανική μάθηση είναι το υποσύνολο της τεχνητής νοημοσύνης το οποίο ασχολείται με την επεξεργασία δεδομένων και τη δημιουργία αλγορίθμων και μοντέλων, με σκοπό τη λήψη αποφάσεων και προβλέψεων. Προσομοιάζοντας την ανθρώπινη συμπεριφορά, «μαθαίνει» και αντλεί πληροφορίες από τα δεδομένα, χωρίς να έχει προγραμματιστεί εξαρχής με αυτή τη γνώση. [1]

Η μηχανική μάθηση αποτελεί πλέον αναπόσπαστο μέρος πολλών εμπορικών εφαρμογών, όπως προτάσεις ταινιών και μουσικής, προτεινόμενες διαφημίσεις σε μέσα κοινωνικής δικτύωσης και αναγνώριση αντικειμένων και προσώπων τόσο σε φωτογραφίες όσο και σε βίντεο. Επιπλέον, μπορεί να χρησιμοποιηθεί για την πρόβλεψη της τιμής αγαθών ή μετοχών στο χρηματιστήριο και το φιλτράρισμα κειμένων ως σπαμ.

Εκτός των εφαρμογών αυτών χρησιμοποιείται για προβλέψεις σε διάφορους επιστημονικούς και μη τομείς, όπως πιθανότητα βροχής στη μετεωρολογία, ανάπτυξη στρατηγικής στον αθλητισμό και εύρεση ασθένειας φυτών μέσω του φυλλώματος στη γεωργία. Επιπλέον, μπορεί να εφαρμοστεί για την διάγνωση καρδιακών αρρυθμιών στην ιατρική, το οποίο αποτελεί το κυρίως ζητούμενο στην παρούσα εργασία.

Ανάλογα με τη φύση του προβλήματος, η μηχανική μάθηση χωρίζεται σε τέσσερις υποκατηγορίες. Αυτές είναι η μηχανική μάθηση με επίβλεψη, η μηχανική μάθηση χωρίς επίβλεψη, η ημιεπιβλεπόμενη μηχανική μάθηση και η ενισχυτική μάθηση. Στις υποκατηγορίες αυτές δύνανται να εφαρμοστούν διάφορες τεχνικές, με σκοπό την βελτίωση των αποτελεσμάτων.




*Εικόνα 1: Υποκατηγορίες Μηχανικής Μάθησης[35]*

## 1.2 Μηχανική Μάθηση με επίβλεψη (Supervised)

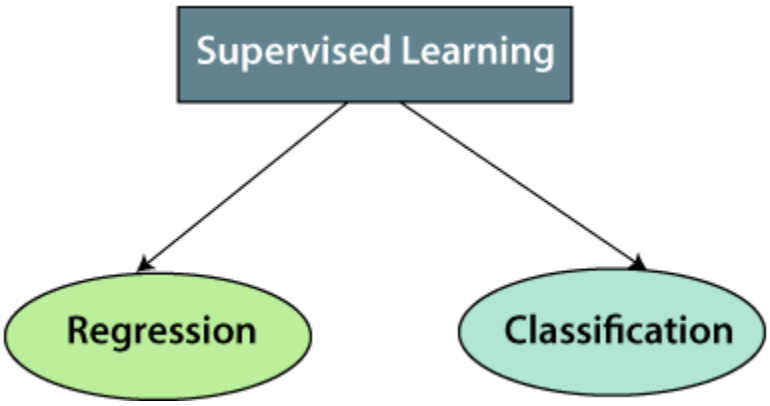
Οι αλγόριθμοι μηχανικής μάθησης με επίβλεψη μπορούν να εφαρμοστούν σε νέα δεδομένα, ενώ έχουν εκπαιδευτεί παλαιότερα σε άλλα δεδομένα. Αρχικά, το σύνολο των δεδομένων διαχωρίζεται σε δεδομένα εκπαίδευσης και δεδομένα ελέγχου. Στους αλγορίθμους μηχανικής μάθησης με επίβλεψη, οι κλάσεις του συνόλου εκπαίδευσης και του συνόλου ελέγχου είναι προκαθορισμένες ήδη από τον άνθρωπο. Ο αλγόριθμος αναλύει το γνωστό σύνολο δεδομένων και εκπαιδεύεται σε αυτά τα δεδομένα εκπαίδευσης εξάγοντας χαρακτηριστικά, προκειμένου μελλοντικά να είναι ικανός να προβλέπει τις τιμές εξόδου. Συνεπώς, στόχος του αλγορίθμου είναι να βρει μοτίβα και να κατασκευάσει μαθηματικά μοντέλα για να προβλέψει τις ετικέτες εξόδου. Επιπλέον, συγκρίνει την προβλεπόμενη τιμή εξόδου από τα δεδομένα ελέγχου με την πραγματική τιμή εξόδου του συνόλου δεδομένων και εντοπίζει τα σφάλματα και τις αποκλείσεις, τροποποιώντας το μοντέλο ανάλογα ώστε να το βελτιώσει. [2]

Η μηχανική μάθηση με επίβλεψη χωρίζεται σε δύο υποκατηγορίες, ανάλογα με το είδος της ετικέτας εξόδου των δεδομένων και κατεπέκταση τη φύση του προβλήματος. Οι υποκατηγορίες αυτές είναι η ταξινόμηση και η παλινδρόμηση.

Η ταξινόμηση (classification) είναι η υποκατηγορία της μηχανικής μάθησης με επίβλεψη που ασχολείται με την πρόβλεψη της ετικέτας ή κλάσης των δεδομένων. Οι αλγόριθμοι που χρησιμοποιούνται μαθαίνουν τις σχέσεις μεταξύ των χαρακτηριστικών εισόδου και των τιμών εξόδου, με σκοπό να προβλέπουν με ακρίβεια την ετικέτα των νέων δεδομένων βάσει των χαρακτηριστικών τους. Η ταξινόμηση χρησιμοποιείται σε προβλήματα με διακριτές εξόδους, όπως διάγνωση παθήσεων και αναγνώριση διαφορετικών ζώων.

Η παλινδρόμηση (regression) είναι η υποκατηγορία της μηχανικής μάθησης με επίβλεψη που ασχολείται με τη διερεύνηση και τη μοντελοποίηση της σχέσης μεταξύ των μεταβλητών. Οι αλγόριθμοι που χρησιμοποιούνται μαθαίνουν τις σχέσεις μεταξύ των χαρακτηριστικών εισόδου και των τιμών εξόδου, με σκοπό να προβλέπουν με ακρίβεια την τιμή νέων δεδομένων χωρίς ετικέτα. Η παλινδρόμηση χρησιμοποιείται σε προβλήματα με συνεχείς εξόδους, όπως πρόβλεψη τιμής μετοχών και βαθμολογίας φοιτητών.

Μερικές φορές το ίδιο ζητούμενο μπορεί να αντιμετωπιστεί εξίσου ως πρόβλημα ταξινόμησης και ως παλινδρόμησης, ανάλογα με τη διατύπωσή του. Για παράδειγμα, η πρόβλεψη της τιμής της θερμοκρασίας είναι πρόβλημα παλινδρόμησης, ενώ αν μας ενδιαφέρει απλά αν θα έχει κρύο ή ζέστη θεωρείται πρόβλημα ταξινόμησης.



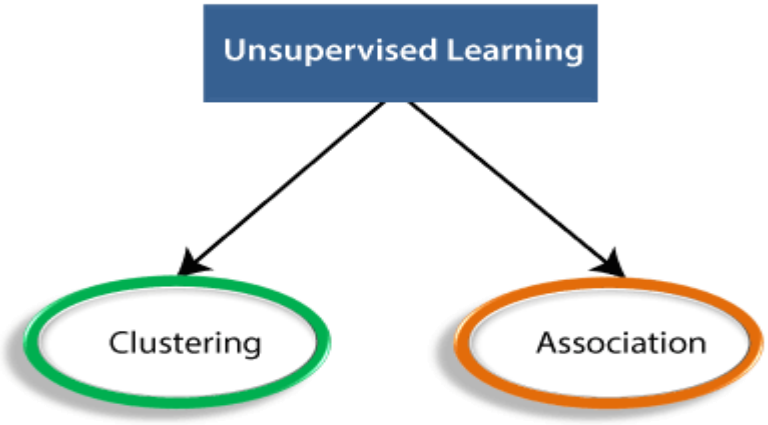

*Εικόνα 2: Μηχανική Μάθηση με Επίβλεψη [36]*

## 1.3 Μηχανική Μάθηση χωρίς επίβλεψη (Unsupervised)

Σε αντίθεση με τη μηχανική μάθηση με επίβλεψη, εδώ δεν υπάρχουν προκαθορισμένες κλάσεις/ετικέτες εξόδου. Ο κύριος στόχος της μηχανικής μάθησης χωρίς επίβλεψη είναι να ανακαλύψει μοτίβα, σχέσεις ή δομές στα δεδομένα που μπορεί να μην είναι άμεσα προφανή. Αυτό μπορεί να επιτευχθεί μέσω διαφόρων τεχνικών, όπως την ομαδοποίηση και την συσχέτιση.

Στην ομαδοποίηση/συσταδοποίηση (clustering), κύριος στόχος των αλγορίθμων μηχανικής μάθησης χωρίς επίβλεψη είναι να εντοπίσουν κοινά χαρακτηριστικά μεταξύ των δεδομένων, με σκοπό να τα κατηγοριοποιήσουν σε ομάδες με ανάλογα με ομοιότητες που παρουσιάζουν. Όταν εισάγονται νέα δεδομένα, ο αλγόριθμος χρησιμοποιεί τα χαρακτηριστικά που έχει μάθει προηγουμένως για να αναγνωρίσει την κλάση των δεδομένων. Αυτή η υποκατηγορία μηχανικής μάθησης χρησιμοποιείται κυρίως για την ομαδοποίηση των δειγμάτων και τη μείωση χαρακτηριστικών γνωρισμάτων. [2]

Στην συσχέτιση (association), βασικός στόχος των αλγορίθμων είναι η ανακάλυψη σχέσεων μεταξύ μεταβλητών σε μεγάλα σύνολα δεδομένων. Τα δεδομένα απεικονίζονται ως σύνολα συναλλαγών, όπου κάθε συναλλαγή είναι μια συλλογή στοιχείων. Ο αλγόριθμος προσδιορίζει τα στοιχεία που εμφανίζονται συχνά μαζί στις συναλλαγές και αντιπροσωπεύει αυτές τις σχέσεις ως κανόνες συσχέτισης. Χρησιμοποιείται συνήθως σε προβλήματα που οι έξοδοι είναι συμπληρωματικές, όπως να προσδιοριστούν τα αγαθά που αγοράζονται συχνά μαζί σε μια αγορά, πχ ο καφές με την ζάχαρη.[3]




*Εικόνα 3: Μηχανική Μάθηση χωρίς Επίβλεψη [37]*

## 1.4 Ημιεπιβλεπόμενη Μηχανική Μάθηση (Semi-Supervised)

Αυτή η τεχνική χρησιμοποιείται όταν έχουμε μια μικρή ποσότητα δεδομένων με ετικέτα και μια μεγάλη ποσότητα δεδομένων χωρίς ετικέτα. Λειτουργεί σαν συνδυασμός μηχανικής μάθησης με επίβλεψη και μηχανικής μάθησης χωρίς επίβλεψη. Χρησιμοποιεί τεχνικές μηχανικής μάθησης χωρίς επίβλεψη για την αρχική εκπαίδευση του μοντέλου και την πρόβλεψη των ετικετών και στη συνέχεια τροφοδοτεί αυτές τις ετικέτες σε αλγορίθμους μηχανικής μάθησης με επίβλεψη.

## 1.5 Ενισχυτική Μάθηση (Reinforcement)

Η ενισχυτική μάθηση είναι η υποκατηγορία μηχανικής μάθησης που ασχολείται με τον τρόπο με τον οποίο οι πράκτορες αλληλεπιδρούν με το περιβάλλον. Ο πράκτορας αντιλαμβάνεται το περιβάλλον του μέσω αισθητήρων και ενεργεί σε αυτό. Το περιβάλλον με τη σειρά του ανατροφοδοτεί τις ενέργειες του πράκτορα με θετικές ανταμοιβές ή τιμωρίες, αναλόγως αν η ενέργεια βελτίωσε το αποτέλεσμα ή όχι. Ο πράκτορας χρησιμοποιεί αυτή την πληροφορία για να λαμβάνει καλύτερες αποφάσεις στο μέλλον.

Σε αντίθεση με τη μηχανική μάθηση με επίβλεψη, εδώ η επιθυμητή έξοδος δεν είναι γνωστή. Επιπλέον, δεν υπάρχει κάποιος άνθρωπος που επηρεάζει τη διαδικασία της μάθησης, καθώς η μόνη πληροφορία που δέχεται ο πράκτορας είναι μια αριθμητική τιμή, είτε ως τιμωρία είτε ως ανταμοιβή. [4]

Στην θετική ενίσχυση (positive reinforcement) μια ενέργεια του πράκτορα ενισχύεται με την προσθήκη ανταμοιβής. Η θετική ενίσχυση λειτουργεί αυξάνοντας την πιθανότητα να επαναληφθεί μια συμπεριφορά στο μέλλον ως απάντηση σε συγκεκριμένα ερεθίσματα, αφού ακολουθηθεί από ένα θετικό αποτέλεσμα. Η θετική ενίσχυση χρησιμοποιείται για να ενθαρρύνει έναν πράκτορα να επαναλάβει ενέργειες που οδηγούν σε θετικά αποτελέσματα, όπως η λήψη ανταμοιβής. Ο πράκτορας μαθαίνει να συσχετίζει τη δράση με το θετικό αποτέλεσμα και με την πάροδο του χρόνου, γίνεται πιο πιθανό να επαναλάβει αυτή τη δράση σε παρόμοιες συνθήκες.

Αντίθετα, στην αρνητική ενίσχυση (negative reinforcement) μια συμπεριφορά του πράκτορα ενισχύεται με την αφαίρεση ενός αποτρεπτικού ερεθίσματος. Η αρνητική ενίσχυση λειτουργεί αυξάνοντας την πιθανότητα να επαναληφθεί μια συμπεριφορά στο μέλλον ως απάντηση σε συγκεκριμένα ερεθίσματα, αφού ακολουθηθεί από την αφαίρεση ενός αποτρεπτικού αποτελέσματος. Η αρνητική ενίσχυση χρησιμοποιείται για να ενθαρρύνει έναν πράκτορα να επαναλάβει ενέργειες που οδηγούν στην απομάκρυνση ενός αρνητικού αποτελέσματος, όπως η αποφυγή τιμωρίας. Ο πράκτορας μαθαίνει να συσχετίζει τη δράση με την απομάκρυνση του αρνητικού αποτελέσματος και με την πάροδο του χρόνου, γίνεται πιο πιθανό να επαναληφθεί αυτή η ενέργεια σε παρόμοιες συνθήκες. [5]

Η εξαφάνιση αναφέρεται στη μείωση ή την εξάλειψη μιας συμπεριφοράς λόγω της απουσίας ενίσχυσης. Η εξαφάνιση συμβαίνει όταν η ενίσχυση δεν είναι πλέον διαθέσιμη και η συμπεριφορά αρχίζει να εξασθενεί και τελικά εξαφανίζεται. Στην ενισχυτική μάθηση, η εξαφάνιση χρησιμοποιείται για τη μείωση ή την εξάλειψη ανεπιθύμητων συμπεριφορών.

Η τιμωρία στην ενισχυτική μάθηση είναι η προσθήκη ενός αρνητικού αποτελέσματος για τη μείωση της συχνότητας μιας συμπεριφοράς. Η τιμωρία λειτουργεί μειώνοντας την πιθανότητα να επαναληφθεί μια συμπεριφορά στο μέλλον ως απάντηση σε συγκεκριμένα ερεθίσματα, αφού ακολουθηθεί από αρνητικό αποτέλεσμα.

## 1.6 Τεχνικές Ensemble

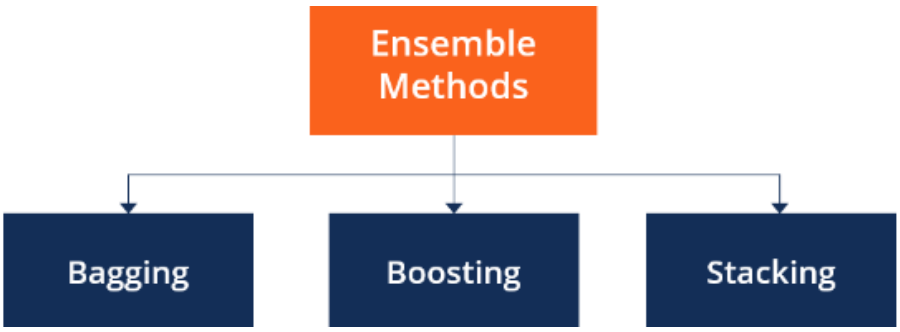
Οι τεχνικές ensemble είναι διαδικασίες δημιουργίας πολλαπλών μοντέλων, τα οποία συνδυάζονται για την επίλυση ενός συγκεκριμένου προβλήματος μηχανικής μάθησης. Χρησιμοποιούνται κυρίως για τη βελτίωση της απόδοσης ενός μοντέλου με μικρότερη διακύμανση ή για τη μείωση της πιθανότητας μιας λανθασμένης επιλογής. Οι πιο συχνές τεχνικές είναι η ενθυλάκωση, η ενίσχυση , η επικλινής ενίσχυση και η συσσώρευση. Η επικλινής ενίσχυση, ή αλλιώς ενίσχυση κλίσης, υπάγεται στην ενίσχυση.

Η ενθυλάκωση (bagging - bootstrap aggregating) χρησιμοποιείται για τη βελτίωση της σταθερότητας ενός μοντέλου. Συνδιάζει πολλαπλές περιπτώσεις του ίδιου μοντέλου, το οποίο είναι κάθε φορά εκπαιδευμένο σε διαφορετικό τυχαίο δείγμα των δεδομένων. Τα υποσύνολα δημιουργούνται με τυχαία δειγματοληψία των δεδομένων. Κάποια δεδομένα εκπαίδευσης μπορεί να επαναληφθούν για ορισμένα μοντέλα ή και να μη συμπεριληφθούν. Η τελική πρόβλεψη γίνεται βάσει του μέσου όρου των προβλέψεων όλων των μοντέλων.

Η ενίσχυση (boosting) συνδυάζει αδύναμα μοντέλα για να παράγει ένα πιο ισχυρό μοντέλο. Βασίζεται στην διαδοχική εκπαίδευση αδύναμων μοντέλων, όπου κάθε μοντέλο προσπαθεί να διορθώσει τα λάθη που έκανε το προηγούμενο. Τα βάρη που αποδίδονται κατά τις εκπαιδεύσεις προσαρμόζονται δυναμικά, ώστε οι περιπτώσεις που ταξινομούνται εσφαλμένα από προηγούμενα μοντέλα να έχουν μεγαλύτερο βάρος στην εκπαίδευση των επόμενων. Με τον τρόπο αυτό, το επόμενο κατά σειρά μοντέλο επικεντρώνεται σε αυτές τις περιπτώσεις, καθιστώντας έτσι το επόμενο μοντέλο πιο ακριβές. Το τελικό αποτέλεσμα είναι ένα σύνολο αδύναμων μοντέλων που συνεργάζονται για να παράγουν ένα ενιαίο, πιο ισχυρό μοντέλο. [6]

Η ενίσχυση κλίσης (gradient boosting) έχει τις ίδιες αρχές με την ενίσχυση, αλλά χρησιμοποιεί αποκλειστικά δέντρα αποφάσεων. Τα δέντρα αποφάσεων εκπαιδεύονται με διαδοχικό τρόπο για να ελαχιστοποιήσουν την συνάρτηση απώλειας, χρησιμοποιώντας την κλίση της συνάρτησης απώλειας σε σχέση με τις προβλέψεις του μοντέλου. [7]

Η συσσώρευση (stacking) αποτελείται από δύο φάσεις. Αρχικά, τα βασικά μοντέλα εκπαιδεύονται στη βάση δεδομένων και ύστερα οι προβλέψεις των βασικών μοντέλων χρησιμοποιούνται ως χαρακτηριστικά για την εκπαίδευση του μετα-μοντέλου. Τέλος, οι προβλέψεις των βασικών μοντέλων χρησιμοποιούνται ως είσοδοι στο μετα-, προκειμένου να παρέχει το τελικό αποτέλεσμα. Η συσσώρευση μπορεί να συνδιάσει μοντέλα διαφορετικών τύπων, όπως δέντρα αποφάσεων με K-Γείτονες.




*Εικόνα 4: Τεχνικές Ensemble [38]*

# Κεφάλαιο 2 : Καρδιολογία και καρδιαγγειακές αρρυθμίες

## 2.1 Εισαγωγή

Οι καρδιαγγειακές παθήσεις (CVDs) είναι η κυρίαρχη αιτία θανάτου παγκοσμίως, αντιπροσωπεύοντας το 31% όλων των θανάτων παγκοσμίως. Το 80% των θανάτων λόγω CVD οφείλονται σε καρδιακές προσβολές και εγκεφαλικά επεισόδια και το ένα τρίτο αυτών συμβαίνουν πρόωρα σε άτομα κάτω των 70 ετών. Η καρδιακή ανεπάρκεια είναι ένα κοινό συμβάν που προκαλείται από CVDs. Ο όρος καρδιακή νόσος αναφέρεται σε διάφορους τύπους καρδιακών παθήσεων. Οι καρδιακές παθήσεις περιγράφουν μια σειρά από καταστάσεις που επηρεάζουν την καρδιά. Οι καρδιακές παθήσεις περιλαμβάνουν: [8]

• Ασθένειες αιμοφόρων αγγείων, όπως στεφανιαία νόσο

• Καρδιακά ελαττώματα που υπάρχουν εκ γενετής (κληρονομικά καρδιακά ελαττώματα)

• Ασθένεια καρδιακής βαλβίδας

• Ασθένεια του καρδιακού μυός

• Καρδιακή λοίμωξη

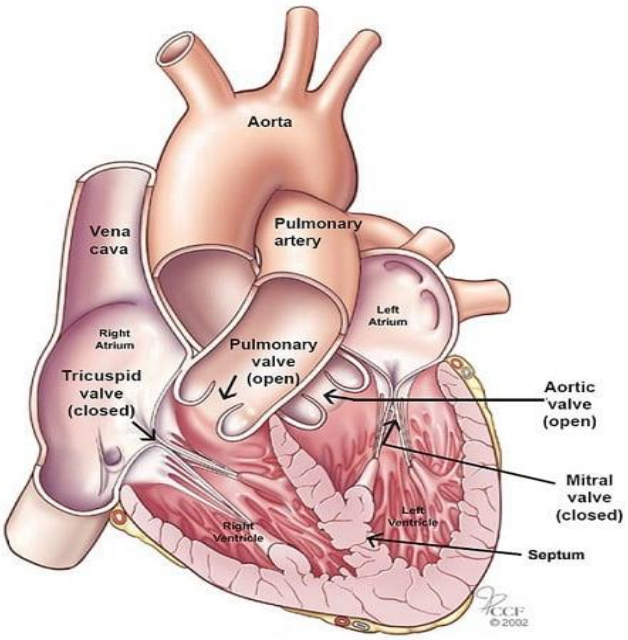
• Προβλήματα καρδιακού ρυθμού (αρρυθμίες)

Τα άτομα με καρδιαγγειακή νόσο ή που διατρέχουν υψηλό καρδιαγγειακό κίνδυνο (λόγω της παρουσίας ενός ή περισσότερων παραγόντων κινδύνου) χρειάζονται έγκαιρη ανίχνευση και διαχείριση της πάθησης. Ως εκ τούτου, ο κύριος στόχος είναι να προβλεφθεί εάν κάποιος διατρέχει υψηλό κίνδυνο διάγνωσης ως καρδιακός ασθενής. Μέσω της μηχανικής μάθησης δίδεται η ικανότητα πρόβλεψης πιθανής καρδιακής πάθησης. Ένα μοντέλο μηχανικής μάθησης μπορεί να βοηθήσει, καθώς είναι πιο γρήγορο και δεν απαιτεί παρουσία γιατρού. Σε περίπτωση που κάποιος πιθανός ασθενής βρίσκεται σε κατηγορία υψηλού κινδύνου, μπορεί να παρεπεμφθεί γρήγορα στον καρδιολόγο. Με τον τρόπο αυτό, αποφεύγονται τα πολλά ραντεβού από άτομα που δεν τα χρειάζονται, αφήνοντας διαθέσιμα για ασθενείς που τα έχουν ανάγκη. [9] [10]

## 2.2 Ανατομία καρδιάς

Η καρδιά είναι μια μυϊκή αντλία που βρίσκεται μεταξύ των πνευμόνων στη μέση του στήθους, πίσω και ελαφρώς προς τα αριστερά του στέρνου. Αντλεί αίμα σε όλο το σώμα με ρυθμικές συσπάσεις και αποτελεί το κύριο όργανο του κυκλοφορικού συστήματος. Το βάρος της κυμαίνεται από 200 - 425 γραμμάρια και έχει περίπου το μέγεθος μίας γροθιάς. [11]

Η καρδιά περιβάλλεται από το περικάρδιο, μια μεμβράνη διπλής στρώσης. Το εξωτερικό στρώμα του περικαρδίου περιβάλλει τις ρίζες των μεγάλων αιμοφόρων αγγείων της καρδιάς και συνδέεται με συνδέσμους στη σπονδυλική στήλη, το διάφραγμα και άλλα μέρη του σώματος. Το εσωτερικό στρώμα του περικαρδίου συνδέεται με τον καρδιακό μυ. Μια υγρή επικάλυψη διαχωρίζει τα δύο στρώματα της μεμβράνης, επιτρέποντας στην καρδιά να κινείται ελεύθερα χωρία να τρίβεται καθώς συσπάται. [12]




*Εικόνα 5: Ανατομία Καρδιάς[12]*

Η καρδιά χωρίζεται σε τέσσερις θαλάμους, δύο στην κορυφή (κόλπους) και δύο στο κάτω μέρος (κοιλίες), ένα σε κάθε πλευρά της καρδιάς. Οι κόλποι και οι κοιλίες είναι διαχωρισμένοι από ένα τοίχωμα αλλά συνδέονται με βαλβίδες, ώστε να ωθείται σωστά το αίμα εντός της καρδιάς. Η ανώτερη κοίλη φλέβα μεταφέρει αίμα από το άνω μέρος του σώματος και η κατώτερη κοίλη φλέβα φέρνει αίμα από το κάτω μέρος του σώματος. Στη συνέχεια, ο δεξιός κόλπος αντλεί το αποξυγονωμένο αίμα στη δεξιά κοιλία. Η δεξιά κοιλία αντλεί το αίμα στους πνεύμονες μέσω της πνευμονικής αρτηρίας ώστε να οξυγονωθεί. Αφού οι πνεύμονες γεμίσουν το αίμα με οξυγόνο, οι πνευμονικές φλέβες μεταφέρουν το αίμα στον αριστερό κόλπο, ο οποίος αντλεί το αίμα στην αριστερή κοιλία. Τέλος, η αριστερή κοιλία αντλεί το οξυγονωμένο αίμα στο υπόλοιπο σώμα.

Τα ηλεκτρικά ερεθίσματα από τον καρδιακό μυ προκαλούν τη ρυθμική σύσπαση της καρδιάς, δηλαδή τον καρδιακό παλμό. Αυτό το ηλεκτρικό ερέθισμα ξεκινά στον κόμβο sinoatrial, που βρίσκεται στην κορυφή του δεξιού κόλπου. Ο κόμβος αυτός ονομάζεται μερικές φορές ο "φυσικός βηματοδότης της καρδιάς". Κατά τη συστολή, αυξάνεται η πίεση στην καρδιά και αντλεί το αίμα, ενώ κατά τη διαστολή οι κόλποι ή κοιλίες αντίστοιχα χαλαρώνουν και δέχονται το αίμα.

## 2.3 Ηλεκτροκαρδιογράφημα

Ένα ηλεκτροκαρδιογράφημα (ΗΚΓ) είναι ένα γράφημα τάσης συναρτήσει του χρόνου της ηλεκτρικής δραστηριότητας της καρδιάς χρησιμοποιώντας ηλεκτρόδια τοποθετημένα στο δέρμα. Ένα ΗΚΓ είναι απολύτως ασφαλές, διαρκεί λίγα λεπτά και δεν προκαλεί πόνο στον ασθενή, καθώς είναι μια απλή δοκιμή που πραγματοποιείται τοποθετώντας κολλώδη ηλεκτρόδια στο στήθος, τα πόδια και τους καρπούς του ασθενούς.

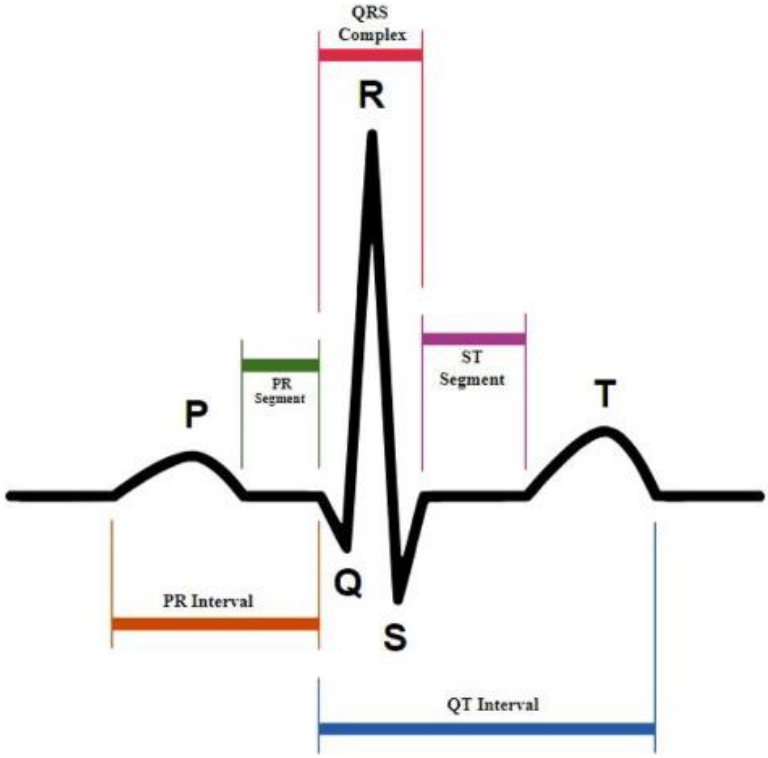
Το ΗΚΓ αποκαλύπτει πολλά πράγματα για την καρδιά, συμπεριλαμβανομένου του ρυθμού της, εάν οι διαδρομές ηλεκτρικής αγωγιμότητας είναι άθικτες, εάν ορισμένοι θάλαμοι είναι διευρυμένοι και ακόμη και η κατά προσέγγιση ισχαιμική θέση σε περίπτωση καρδιακής προσβολής (έμφραγμα του μυοκαρδίου). [13] Το ΗΚΓ χρησιμοποιείται κλινικά για τη διάγνωση διαφόρων ανωμαλιών και καταστάσεων που σχετίζονται με την καρδιά. Η ανάλυση του ΗΚΓ γίνεται για τον προσδιορισμό της κατάστασης των καρδιακών αρρυθμιών, της κολπικής και κοιλιακής υπερτροφίας, του ισχαιμικού και του εμφράγματος του μυοκαρδίου, των επιδράσεων των φαρμάκων και της αξιολόγησης των λειτουργιών του βηματοδότη.

Η μη επεμβατική διάγνωση αρρυθμίας βασίζεται στο πρότυπο ηλεκτροκαρδιογράφημα 12-lead, το οποίο μετρά ηλεκτρικά δυναμικά από 10 ηλεκτρόδια τοποθετημένα σε διαφορετικά μέρη της επιφάνειας του σώματος, έξι στο στήθος και τέσσερα στα άκρα του ασθενούς. Προκειμένου να παρέχεται αποτελεσματική θεραπεία για αρρυθμίες, είναι σημαντική η έγκαιρη διάγνωση. Η έγκαιρη ανίχνευση ορισμένων τύπων παροδικών, βραχυπρόθεσμων ή σπάνιων αρρυθμιών απαιτεί μακροχρόνια παρακολούθηση της ηλεκτρικής δραστηριότητας της καρδιάς.

Τυπικά, το εύρος συχνότητας ενός σήματος ΗΚΓ είναι 0,05 - 100 Hz και το δυναμικό του εύρος μεταξύ 1-10 mV. [14] Τα σήματα του ΗΚΓ αποτελούνται από διάφορα κύματα που αντιπροσωπεύουν διάφορες δραστηριότητες των κοιλιών και τον κόλπων της καρδιάς. Τα κυριότερα σημεία του συμβολίζονται με τους λατινικούς χαρακτήρες P, Q, R, S, T. Συγκρεκριμένα:

- P κύμα, μια εικόνα που προκύπτει από την διαστολή του κόλπου. Αυτό το κύμα είναι ≤0,3 mV όταν η καρδιά βρίσκεται σε κανονική κατάσταση και έχει πλάτος ≤0.12 s. Το κύμα P λαμβάνεται ανιχνεύοντας την υψηλότερη τιμή σήματος στο διάστημα των 200 ms πριν από το σημείο Q.
- Το σημείο Q λαμβάνεται με την ανίχνευση της χαμηλότερης αξίας σημάτων στο διάστημα 80 ms πριν από το σημείο P

- Το σημείο R είναι στην επόμενη άνοδο του γραφήματος, υπό την προϋπόθεση ότι διασχίζει την ισοηλεκτρική γραμμή και γίνεται "θετικό".
- Το σημείο S λαμβάνεται με την ανίχνευση της χαμηλότερης αξίας σημάτων στο διάστημα 80 ms μετά από το σημείο P.
- Κύμα QRS, μια εικόνα που προκύπτει από την συστολή της κοιλίας. Όταν η καρδιά βρίσκεται σε κανονική κατάσταση, αυτό το κύμα έχει πλάτος 0,06–0,12 s και το ύψος του εξαρτάται από το ηλεκτρόδιο που μετράται.
- Κύμα T, ένα κύμα που προκύπτει από την διαστολή της κοιλίας. Σε μια φυσιολογική καρδιακή κατάσταση, το κύμα T έχει θετική αξία σε όλους τους αγωγούς. Το κύμα T λαμβάνεται με την ανίχνευση της υψηλότερης αξίας σημάτων στο διάστημα 400 ms μετά από το σημείο S.
- Διάστημα PR, μετρούμενο από την αρχή του κύματος P έως την αρχή του κύματος κύμα QRS. Σε κανονική καρδιακή κατάσταση, αυτό το κύμα έχει πλάτος 0,12–0,20 δευτερόλεπτα.
- Τμήμα ST, μετρούμενο από το τέλος του κύματος QRS έως την αρχή του κύματος T. Αντιπροσωπεύει την περίοδο ηλεκτρικής σταθερότητας μεταξύ κοιλιακής συστολής (σύμπλεγμα QRS) και διαστολής (κύμα T).




*Εικόνα 6: Χαρακτηριστικά Στοιχεία ΗΚΓ [14]*

Το QRS είναι η πιο χαρακτηριστική κυματομορφή στο ΗΚΓ. Δεδομένου ότι αντικατοπτρίζει την ηλεκτρική δραστηριότητα μέσα στην καρδιά κατά τη διάρκεια της κοιλιακής συστολής, ο χρόνος εμφάνισής της και το σχήμα της παρέχουν πολύτιμες πληροφορίες για την τρέχουσα κατάσταση της καρδιάς. Συνεπώς, η ανίχνευση του QRS παρέχει τις βασικές αρχές για σχεδόν όλους τους αυτοματοποιημένους αλγόριθμους ανάλυσης ΗΚΓ. [15]

Το διάστημα RR είναι ο χρόνος μεταξύ δύο κυμάτων QRS, από το οποίο λαμβάνεται ο στιγμιαίος καρδιακός ρυθμός. Ο στιγμιαίο καρδιακός ρυθμός και ο καρδιακός ρυθμός δεν πρέπει να συγχέονται, καθώς ο καρδιακός ρυθμός αποτελεί τον μέσο όρο των παλμών της καρδιάς ανά λεπτό.

Ένας φυσιολογικός καρδιακός παλμός κυμαίνεται από 60 έως 100 παλμούς/λεπτό. Ένας βραδύτερος ρυθμός από αυτό ονομάζεται βραδυκαρδία και ένας υψηλότερος ρυθμός ονομάζεται ταχυκαρδία. Εάν οι κύκλοι δεν απέχουν ομοιόμορφα, μπορεί να ενδείκνυται αρρυθμία. Εάν το διάστημα PR είναι μεγαλύτερο από 0,2 δευτερόλεπτα, μπορεί να υποδηλώνει απόφραξη κόμβου. [16]

## 2.4 Καρδιαγγειακές Αρρυθμίες

Οι καρδιαγγειακές αρρυθμίες χωρίζονται σε έντεκα ξεχωριστές κατηγορίες. Αν και οι κατηγορίες αυτές δύνανται μερικές φορές να φέρουν ομοιότητες μεταξύ τους, θεωρείται ότι είναι ανεξάρτητες.

**Φλεβοκομβική Βραδυκαρδία:** Φυσιολογικά ο φλεβόκομβος βηματοδοτεί την καρδιά με συχνότητα από 60-100 παλμούς ανά λεπτό. Εάν η συχνότητα πέσει κάτω από 60 παλμούς ανά λεπτό, πρόκειται για φλεβοκομβική βραδυκαρδία.[17]

**Φλεβοκομβικός Ρυθμός:** Ένας φλεβοκομβικός ρυθμός είναι οποιοσδήποτε καρδιακός ρυθμός στον οποίο η αποπόλωση του καρδιακού μυός ξεκινά στον κόλπο. Χαρακτηρίζεται από την παρουσία σωστών προσανατολισμένων κυμάτων P στο ηλεκτροκαρδιογράφημα.

**Κολπική Μαρμαρυγή:** Η κολπική μαρμαρυγή είναι ένας μη φυσιολογικός καρδιακός ρυθμός που αυξάνει δραστικά τον κίνδυνο κάποιου πιθανού εγκεφαλικού επεισοδίου, καρδιακής ανεπάρκειας και άλλων καρδιακών επιπλοκών. Κατά την κολπική μαρμαρυγή, οι κόλποι χωρίς συγκεκριμένο ρυθμό, χωρίς να συντονίζονται με τις κοιλίες. Η κολπική μαρμαρυγή αποτελεί την συχνότερη μορφή αρρυθμίας (33% όλων των αρρυθμιών). [18]

**Φλεβοκομβική Ταχυκαρδία:** Ως ταχυκαρδία ορίζεται η κατάσταση όπου σε φάση ηρεμίας η καρδιά συσπάται γρήγορα, με περισσότερες από 100 συσπάσεις (παλμούς) ανά λεπτό στους ενήλικες, ενώ παιδιά ηλικίας 1-2 ετών και 12-15 ετών το όριο είναι > 150/ λεπτό και > από 120 / λεπτό αντίστοιχα. Αν οφείλεται σε φυσιολογική ταχεία παραγωγή ηλεκτρικού ρεύματος στον φλεβόκομβο, λέγεται φυσιολογική φλεβοκομβική ταχυκαρδία. [19]

**Κολπικός Πτερυγισμός**: Ο κολπικός πτερυγισμός αποτελεί την δεύτερη πιο συχνή ταχυρρυθμία μετά την κολπική μαρμαρυγή. Είναι ένας τύπος μη φυσιολογικού καρδιακού ρυθμού ή αρρυθμίας που εμφανίζεται όταν ένα βραχυκύκλωμα στην καρδιά προκαλεί την συνεχόμενη

διαστολή και συστολή των άνω κόλπων πολύ γρήγορα. Ο κολπικός πτερυγισμός είναι σημαντικός όχι μόνο λόγω των συμπτωμάτων του, αλλά επειδή μπορεί να προκαλέσει εγκεφαλικό επεισόδιο που μπορεί να οδηγήσει σε μόνιμη αναπηρία ή θάνατο.[ 20]

**Αναπνευστική Αρρυθμία**: Γνωστό και ως Sinus Arrhythmia/Αναπνευστική Αρρυθμία . Πρόκειται για μία καλοήθης αρρυθμία κατά την οποία η καρδιακή συχνότητα μεταβάλλεται σε μικρό βαθμό ανάλογα με τις αναπνευστικές κινήσεις. Ο χρόνος μεταξύ καρδιακών παλμών μπορεί να είναι ελαφρώς μικρότερος ή μεγαλύτερος ανάλογα με την εισπνοή και την εκπνοή (αυξάνεται και επιβραδύνεται αντίστοιχα). [21]

**Υπερκοιλιακή Ταχυκαρδία**: Η υπερκοιλιακή ταχυκαρδία (supraventricular tachycardia-SVT) είναι αρρυθμία που χαρακτηρίζεται από αυξημένους καρδιακούς παλμούς, οι οποίοι κυμαίνονται μεταξύ 150-250 σφίξεις ανά λεπτό και εμφανίζονται απότομα. Επιπλέον, μπορεί να προέρχεται από οποιοδήποτε σημείο της καρδιάς εκτός των κοιλιών. Εμφανίζεται συχνά σε νέους οι οποίοι είναι αρκετά δραστήριοι στην καθημερινότητά τους, καθιστώντας την διάγνωσή της δύσκολη μέσω ενός ηλεκτρογραφήματος (ΗΚΓ).

**Κολπική Ταχυκαρδία**: Η κολπική ταχυκαρδία είναι ένας γρήγορος καρδιακός παλμός (αρρυθμία). Είναι ένας τύπος υπερκοιλιακής ταχυκαρδίας (SVT). Κατά τη διάρκεια ενός επεισοδίου κολπικής ταχυκαρδίας, ο καρδιακός ρυθμός αυξάνεται σε περισσότερους από 100 παλμούς το λεπτό πριν επιστρέψει σε έναν τυπικό καρδιακό ρυθμό περίπου 60 έως 80 παλμούς το λεπτό.

**SAAWR**: Ο ρυθμός του καρδιακού παλμού που εξελίσσεται σε περιπλανώμενο κολπικό βηματοδότη (WAP). Ως WAP ορίζεται η λειτουργία της καρδιάς με κανονικό ρυθμό, παρά τη μετατόπιση του ελέγχου του καρδιακού παλμού από τον βηματοδότη στους κόλπους. [23]

**Κομβική Ταχυκαρδία Επανεισόδου**: Η κομβική ταχυκαρδία επανεισόδου οφείλεται στην ύπαρξη ενός ηλεκτρικού κυκλώματος επανεισόδου, το οποίο βρίσκεται στον κολποκοιλιακό κόμβο και προκαλεί αρρυθμία στους παλμούς.

**Κολποκοιλιακή Ταχυκαρδία:** Ως κολποκοιλιακή ταχυκαρδία, πλήρης ονομασία κολποκοιλιακή ταχυκαρδία επανεισόδου, χαρακτηρίζεται το ηλεκτρικό κύκλωμα επανεισόδου όπου τα ηλεκτρικά σήματα που αποστέλλονται μέσω του κολποκοιλιακού κόμβου συγχέονται με ένα παραπληρωματικό δεμάτιο. Η συνέλιξη αυτή έχει ως αποτέλεσμα την αύξηση των καρδιακών παλμών σε τιμές ύψους έως και 220 παλμούς ανά λεπτό.

Η κομβική ταχυκαρδία επανεισόδου και η κολποκοιλιακή ταχυκαρδία επανεισόδου αποτελούν τις δύο πιο συχνές μορφές υπερκοιλιακής ταχυκαρδίας (ΥΚΤ) και η εμφάνισή τους είναι τόσο ξαφνική όσο και ο τερματισμός τους. Η ΥΚΤ είναι υποκατηγορία της SVT και τα αίτια ύπαρξής της είναι οι παραπάνω του φυσιολογικού αριθμού ηλεκτρικές συνδέσεις στην καρδιά. Οι συνδέσεις αυτές δημιουργούν ένα ηλεκτρικό «βραχυκύκλωμα» το οποίο προκαλεί την ταχυκαρδία. Δεν εμφανίζεται λόγω κληρονομικών παραγόντων ούτε προκαλείται από τον τρόπο ζωής του ασθενή.

# Κεφάλαιο 3 : Μέσα υλοποίησης

Προκειμένου να βρεθεί η υψηλότερη δυνατή απόδοση, χρησιμοποιήθηκαν οι εξής αλγόριθμοι μηχανικής μάθησης με επίβλεψη: LightGBM, Δέντρα Αποφάσεων, Random Forest, K-Γείτονες και Extreme Gradient Boosting. Για την επεξεργασία της βάσης δεδομένων και την υλοποίηση των μοντέλων, χρησιμοποιήθηκε η γλώσσα προγραμματισμού Python σε Anaconda Jupyter Notebooks.

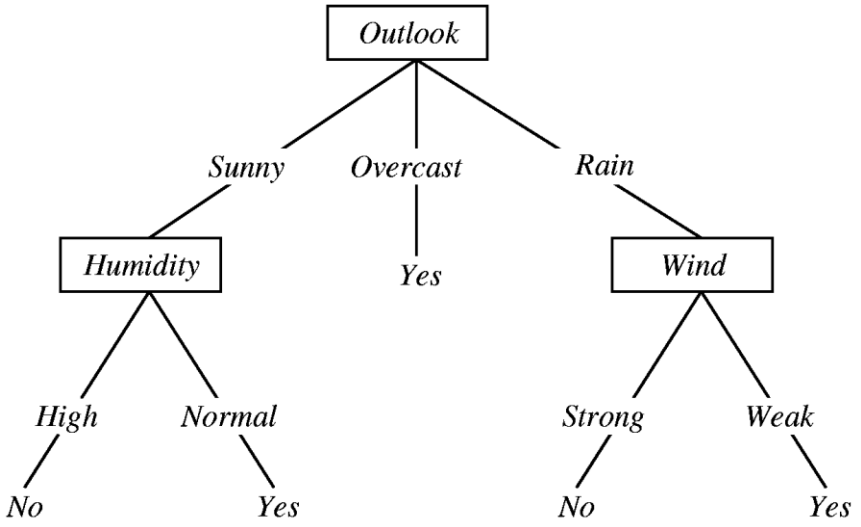
## 3.1 Αλγόριθμοι

### 3.1.1 Δέντρα αποφάσεων

Τα δέντρα αποφάσεων είναι ένα είδος αλγόριθμου μάθησης με επίβλεψη που χρησιμοποιείται κυρίως για προβλήματα ταξινόμησης. Βασίζεται σε ένα δενδροειδές μοντέλο που ελέγχει πιθανές περιπτώσεις. [25]

Το δέντρο αποτελείται από κόμβους, κλαδιά και φύλλα. Κάθε εσωτερικός κόμβος αντιπροσωπεύει μια δοκιμή σε κάποιο χαρακτηριστικό, κάθε κλάδος αντιπροσωπεύει το αποτέλεσμα μιας δοκιμής και κάθε φύλλο αντιπροσωπεύει μια ετικέτα κλάσης. Ο αρχικός κόμβος στο δέντρο αποφάσεων είναι γνωστός ως κόμβος ρίζας. Η ιδέα πίσω από τον ταξινομητή δέντρου αποφάσεων είναι να δημιουργηθεί ένα μοντέλο που προβλέπει την τιμή μιας μεταβλητής μαθαίνοντας ιεραρχικά απλούς κανόνες απόφασης που εξάγονται από τα χαρακτηριστικά δεδομένων.

Ο αλγόριθμος του δέντρου αποφάσεων ξεκινά επιλέγοντας το καλύτερο χαρακτηριστικό για να χωρίσει τα δεδομένα σε μικρότερα υποσύνολα. Στη συνέχεια, διαχωρίζει αναδρομικά τα υποσύνολα με βάση τις τιμές του επιλεγμένου χαρακτηριστικού, δημιουργώντας έναν νέο εσωτερικό κόμβο και νέους κλάδους για κάθε τιμή. Η διαδικασία επαναλαμβάνεται για κάθε εσωτερικό κόμβο μέχρι να ικανοποιηθεί ένα προκαθορισμένο κριτήριο, όπως ένα μέγιστο βάθος δέντρου ή ένας ελάχιστος αριθμός δειγμάτων ανά φύλλο. [26]




*Εικόνα 7: Ταξινόμηση Πιθανότητας Βροχής Βάσει Καιρικών Χαρακτηριστικών [26]*

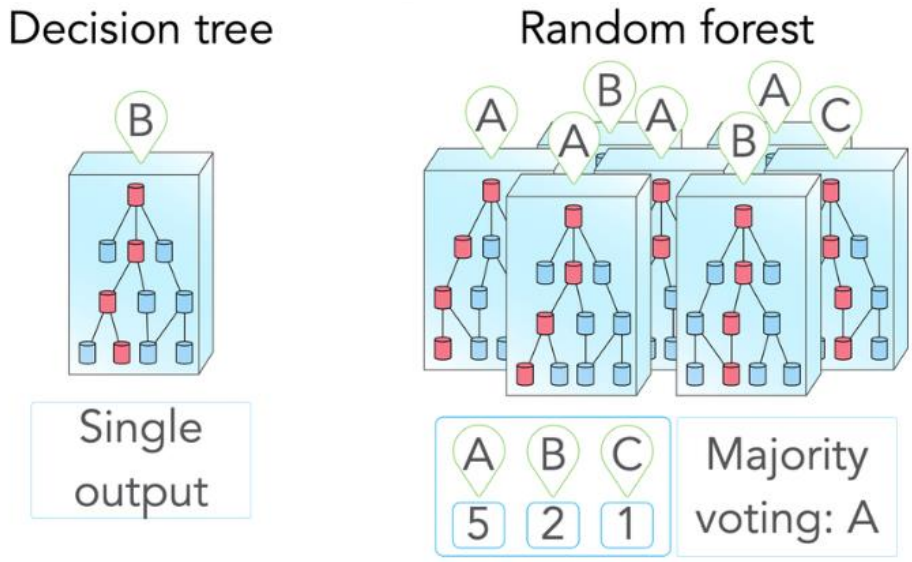
### 3.1.2 Random Forest

Ο αλγόριθμος αυτός βασίζεται στα δέντρα αποφάσεων και κατασκευάζει ένα πλήθος ανεξάρτητων δέντρων αποφάσεων επιλέγοντας τυχαία δείγματα από τα χαρακτηριστικά εισόδου κατά την εκπαίδευση, με σκοπό τη βελτίωση της συνολικής απόδοσης και σταθερότητας του μοντέλου. Στην πραγματικότητα χρησιμοποιεί την τεχνική της ενθυλάκωσης.

Η βασική ιδέα πίσω από τον αλγόριθμο είναι να συνδυάσει τις προβλέψεις πολλαπλών δέντρων αποφάσεων, όπου κάθε δέντρο είναι χτισμένο σε ένα διαφορετικό τυχαίο υποσύνολο των δεδομένων. Αυτή η διαδικασία επαναλαμβάνεται πολλές φορές, οδηγώντας σε ένα σύνολο δέντρων αποφάσεων. Η τελική πρόβλεψη του Random Forest καθορίζεται από τον μέσο όρο των προβλέψεων των μεμονωμένων δέντρων ή επιλέγοντας εκείνο που έχει την μεγαλύτερη ψηφοφορία των προβλέψεων.

Η διαδικασία εκπαίδευσης για έναν Random Forest είναι παρόμοια με αυτή ενός δέντρου αποφάσεων, αλλά με δύο σημαντικές διαφορές. Αρχικά, ο αλγόριθμος ξεκινά επιλέγοντας ένα τυχαίο υποσύνολο των δεδομένων, που ονομάζεται δείγματα bootstrap, από τα δεδομένα εκπαίδευσης. Στη συνέχεια, δημιουργεί ένα δέντρο αποφάσεων για κάθε δείγμα που χρησιμοποιεί ως εκκίνηση και επιλέγει ένα τυχαίο υποσύνολο χαρακτηριστικών για κάθε διαχωρισμό. Αυτές οι διαδικασίες επαναλαμβάνονται πολλές φορές, οδηγώντας σε ένα σύνολο δέντρων αποφάσεων.

Λαμβάνοντας τον μέσο όρο ή ψηφίζοντας τις προβλέψεις πολλών δέντρων αποφάσεων, ο Random Forest είναι σε θέση να μειώσει τη διακύμανση και την προκατάληψη (overfitting) των μεμονωμένων δέντρων, με αποτέλεσμα ένα πιο ισχυρό και ακριβές μοντέλο. Επιπλέον, έχει τη δυνατότητα να μετρήσει τη βαρύτητα των χαρακτηριστικών, κάτι που μπορεί να είναι χρήσιμο για την επιλογή των βέλτιστων χαρακτηριστικών. Παράλληλα όμως είναι υπολογιστικά δαπανηρός και ευαίσθητος στον θόρυβο.



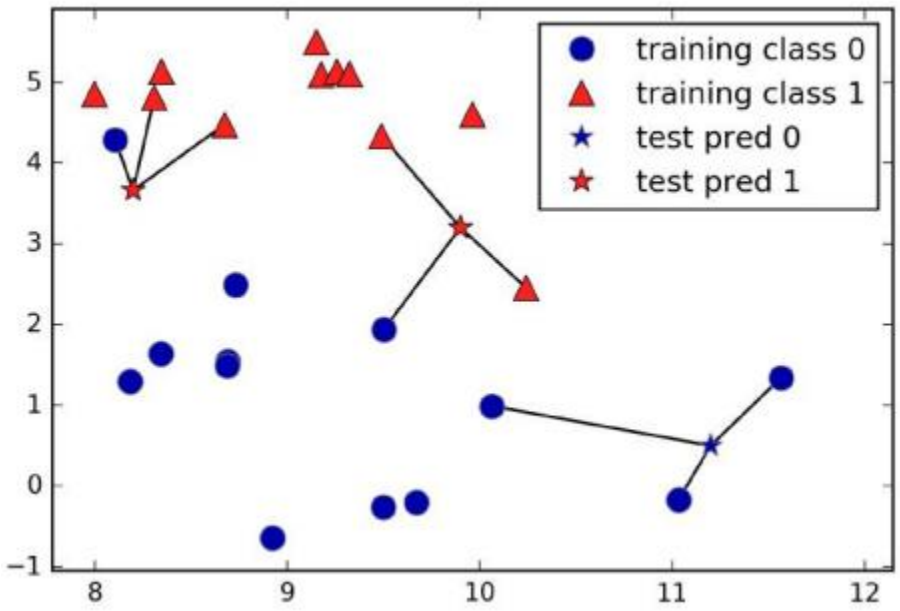

*Εικόνα 8: Σύγκριση Δέντρου Απόφασης και Random Forest [39]*

### 3.1.3 K-Γείτονες

Συναντάται συχνά σε εφαρμογές μηχανικής μάθησης, καθώς είναι ένας από τους πιο απλούς αλγορίθμους. Είναι ένας μη παραμετρικός αλγόριθμος μηχανικής μάθησης, ο οποίος αναζητά τους k-κοντινότερους γείτονες ενός συγκεκριμένου κάθε φορά δείγματος δεδομένων εκπαίδευσης και στη συνέχεια χρησιμοποιεί τις ετικέτες κλάσης των γειτόνων για να προβλέψει την ετικέτα κλάσης του δεδομένου σημείου. Συνεπώς, κατατάσσει τα σημεία σύμφωνα με ομοιότητάς τους. Λαμβάνει υπόψη τους k-κοντινότερους γείτονες, μία ακέραια παράμετρος που μπορεί να καθοριστεί από το χρήση. Η μετρική απόστασης που χρησιμοποιεί συνώθως είναι κυρίως η Ευκλείδεια απόσταση, με εναλλακτικές μετρικές την απόσταση Manhattan και την απόσταση Minkowski.

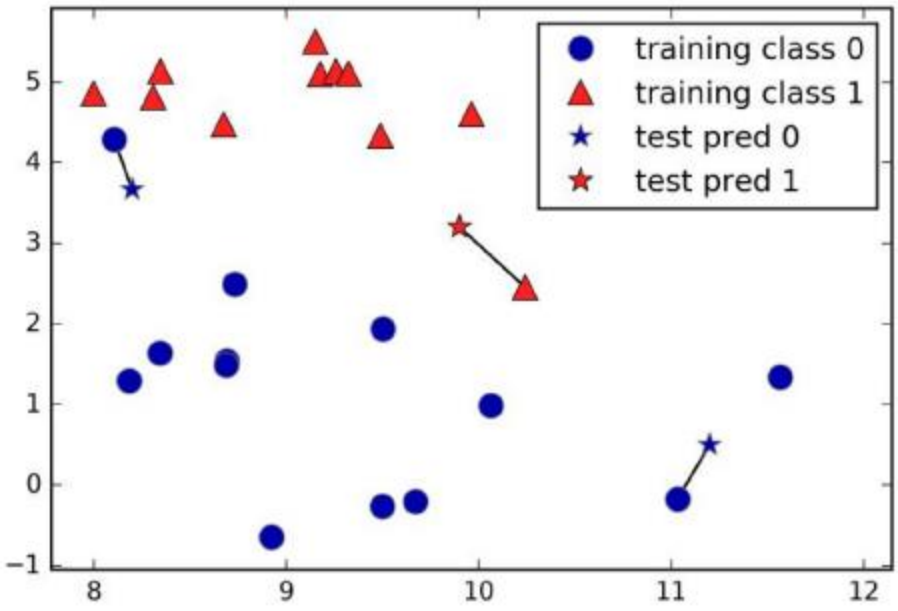
Στα προβλήματα ταξινόμησης, ένα αντικείμενο ταξινομείται ανάλογα με την ετικέτα που έχει η πλειοψηφία των k-γειτόνων του και κατατάσσεται στην τάξη που κυριαρχεί μεταξύ αυτών. Σε προβλήματα παλινδρόμησης, η έξοδος είναι ο μέσος όρος των τιμών των k-πλησιέστερων γειτόνων.

Για παράδειγμα στην παρακάτω εικόνα, για k = 3 και χρησιμοποιώντας ως μετρική απόστασης την Ευκλείδεια απόσταση, βρίσκουμε τα τρία κοντινότερα σημεία-δεδομένα εκπαίδευσης (κόκκινα τρίγωνα και μπλε κύκλοι) για κάθε σημείο που ανήκει στα δεδομένα ελέγχου (αστέρι). Ύστερα, του αποδίδουμε την κλάση (χρώμα) που έχει η πλειοψηφία των 3 γειτόνων του.




*Εικόνα 9: Ταξινόμηση με 3-κοντινότερους γείτονες[25]*

Αντίστοιχα, στην παρακάτω εικόνα φαίνεται η αλλαγή της κλάσης για k = 1 με ίδια μετρική απόστασης.




*Εικόνα 10: Ταξινόμηση με 3-κοντινότερους γείτονες[25]*

Ο KNN (K-Nearest Neighbors) είναι ένας απλός και εύκολος στην εφαρμογή αλγόριθμος. Είναι αποδοτικός και μπορεί να διαχειριστεί μεγάλα σύνολα δεδομένων. Συστήνεται επίσης για το χειρισμό μη γραμμικών ορίων αποφάσεων και για προβλήματα πολλαπλών κλάσεων. Ωστόσο, μπορεί να είναι ευαίσθητος στην επιλογή του k-γείτονα και της μετρικής απόστασης και δεν λειτουργεί καλά με δεδομένα υψηλών διαστάσεων ή μη δομημένα. Τέλος, μπορεί να είναι υπολογιστικά δαπανηρός ώστε να βρεθούν οι πλησιέστεροι k-γείτονες για μεγάλα σύνολα δεδομένων και ο αλγόριθμος δεν παράγει ένα μοντέλο, άρα μπορεί να είναι δύσκολο να ερμηνευτεί.

### 3.1.4 Extreme Gradient Boosting Tree

Ο αλγόριθμος XGBoost είναι ένας αλγόριθμος ενίσχυσης κλίσης, ο οποίος χρησιμοποιείται συχνά σε προβλήματα παλινδρόμησης και ταξινόμησης. Είναι ικανός να διαχειριστεί ταχύτατα μεγάλες ποσότητες γραμμικών και μη γραμμικών δεδομένων, καθώς και αρχεία δεδομένων με κενά στοιχεία στα δείγματα και τα χαρακτηριστικά τους. Υποστηρίζεται από πολλές γλώσσες προγραμματισμού, συμπεριλαμβανομένων των Python, R και Julia, ενσωματώνεται εύκολα σε πολλές εφαρμογές μηχανικής μάθησης. [27]

Σε πολλές περιπτώσεις, ο XGBoost είναι καλύτερος από τους υπόλοιπους αλγόριθμους ενίσχυσης κλίσης. Η Python καθιστά δυνατή την πρόσβαση σε έναν τεράστιο αριθμό εσωτερικών παραμέτρων που μπορούν να τροποποιηθούν, ώστε να επιτύχουμε τη βέλτιστη δυνατή ακρίβεια και απόδοση.

Ο αλγόριθμος λειτουργεί με δέντρα αποφάσεων χτισμένα παράλληλα, αντί διαδοχικά. Ο XGBoost προσθέτει σταδιακά όλο και περισσότερες συνθήκες "if" στο δέντρο αποφάσεων για την κατασκευή ενός ισχυρότερου μοντέλου. Περνώντας από κάθε επίπεδο, ελέγχει τις κλίσεις

και χρησιμοποιεί τα μερικά αθροίσματά τους για να αξιολογήσει την ποιότητα των διαχωρισμών σε κάθε πιθανή διάσπαση στα δεδομένα εκπαίδευσης. Άρα, αντί να εκπαιδεύουμε το καλύτερο δυνατό μοντέλο στα δεδομένα (όπως στις παραδοσιακές μεθόδους μηχανικής μάθησης), εκπαιδεύουμε χιλιάδες μοντέλα σε διάφορα υποσύνολα του συνόλου των δεδομένων εκπαίδευσης και στη συνέχεια επιλέγουμε αυτόματα το μοντέλο με τις καλύτερες επιδόσεις.

Μερικά σημαντικά χαρακτηριστικά του XGBoost είναι:

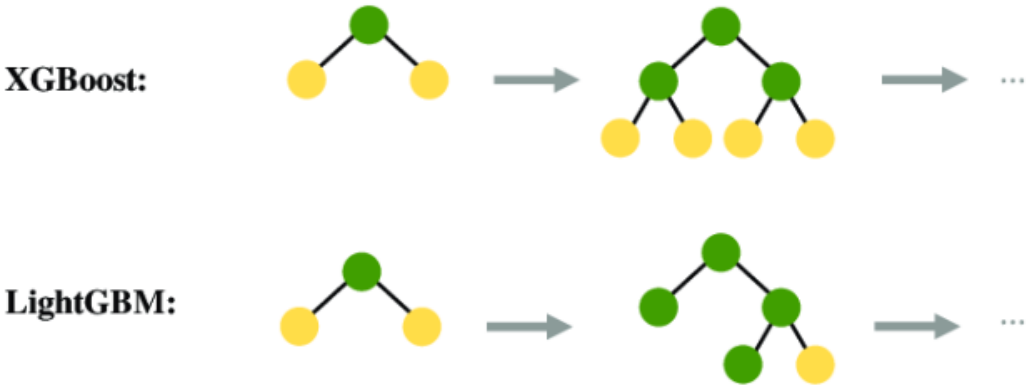
1. Παραλληλισμός: το μοντέλο εφαρμόζεται για εκπαίδευση με πολλαπλούς πυρήνες CPU.
2. Κανονικοποίηση: περιλαμβάνει διαφορετικές κυρώσεις κανονικοποίησης για την αποφυγή overfitting. Η σωστή κατανομή ποινών οδηγεί σε πιο επιτυχημένη εκπαίδευση, ώστε το μοντέλο να μπορεί να γενικευτεί επαρκώς.
3. Μη γραμμικότητα: το XGBoost μπορεί να ανιχνεύσει και να μάθει από μη γραμμικά μοτίβα δεδομένων.
4. Cross-validation: ενσωματωμένη ήδη στον αλγόριθμο.
5. Επεκτασιμότητα: μπορεί να τρέξει κατανεμημένο χάρη σε κατανεμημένους διακομιστές και συμπλέγματα όπως το Hadoop και το Spark, παρέχοντας έτσι τη δυνατότητα επεξεργασίας τεράστιων ποσοτήτων δεδομένων.
6. Ελλιπείς τιμές: σε αντίθεση με τους προαναφερόμενους αλγορίθμους, μπορεί να διαχειριστεί ελλιπείς τιμές.

### 3.1.5 LightGBM

Όπως ο XGBoost, ο LightGBM (light gradient-boosting machine) είναι ένας αλγόριθμος ενίσχυσης κλίσης που βασίζεται σε δέντρα. Είναι ταχύτερος από τον XGBoost, διαθέτει επίσης ενσωματωμένη υποστήριξη για παράλληλη εκμάθηση και εκμάθηση GPU και μπορεί να διαχειριστεί μεγάλες ποσότητες γραμμικών και μη δεδομένων και κενές τιμές. Ο LGBM διαθέτει επίσης διεπαφές για πολλές γλώσσες προγραμματισμού, συμπεριλαμβανομένων των Python, R και C++, καθιστώντας το εύκολο στη χρήση και την ενσωμάτωσή του σε ένα ευρύ φάσμα εφαρμογών μηχανικής μάθησης. [28]

Μια σημαντική διαφορά μεταξύ των δύο αλγορίθμων έγκειται στον τρόπο κατασκευής των δέντρων. Ο LightGBM δεν αναπτύσσει τα δέντρα αποφάσεων γραμμικά ή παράλληλα, όπως κάνουν οι περισσότερες άλλες υλοποιήσεις. Αντ ' αυτού, επεκτείνει τα δέντρα βάσει των κόμβων-φύλλων. Επιλέγει το φύλλο που πιστεύει ότι θα μειώσει περισσότερο την απώλεια.

Εκτός αυτού, ο LightGBM δεν χρησιμοποιεί τον ευρέως χρησιμοποιούμενο αλγόριθμο εκμάθησης δέντρων αποφάσεων με ταξινομημένη βάση, ο οποίος αναζητά το καλύτερο σημείο διαχωρισμού σε ταξινομημένες τιμές χαρακτηριστικών, όπως κάνουν το XGBoost ή άλλες υλοποιήσεις. Αντιθέτως, εφαρμόζει έναν εξαιρετικά βελτιστοποιημένο αλγόριθμο εκμάθησης δέντρων αποφάσεων που βασίζεται σε ιστόγραμμα, ο οποίος αποδίδει μεγάλα πλεονεκτήματα τόσο στην απόδοση όσο και στην κατανάλωση μνήμης.




*Εικόνα 11: Σύγκριση Ανάπτυξης XGBoost και LightGBM [40]*

## 3.2 Περιβάλλον υλοποίησης

### 3.2.1 Jupyter Notebook

Η εφαρμογή Jupyter Notebook (πρώην IPython Notebooks) είναι μια διαδικτυακή πλατφόρμα που λειτουργεί ως υπολογιστικό περιβάλλον για τη δημιουργία εγγράφων Jupyter notebooks. Τα έγγραφα Notebook είναι αρχεία JSON που παράγονται από την εφαρμογή Jupyter Notebook, τα οποία περιέχουν τόσο κώδικα (π.χ. python) όσο και στοιχεία εμπλουτισμένου κειμένου (παράγραφος, εξισώσεις, αριθμοί, ειδικοί χαρακτήρες κ. λπ.). Τα έγγραφά του κατανοητά και αναγνώσιμα από τον άνθρωπο καθώς περιέχουν την περιγραφή της ανάλυσης και τα αποτελέσματα (αριθμοί, πίνακες κ.λπ..) και συγχρόνως έγγραφα κώδικα που μπορούν να εκτελεστούν από τον υπολογιστή. [29]

Τα Jupyter notebooks έχουν την χαρακτηριστική κατάληξη ".ipynb". Τα αρχεία μπορούν να εξαχθούν ως HTML (για παράδειγμα, για αναρτήσεις ιστολογίου), reStructuredText, LaTeX, PDF και διαφάνειες, μέσω της εντολής nbconvert. Επιπλέον, οποιαδήποτε έγγραφο .ipynb που διατίθεται από μια δημόσια διεύθυνση URL μπορεί να κοινοποιηθεί μέσω του Jupyter Notebook Viewer με την εντολή <nbviewer>. Αυτή η υπηρεσία φορτώνει το έγγραφο από τη διεύθυνση URL και το μετατρέπει σε αρχείο HTML.

Η εφαρμογή Jupyter Notebook μπορεί να εκτελεστεί σε τοπική επιφάνεια εργασίας που δεν απαιτεί πρόσβαση στο διαδίκτυο ή μπορεί να εγκατασταθεί σε απομακρυσμένο διακομιστή και να έχει πρόσβαση μέσω του διαδικτύου. Εκτός από την εμφάνιση, επεξεργασία και εκτέλεση εγγράφων notebook, η εφαρμογή διαθέτει ένα Dashboard, δηλαδή έναν πίνακα ελέγχου που εμφανίζει τοπικά αρχεία και επιτρέπει το άνοιγμα εγγράφων notebook ή τον τερματισμό των πυρήνων τους (kernel).

## 3.3 Βιβλιοθήκες

### 3.3.1 Numpy

Το πακέτο Numpy αποτελεί τη βάση της επιστημονικής πληροφορικής με γλώσσα προγραμματισμού Python. Παρέχει N-διάστατες συστοιχίες, διαχειρίζεται ποικίλους τύπους και μορφές δεδομένων και δύναται να ενσωματώνει μεγάλες βάσεις δεδομένων. Επιπλέον, προσφέρει λειτουργίες μαθηματικών, όπως επεξεργασία σχημάτων, πιθανότητες, διακριτό μετασχηματισμό Fourier, δυαδική λογική, γραμμική άλγεβρα, πολυώνυμα και προσομοιώσεις. [30]

### 3.3.2 Pandas

Το πακέτο Pandas είναι μια βιβλιοθήκη ανοιχτού κώδικα που παρέχει υψηλής απόδοσης και συγχρόνως εύχρηστες δομές και εργαλεία ανάλυσης δεδομένων για τη γλώσσα προγραμματισμού Python.[31] Συγκεκριμένα, περιλαμβάνει:

1. Ένα γρήγορο και αποτελεσματικό DataFrame για επεξεργασία δεδομένων με ενσωματωμένο indexing.
2. Εργαλεία ανάγνωσης και εγγραφής δεδομένων μεταξύ δομών, ακόμα και αν είναι με διαφορετική μορφή μεταξύ τους. Χαρακτηριστικά, διαχειρίζεται αρχεία CSV και κειμένου, Microsoft Excel, βάσεις δεδομένων SQL και HDF5.
3. Ευέλικτη αναδιαμόρφωση και διαχείριση δεδομένων.
4. Εύρεση και διαχείριση κενών δεδομένων.
5. Διαχείριση και επεξεργασία μεγάλων βάσεων δεδομένων.
6. Οι στήλες (χαρακτηριστικά) μπορούν να τροποποιηθούν και να διαγραφούν από δομές δεδομένων, μεταβάλλοντας αντίστοιχα το μέγεθος της βάσης δεδομένων χωρίς να επηρεάσει το αρχικό αρχείο.

### 3.3.3 Matplotlib

Το πακέτο Matplotlib είναι μια ολοκληρωμένη βιβλιοθήκη για τη δημιουργία τόσο στατικών όσο και κινούμενων ή διαδραστικών απεικονίσεων στην Python. Βασίζεται σε αριθμητικές συστοιχίες και έχει σχεδιαστεί για να λειτουργεί με το Pyplot, μία διεπαφή σαν το MATLAB. Ένα από τα πιο σημαντικά χαρακτηριστικά του Matplotlib είναι η συμβατότητά του με πολλά λειτουργικά συστήματα και backend γραφικά. Το Matplotlib υποστηρίζει δεκάδες backends, με αποτέλεσμα να λειτουργεί ανεξάρτητα από το λειτουργικό σύστημα. [32]

### 3.3.4 Sklearn

Το Scikit-learn, επίσης γνωστό ως sklearn, είναι μια βιβλιοθήκη Python για μηχανική μάθηση. Βασίζεται στις βιβλιοθήκες NumPy και SciPy και έχει σχεδιαστεί για να ενσωματώνεται με το υπόλοιπο επιστημονικό οικοσύστημα Python, όπως το Matplotlib και την Pandas. Παρέχει ένα ευρύ φάσμα εργαλείων για μάθηση με και χωρίς επίβλεψη, συμπεριλαμβανομένων γραμμικών και μη γραμμικών μοντέλων, ομαδοποίησης, μείωσης διαστάσεων, κωδικοποίησης ετικετών και προεπεξεργασίας. Επιπλέον, προσφέρει ποικιλία εργαλείων για την επιλογή μοντέλου, αξιολόγηση μετρικών απόδοσης και επιλογή υπερπαραμέτρων.

### 3.3.5 Xgboost

Το XGBoost είναι μια βελτιστοποιημένη κατανεμημένη βιβλιοθήκη ενίσχυσης κλίσης που έχει σχεδιαστεί για να είναι εξαιρετικά αποδοτική και ευέλικτη. Εφαρμόζει αλγόριθμους μηχανικής μάθησης κάτω από το πλαίσιο ενίσχυσης κλίσης. Το XGBoost παρέχει παράλληλη ενίσχυση δέντρων (επίσης γνωστή ως GBDT, GBM) που επιλύει πολλά προβλήματα μηχανικής μάθησης με γρήγορο και ακριβή τρόπο.

### 3.3.6 Lightgbm

Το LightGBM είναι ένα πλαίσιο ενίσχυσης κλίσης που χρησιμοποιεί αλγόριθμους μηχανικής μάθησης με βάση τα δέντρα αποφάσεων. Τα κυριότερα πλεονεκτήματά του είναι:

1. Γρηγορότερη ταχύτητα εκπαίδευσης και υψηλότερη αποδοτικότητα.
2. Χαμηλότερη χρήση μνήμης.
3. Καλύτερη ακρίβεια.
4. Υποστήριξη παράλληλης μάθησης, κατανεμημένης μάθησης και μάθησης με χρήση GPU.
5. Δυνατότητα διαχείρισης δεδομένων μεγάλης κλίμακας.

### 3.3.7 Imblearn

Το unbalanced-learn (που εισάγεται ως imblearn) είναι μια βιβλιοθήκη ανοιχτού κώδικα με άδεια MIT που βασίζεται στο scikit-learn (εισάγεται ως sklearn) και παρέχει εργαλεία που διαχειρίζονται την ταξινόμηση με ανισορροπημένες τάξεις. Συγκεκριμένα, χρησιμοποιείται το RandomOverSampler, μία τεχνική δημιουργίας ενός πιο ισορροπημένου συνόλου δεδομένων. Συμπληρώνει τα δεδομένα εκπαίδευσης τυχαία δεδομένα από τις κατηγορίες εξόδου που έχουν το μικρότερο αριθμό δειγμάτων.

# Κεφάλαιο 4: Υλοποίηση

Στο κεφάλαιο αυτό αναλύεται η διαδικασία και ο τρόπος υλοποίησης της εφαρμογής. Συγκεκριμένα γίνεται λόγος για τον τρόπο επεξεργασίας των δεδομένων, καθώς και για την μεθοδολογία που εφαρμόστηκε για τη δημιουργία και τη βελτιστοποίηση των μοντέλων. Τέλος, ακολουθεί σύγκριση των αποδόσεων των μοντέλων.

## 4.1 Βάση δεδομένων

Για την υλοποίηση της εφαρμογής χρησιμοποιήθηκε η βάση δεδομένων του Πανεπιστημίου Chapman και του Λαϊκού Νοσοκομείου Shaoxing (Shaoxing Hospital Zhejiang University School of Medicine). Η συγκεκριμένη βάση δεδομένων αποτελείται από σήματα που προήλθαν από ηλεκτροκαρδιογραφήματα 12-απαγωγών 10.646 διαφορετικών ασθενών, με ρυθμό δειγματοληψίας 500 Hz και διαθέτει 11 κοινούς ρυθμούς (Rhythm) και πρόσθετες καρδιαγγειακές παθήσεις και συνδυασμούς τους (Beat). Το σύνολο δεδομένων αποτελείται από ΗΚΓ διάρκειας 10 δευτερολέπτων, 12 διαστάσεων και με ετικέτες για ρυθμούς και άλλες συνθήκες για κάθε χαρακτηριστικό. [33]

### 4.1.1 Δεδομένα και χαρακτηριστικά

Τα δεδομένα αφορούν την διάγνωση του είδους της καρδιακής αρρυθμίας, χρησιμοποιώντας 14 κλινικά χαρακτηριστικά (δεν συμπεριλαμβάνουμε τη στήλη FileName του αρχείου δεδομένων).

**Χαρακτηριστικά εισόδου:**

- Beat: Ετικέτα ύπαρξης καρδιακών παθήσεων ή/και συνδυασμών τους. Διακρίνονται 56 μοναδικές παθήσεις, οι οποίες δημιουργούν 742 διαφορετικούς συνδυασμούς που παρατηρήθηκαν στους ασθενείς.
- PatientAge: Ηλικία ασθενούς με εύρος τιμών 4-98 χρονών.
- Gender: Φύλο ασθενούς.
- VentricularRate: Ρυθμός κοιλίων σε BPM, με εύρος τιμών 34 - 263 BPM.
- AtrialRate: Κολπικός ρυθμός σε BPM, με εύρος τιμών 0 - 535 BPM.
- QRSDuration: Διάρκεια QRS σε msec, με εύρος τιμών 18 - 256 msec.
- QTInterval: Διάστημα QT σε msec, με εύρος τιμών 114 - 736 msec.
- QTCorrected: Διάστημα QT από την αρχή του QRS ως και το τέλος του κύματος T σε msec, με εύρος τιμών 219 - 760 msec.
- RAxis: Άξονας R, με εύρος τιμών -89 - 270.
- TAxis: Άξονας T, με εύρος τιμών -89 - 270.
- QRSCount: Μέτρηση QRS, με εύρος τιμών 5 - 40.

- QOnset: Q onset (αρχή του κύμματος Q) στα δείγματα, με εύρος τιμών 159 - 414.
- QOffset: Q offset (τέλος του κύμματος Q) στα δείγματα, με εύρος τιμών 249 - 432.
- TOffset: T offset (τέλος του κύμματος T) στα δείγματα, με εύρος τιμών 281 - 582.

**Χαρακτηριστικά εξόδου:**

- Rhythm: Ετικέτα Ρυθμού

Οι πιθανές κατηγορίες εξόδου είναι:

1. SB / Sinus Bradycardia
2. SR / Sinus Rhythm
3. AFIB / Atrial Fibrillation
4. ST / Sinus Tachycardia
5. AF / Atrial Flutter
6. SI / Sinus Irregularity
7. SVT / Supraventricular Tachycardia
8. AT / Atrial Tachycardia
9. AVNRT / Atrioventricular Node Reentrant Tachycardia
10. AVRT / Atrioventricular Reentrant Tachycardia
11. SAAWR / Sinus Atrium to Atrial Wandering Rhythm

## 4.1.2 Επεξεργασία δεδομένων

### 4.1.2.1 Διαχωρισμός Gender

Αντίστοιχα, διαχωρίζουμε το χαρακτηριστικό Gender στις υποκατηγορίες FEMALE με 4690 δείγματα και MALE με 5956 δείγματα. Αυτές οι νέες στήλες αντικαθιστούν την στήλη Gender.

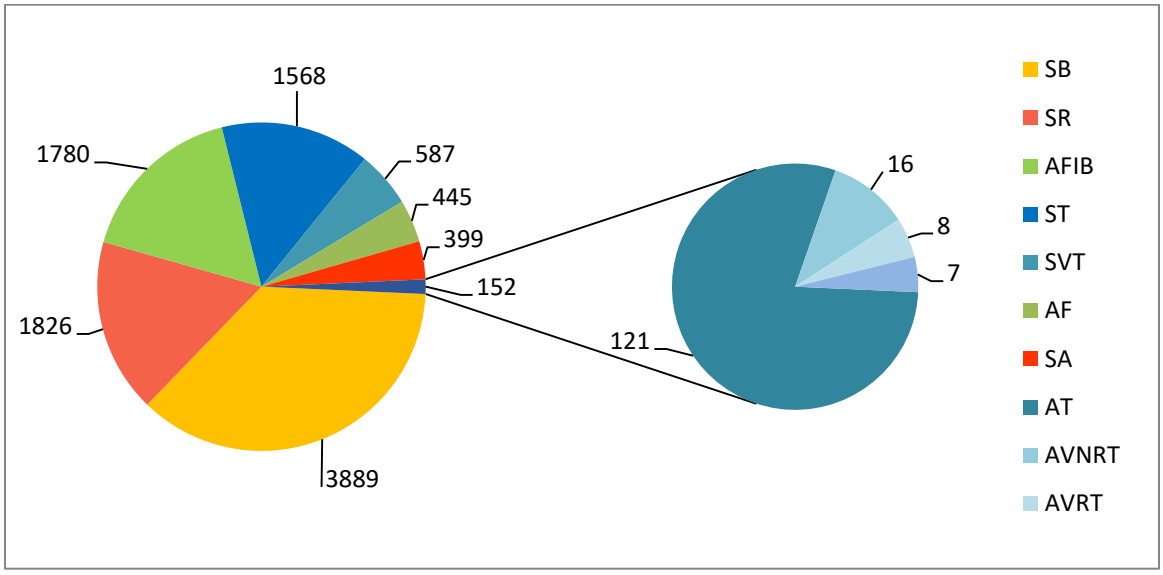
### 4.1.2.2 Διαχωρισμός Beat

Για την καλύτερη εκπαίδευση και επίδοση των μοντέλων, διαχωρίζουμε τις διαφορετικές κατηγορίες στις 742 μοναδικές περιπτώσεις, με σκοπό να έχουμε πιο ακριβή περιγραφή των δειγμάτων. Όπως προαναφέρθηκε, οι 742 περιπτώσεις προκύπτουν από συνδυασμούς 56 βασικών παθήσεων. Σε περίπτωση που ο ασθενής δεν εμφανίζει κάποια πάθηση ή συνδυασμό, αναγράφεται ένδειξη NONE. Αυτές οι νέες στήλες αντικαθιστούν την στήλη Beat.

### 4.1.2.3 Ομαδοποίηση Rhythm

Υπάρχουν 7 κατηγορίες με πλήθος δειγμάτων κάτω από το 10% του συνόλου . Αυτές είναι:

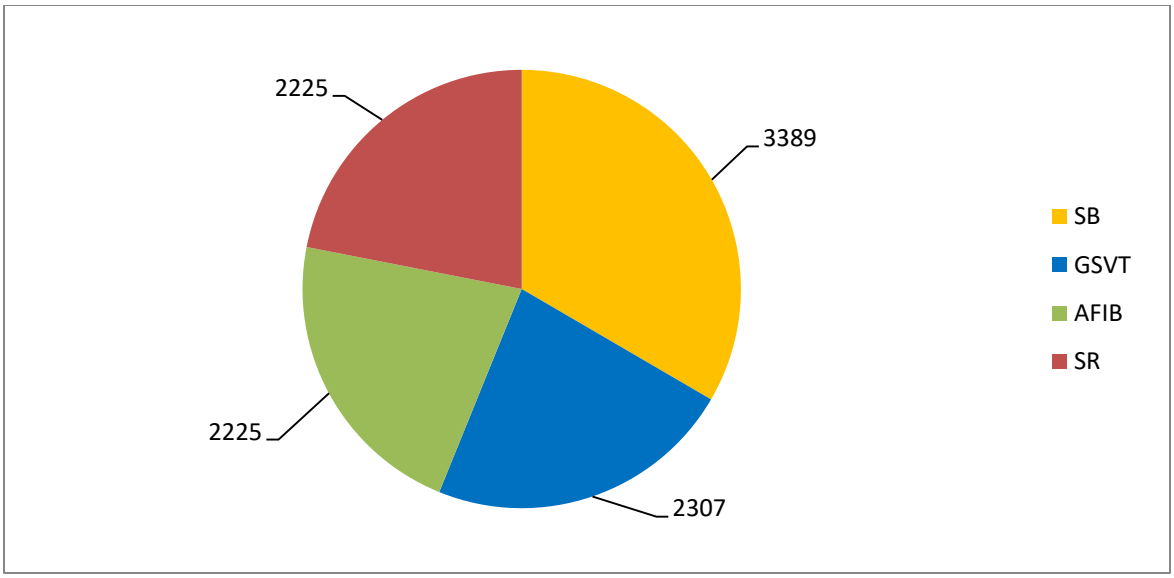
- SVT με 587 δείγματα ή αλλιώς 5.43%,
- AF με 445 δείγματα ή αλλιώς 4.18%,
- SA με 399 δείγματα ή αλλιώς 3.18%,
- AT με 121 δείγματα ή αλλιώς 1.14%,
- AVNRT με 16 δείγματα ή αλλιώς 0.15%,
- AVRT με 8 δείγματα ή αλλιώς 0.08% και
- SAAWR με 7 δείγματα ή αλλιώς 0.07%




*Διάγραμμα 1: Κατανομή Αρχικών Κατηγοριών*

Για την καλύτερη εκπαίδευση και επίδοση των μοντέλων, ομαδοποιούμε τις διαφορετικές κατηγορίες σε 4 ευρύτερες, με σκοπό να έχουν όσο το δυνατόν ίση ποσότητα δειγμάτων. Οι υποκατηγορίες που απαρτίζουν τις ευρύτερες πρέπει να είναι όμοιες ή να σχετίζονται μεταξύ τους. Συγκεκριμένα:

1. Η κατηγορία SB έχει αρκετά δείγματα ώστε να μην ενωθεί με κάποια άλλη.
2. Η AFIB και η AF συχνά συναντούνται μαζί. Συνεπώς, κατηγοριοποιούμε κάθε ΗΚΓ με αυτούς τους ρυθμούς ως AFIB.
3. Η SVT είναι γενικός όρος που χρησιμοποιείται στα ηλεκτροκαρδιογραφήματα σε περιπτώσεις που δεν μπορεί να διαγνωστεί η ακριβής πάθηση. Συνεπώς εδώ, εντάσσουμε όλες τις ταχυκαρδίες στην ευρύτερη κατηγορία GSVT (general SVT).
4. Ενώνοντας την SR και την SA ως SR τις διαχωρίζουμε από την ευρύτερη κατηγορία GSVT. Επιπλέον, η SA μπορεί να διαχωριστεί αργότερα από την SR μέσω της διακύμανσης του χρόνου που πέρασε μεταξύ δύο διαδοχικών R-κυμάτων του σήματος QRS στο ηλεκτροκαρδιογράφημα.




*Διάγραμμα 2: Ομαδοποίηση Κατηγοριών*

| Ακρώνυμο πάθησης | Πλήρης Ονομασία Πάθησης (στα Αγγλικά) | Ακρώνυμο πάθησης | Πλήρης Ονομασία Πάθησης (στα Αγγλικά) |
|---|---|---|---|
| 1AVB | 1 degree atrioventricular block | LVQRSLL | lower voltage QRS in limb lead |
| 2AVB | 2 degree atrioventricular block | MI | myocardial infarction |
| 2AVB1 | 2 degree atrioventricular block(Type one) | MIBW | myocardial infraction in back wall |
| 2AVB2 | 2 degree atrioventricular block(Type two) | MIFW | Myocardial infgraction in the front wall |
| 3AVB | 3 degree atrioventricular block | MILW | Myocardial infraction in the lower wall |
| ABI | atrial bigeminy | MISW | Myocardial infraction in the side wall |
| ALS | Axis left shift | PRIE | PR interval extension |
| APB | atrial premature beats | PWC | P wave Change |
| AQW | abnormal Q wave | QTIE | QT interval extension |
| ARS | Axis right shift | RAH | right atrial hypertrophy |
| AVB | atrioventricular block | RAHV | right atrial high voltage |
| CCR | countercolockwise rotation | RBBB | right bundle branch block |
| CR | colockwise rotation | RVH | right ventricle hypertrophy |
| ERV | Early repolarization of the ventricles | STDD | ST drop down |
| FQRS | fQRS Wave | STE | ST extension |
| IDC | Interior differences conduction | STTC | ST-T Change |
| IVB | Intraventricular block | STTU | ST tilt up |
| JEB | junctional escape beat | TWC | T wave Change |
| JPS | J point shift | TWO | T wave opposite |
| JPT | junctional premature beat | UW | U wave |
| LBBB | left bundle branch block | VB | ventricular bigeminy |
| LBBBB | left back bundle branch block | VEB | ventricular escape beat |
| LFBBB | left front bundle branch block | VFW | ventricular fusion wave |
| LRRI | Long RR interval | VPB | ventricular premature beat |
| LVH | left ventricle hypertrophy | VPE | ventricular preexcitation |
| LVHV | left ventricle high voltage | VET | ventricular escape trigeminy |
| LVQRSAL | lower voltage QRS in all lead | WAVN | Wandering in the atrioventricalualr node |
| LVQRSCL | lower voltage QRS in chest lead | WPW | WPW |

*Εικόνα 12: Ονομασίες και Ακρώνυμα Παθήσεων*

## 4.2 Μεθοδολογία

Αρχικά, έγινε η προαναφερόμενη επεξεργασία των δεδομένων. Μετά την προεπεξεργασία αυτή, ακολούθησε η αναζήτηση των ωφέλιμων χαρακτηριστικών. Οι βαρύτητες του αλγόριθμου Random Forest εμφάνισαν ποια χαρακτηριστικά είχαν μηδενική βαρύτητα, οπότε έγινε ένωση αυτών ως ένα νέο χαρακτηριστικό με ονομασία OTHER. Το χαρακτηριστικό αυτό ουσιαστικά αποτελείται από στήλες που προήλθαν από τον διαχωρισμό της αρχικής στήλης Beat. Στο τέλος αυτής της διαδικασίας, είχαμε 220 χαρακτηριστικά εισόδου αντί 756.

Ύστερα, προκειμένου να βρεθούν οι καλύτερες αποδόσεις, δημιουργήθηκαν πέντε διαφορετικά μοντέλα επί των ίδιων δεδομένων. Για την εκπαίδευσή τους, χρησιμοποιήθηκε το 80% των δειγμάτων (train) και επαληθεύσαμε τα αποτελέσματά τους με το υπόλοιπο 20% (test).

Οι παράμετροι των δεδομένων βρέθηκαν ύστερα από δοκιμές (max_depth=10 για Δέντρο αποφάσεων, max_features =131 για Random Forest και n_neighbors=13 για K-Neighbors) και με τη μέθοδο grid search αντίστοιχα (Lightgbm και XGBoost).

Για την αποφυγή overfitting, δηλαδή τα μοντέλα να έχουν ακριβή αποτελέσματα μόνο για τη συγκεκριμένη βάση δεδομένων, χρησιμοποιήθηκε η τεχνική k-fold validation με k =10.


*Εικόνα 13: 10- fold validation [41]*

Έπειτα, έγινε σύγκριση των επιδόσεών τους για την αξιολόγησή τους και την εύρεση των πιο κατάλληλων μοντέλων. Κύρια μετρική επίδοσης ήταν το F1 score για την γενική απόδοση των μοντέλων και το recall ως συμπληρωματική, καθώς είναι ο λόγος των αληθών θετικών κατηγοριοποιημένων δειγμάτων ως προς το συνολικό αριθμό δειγμάτων που κατηγοριοποιήθηκαν ως θετικά (σύνολο αληθών θετικών και ψευδών θετικών, δηλαδή αρνητικών δειγμάτων που κατηγοριοποιήθηκαν ως θετικά). Αντίστοιχα, ως αληθή αρνητικά χαρακτηρίζονται τα δείγματα που κατηγοριοποιήθηκαν ως αρνητικά και είναι όντως αρνητικά, ενώ ως ψευδή αρνητικά τα δείγματα που κατηγοριοποιήθηκαν ως αρνητικά και στην πραγματικότητα είναι θετικά.

$$Accuracy = \frac{True\ Positive + True\ Negative}{Total\ Population}$$

$$Precision = \frac{True\ Positive}{True\ Positive + False\ Positive}$$

$$Recall = \frac{True\ Positive}{True\ Positive + False\ Negative}$$

$$F_1 = 2 * \frac{Precision\ *\ Recall}{Precision + Recall}$$

*Εικόνα 14: Τυπολόγιο μετρικών επιδόσεων [33]*

Ύστερα, αφού βρέθηκαν τα δύο καλύτερα μοντέλα, χρησιμοποιήθηκε η τεχνική oversampling. Η τεχνική αυτή αποσκοπεί στη δημιουργία ενός πιο ισορροπημένου συνόλου δεδομένων, συμπληρώνοντας μόνο τα δεδομένα εκπαίδευσης με πολλαπλά τυχαία συνθετικά δεδομένα ορισμένων από τις κατηγορίες εξόδου με τον λιγότερο αριθμό δειγμάτων. [34]

Τέλος, ακολούθησε ξανά σύγκριση των τελικών μοντέλων με τον τρόπο που αναφέρθηκε παραπάνω.

## 4.3 Αποτελέσματα και Συγκρίσεις Μοντέλων

Αρχικά, συγκρίνουμε τους αλγορίθμους που χρησιμοποιήθηκαν για την δημιουργία των πέντε μοντέλων. Κάθε αλγόριθμος είχε διαφορετικά αποτελέσματα. Από τη βάση δεδομένων, το 80% των δειγμάτων χρησιμοποιήθηκε ως δεδομένα εκπαίδευσης και το υπόλοιπο 20% ως δεδομένα ελέγχου. Για την εύρεση των καλύτερων μοντέλων, θα συγκρίνουμε τα αποτελέσματα των πινάκων σύγχυσης (confusion matrix) και των αναφορών τους (classification report).

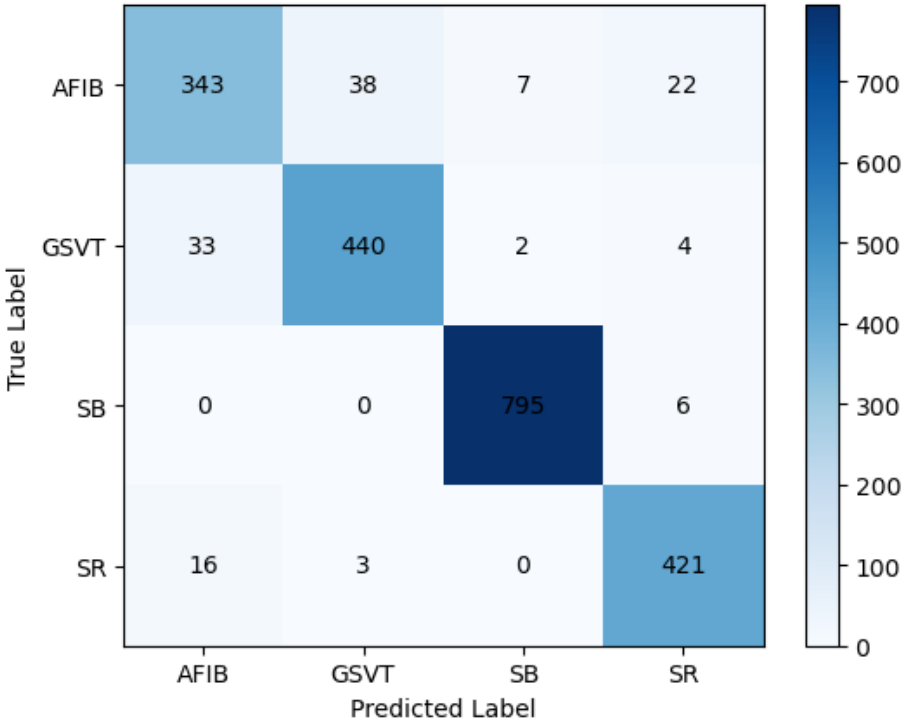
Ένας πίνακας σύγχυσης παρουσιάζει σε μορφή πίνακα τα διαφορετικά αποτελέσματα των προβλέψεων (Predicted Label) και τις πραγματικές ετικέτες (True Label) ενός προβλήματος ταξινόμησης και βοηθά στην απεικόνιση των αποτελεσμάτων αυτών. Σχεδιάζει έναν πίνακα όλων των προβλεπόμενων και πραγματικών τιμών ενός ταξινομητή, συμπληρώνοντας ποιες και πόσες ετικέτες προέβλεψε σωστά.

Μια αναφορά ταξινόμησης χρησιμοποιείται για τη μέτρηση της ποιότητας των προβλέψεων ενός αλγόριθμου ταξινόμησης. Πιο συγκεκριμένα, αφορά πόσες προβλέψεις είναι αληθείς και πόσες είναι ψευδείς και περιλαμβάνει τις μετρικές επίδοσης που προαναφέρθηκαν.

### 4.3.1 Αποτελέσματα Μοντέλων

**LightGBM**

Το πρώτο κατά σειρά μοντέλο που δημιουργήθηκε ήταν το LightGBM. Οι χαμηλότερες μετρικές επίδοσης εμφανίζονται στην κατηγορία AFIB, την οποία συχνότερα αναγνωρίζει ως GSVT. Το F1-score και το Recall στις τρεις υπόλοιπες κατηγορίες είναι ικανοποιητικά, καθώς ξεπερνούν το 90%. Οι χειρότερες συγκριτικά μετρικές εμφανίζονται στην κατηγορία AFIB, με F1-score ίσο με 86% και Recall μόλις 84%. Το γενικό ποσοστό επίδοσης, δηλαδή η ακρίβεια, ισούται με 94%.



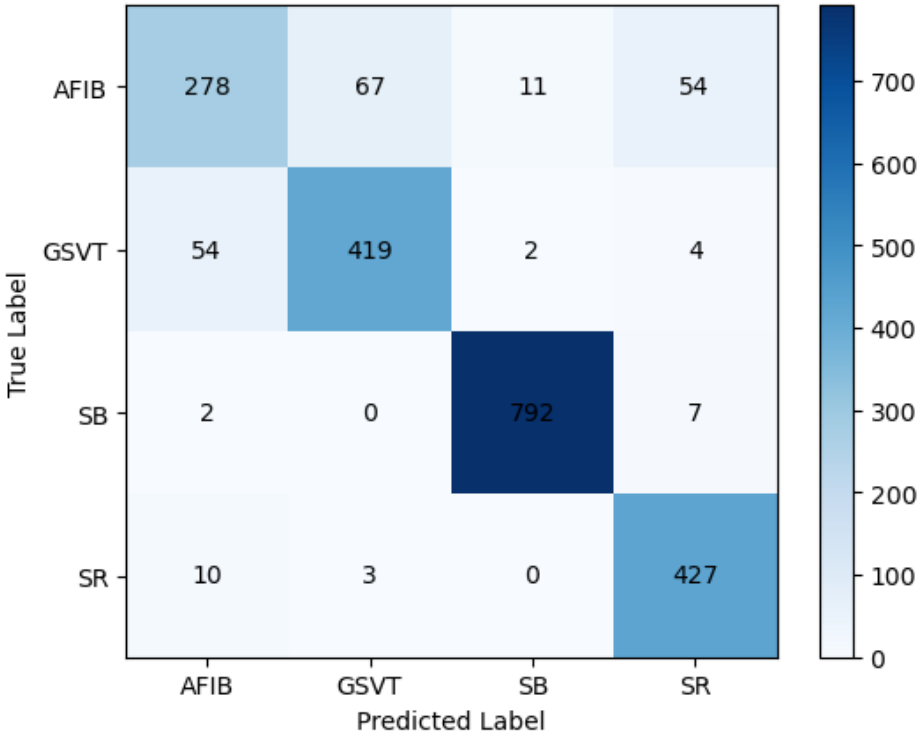

*Εικόνα 15: Πίνακας σύγχυσης LightGBM*

| Lightgbm | Precision | Recall | F1-score |
|---|---|---|---|
| AFIB | 0.88 | 0.84 | 0.86 |
| GSVT | 0.91 | 0.92 | 0.92 |
| SB | 0.99 | 0.99 | 0.99 |
| SR | 0.93 | 0.96 | 0.94 |
| accuracy | - | - | 0.94 |
| macro avg | 0.93 | 0.93 | 0.93 |
| weighted avg | 0.94 | 0.94 | 0.94 |

*Εικόνα 16: Αναφορά Ταξινόμησης LightGBM*

**Δέντρο Αποφάσεων**

Το επόμενο μοντέλο που αναπτύχθηκε είναι το Δέντρο αποφάσεων. Όπως και στο LightGBM, η κατηγορία AFIB συγχέεται με την κατηγορία GSVT πιο συχνά. Οι μετρικές στις κατηγορίες αυτές δεν ξεπερνούν το 87%. Το F1-score και το Recall στις δύο άλλες κατηγορίες είναι ικανοποιητικά, καθώς ξεπερνούν το 90%. Οι χειρότερες συγκριτικά μετρικές εμφανίζονται ξανά στην κατηγορία AFIB, με F1-score ίσο με 74% και Recall μόλις 68%. Η ακρίβεια είναι ίση με 90%.



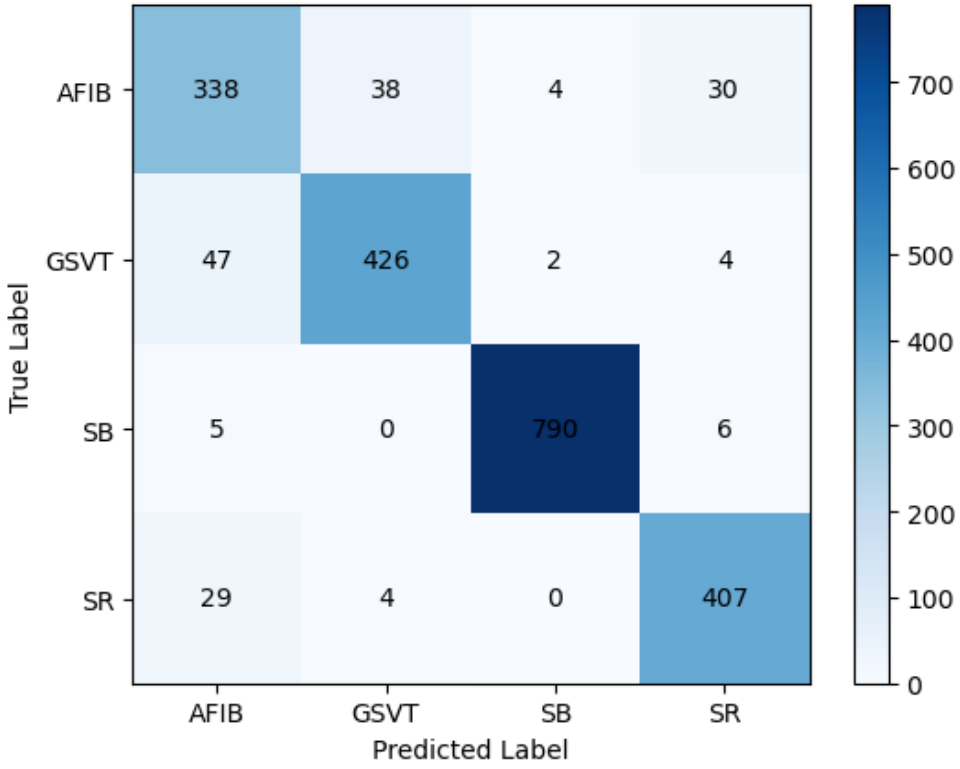

*Εικόνα 17:Πίνακας σύγχυσης Δέντρου Αποφάσεων*

| Δέντρο αποφάσεων | Precision | Recall | F1-score |
|---|---|---|---|
| AFIB | 0.81 | 0.68 | 0.74 |
| GSVT | 0.86 | 0.87 | 0.87 |
| SB | 0.98 | 0.99 | 0.99 |
| SR | 0.87 | 0.97 | 0.92 |
| accuracy | - | - | 0.9 |
| macro avg | 0.88 | 0.88 | 0.88 |
| weighted avg | 0.9 | 0.9 | 0.9 |

*Εικόνα 18: Αναφορά Ταξινόμησης Δέντρου Αποφάσεων*

**Random Forest**

Και στο μοντέλο Random Forest παρατηρείται ότι η κατηγορία AFIB συγχέεται κυρίως με την κατηγορία GSVT. Αυτή τη φορά οι μετρικές της κατηγορίας AFIB είναι μεγαλύτερες του 80%, ενώ της GSVT να πλησιάζουν ή και να ισούνται με 90%. Η κατηγορία AFIB παραμένει η κατηγορία με τις χειρότερες επιδόσεις, καθώς το F1-score και το Recall βρίσκονται στο 82%. Το F1-score και το Recall στις κατηγορίες SB και SR παραμένουν ικανοποιητικά, καθώς ξεπερνούν το 90%. Η ακρίβεια εδώ ισούται με 92%.



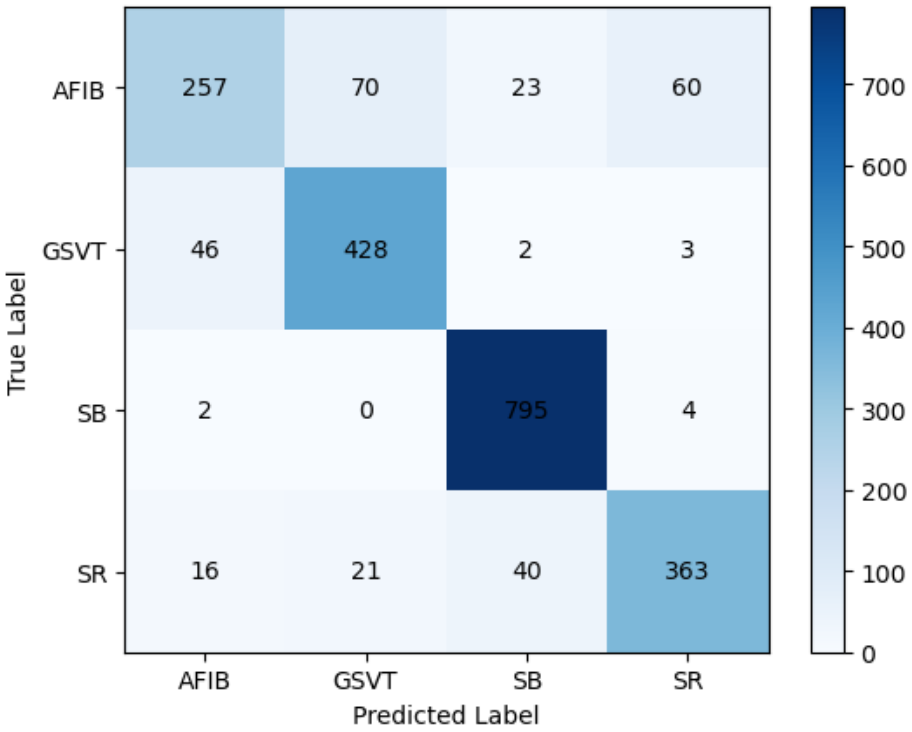

*Εικόνα 19: Πίνακας σύγχυσης Random Forest*

| Random Forest | Precision | Recall | F1-score |
|---|---|---|---|
| AFIB | 0.81 | 0.82 | 0.82 |
| GSVT | 0.91 | 0.89 | 0.9 |
| SB | 0.99 | 0.99 | 0.99 |
| SR | 0.91 | 0.93 | 0.92 |
| accuracy | - | - | 0.92 |
| macro avg | 0.9 | 0.91 | 0.91 |
| weighted avg | 0.92 | 0.92 | 0.92 |

*Εικόνα 20: Αναφορά Ταξινόμησης Random Forest*

**Κ-Γείτονες**

Οι Κ-Γείτονες είναι το μοντέλο με τις χαμηλότερες γενικά επιδόσεις. Συγκεκριμένα, συνεχίζει να συγχέει την κατηγορία AFIB με την κατηγορία GSVT. Επιπλέον, σε αντίθεση με τα παραπάνω μοντέλα, πολλές φορές αναγνωρίζει δείγματα της κατηγορίας SR ως SB. Και εδώ παρατηρούνται οι χαμηλότερες μετρικές επίδοσης στην κατηγορία AFIB, όπου F1-score είναι 70% και Recall 63%. Πλέον μόνο οι μετρικές της κατηγορίας SB είναι μεγαλύτερες του 90%, ενώ η ακρίβεια είναι η μικρότερη από όλα τα μοντέλα και φτάνει μόλις το 87%.



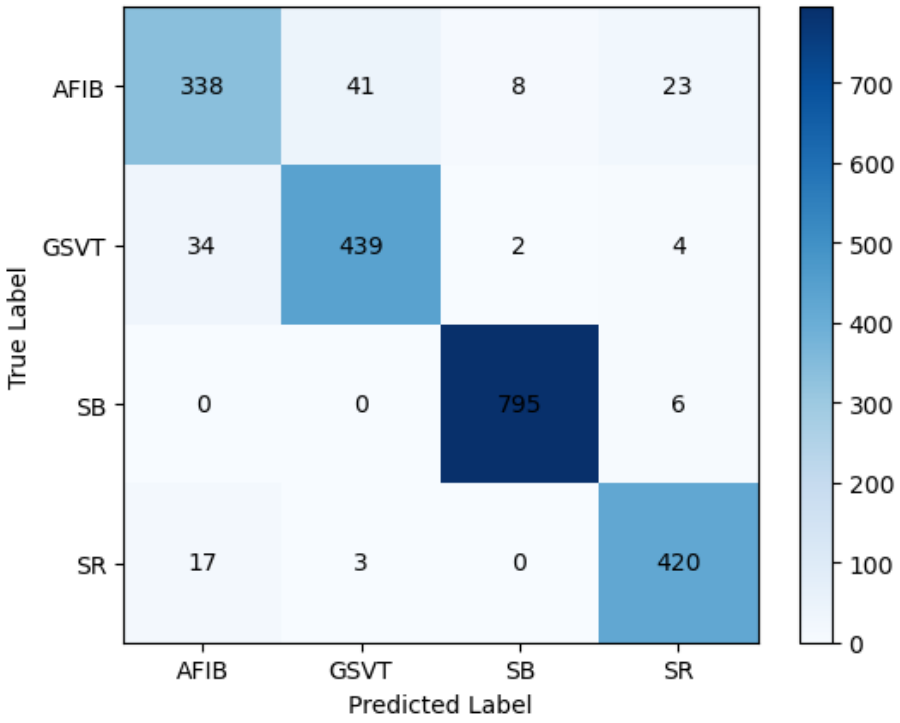

*Εικόνα 21: Πίνακας σύγχυσης Κ-Γειτόνων*

| Κ-Γείτονες | Precision | Recall | F1-score |
|---|---|---|---|
| AFIB | 0.8 | 0.63 | 0.7 |
| GSVT | 0.82 | 0.89 | 0.86 |
| SB | 0.92 | 0.99 | 0.96 |
| SR | 0.84 | 0.82 | 0.83 |
| accuracy | - | - | 0.87 |
| macro avg | 0.85 | 0.83 | 0.84 |
| weighted avg | 0.86 | 0.87 | 0.86 |

*Εικόνα 22: Αναφορά Ταξινόμησης Κ-Γειτόνων*

**XGBoost**

Το τελευταίο μοντέλο που αναπτύχθηκε είναι το XGBoost. Όπως και σε όλα τα προαναφερόμενα μοντέλα, έτσι και εδώ δείγματα της κατηγορίας AFIB κατηγοριοποιούνται μερικές φορές ως GSVT. Η χειρότερη επίδοση εμφανίζεται ξανά στην κατηγορία AFIB, όπου το F1-score ισούται με 85% και το Recall με 82%. Οι μετρικές των τριών άλλων κατηγοριών είναι υψηλότερες του 90%. Η ακρίβεια του μοντέλου ισούται με 94%.




*Εικόνα 23: Πίνακας σύγχυσης XGBoost*

| XGBoost | Precision | Recall | F1-score |
|---|---|---|---|
| AFIB | 0.87 | 0.82 | 0.85 |
| GSVT | 0.91 | 0.92 | 0.91 |
| SB | 0.99 | 0.99 | 0.99 |
| SR | 0.93 | 0.95 | 0.94 |
| accuracy | - | - | 0.94 |
| macro avg | 0.92 | 0.92 | 0.92 |
| weighted avg | 0.93 | 0.94 | 0.93 |

*Εικόνα 24: Αναφορά Ταξινόμησης XGBoost*

### 4.3.2 Επιλογή Μοντέλων

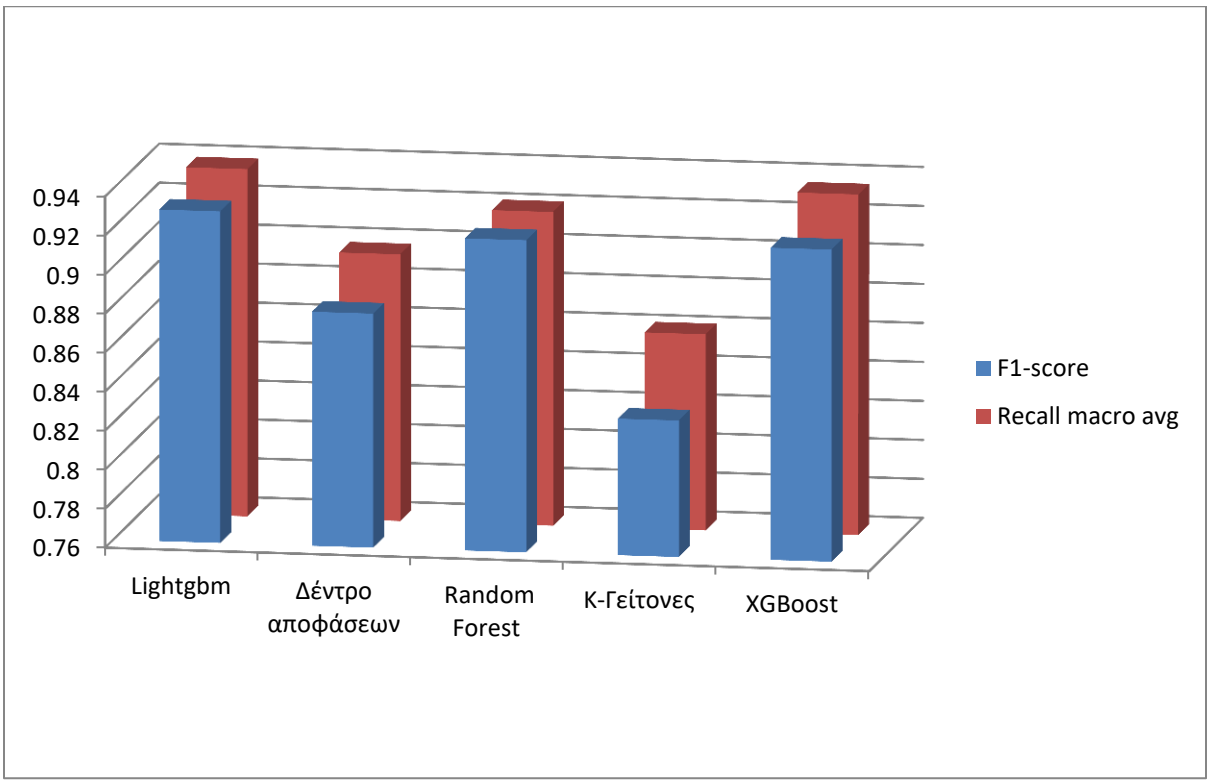
Γενικά παρατηρείται ότι στην κατηγορία SB το Recall είναι σταθερό στο 99% σε όλα τα μοντέλα. Συνεπώς, δεν αποτελεί μέτρο σύγκρισης.

Όπως φαίνεται από τους πίνακες και τα παρακάτω διαγράμματα, το LightGBM είναι το μοντέλο με τις υψηλότερες μετρικές επίδοσης και έχει τα λιγότερα εσφαλμένα δείγματα σε σύγκριση με τα υπόλοιπα μοντέλα.. Συνεπώς, είναι το πρώτο από τα δύο μοντέλα που θα επιλεχθεί για βελτιστοποίηση.

Το μοντέλο των Κ-Γειτόνων απορρίπτεται, καθώς παρουσιάζει τόσο συνολικά όσο και ανά κατηγορία τις χειρότερες μετρικές επίδοσης. Το Δέντρο Αποφάσεων παρουσιάζει μόλις 68% στο Recall στην κατηγορίας AFIB  και έχει επίσης την χειρότερη επίδοση στην κατηγορία GSVT σε σχέση με το Random Forest και το XGBoost.

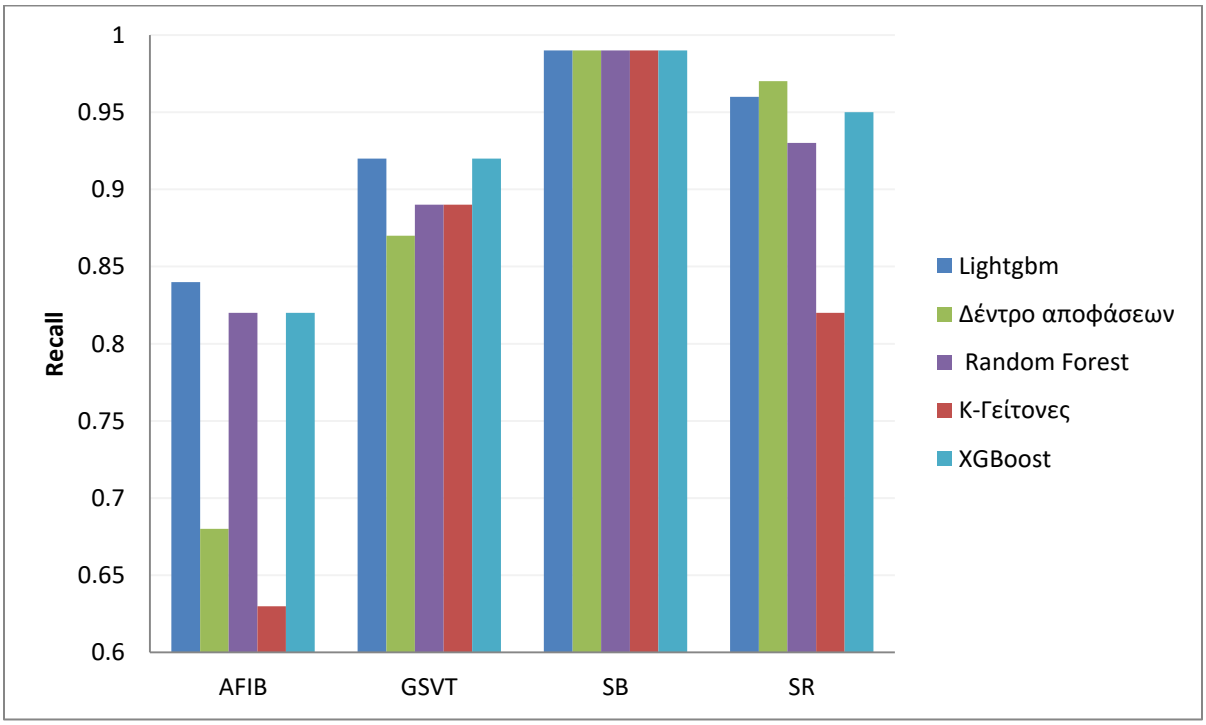
Η επιλογή του δεύτερου ως προς βελτιστοποίηση μοντέλου κείτεται ανάμεσα στο Random Forest και το XGBoost. Έπειτα από σύγκριση των δύο αυτών μοντέλων, καταλήξαμε τα εξής συμπεράσματα:

- Το XGBoost έχει υψηλότερο F1-score από το Random Forest κατά 1.38%.
- Το γενικό Recall είναι κοινό με τιμή 92%, άρα δεν αποτελεί κριτήριο επιλογής.




*Διάγραμμα 3: F1-Score και Recall Μοντέλων*

- Στην κατηγορία AFIB το Random Forest και το XGBoost έχουν ίσο Recall, οπότε δεν αποτελεί μέτρο σύγκρισης.
- Στις κατηγορία GSVT και SR, το XGBoost έχει μεγαλύτερο Recall κατά 3% και 2% αντίστοιχα.




*Διάγραμμα 4: Recall Μοντέλων ανά κατηγορία*

Σύμφωνα με τα παραπάνω συμπεράσματα, καταλήγουμε στην απόφαση ότι το το XGBoost είναι το καταλληλότερο προς βελτιστοποίηση μοντέλο από τα δύο. Συνεπώς, τα μοντέλα που θα χρησιμοποιηθούν είναι το LightGBM και το XGBoost.

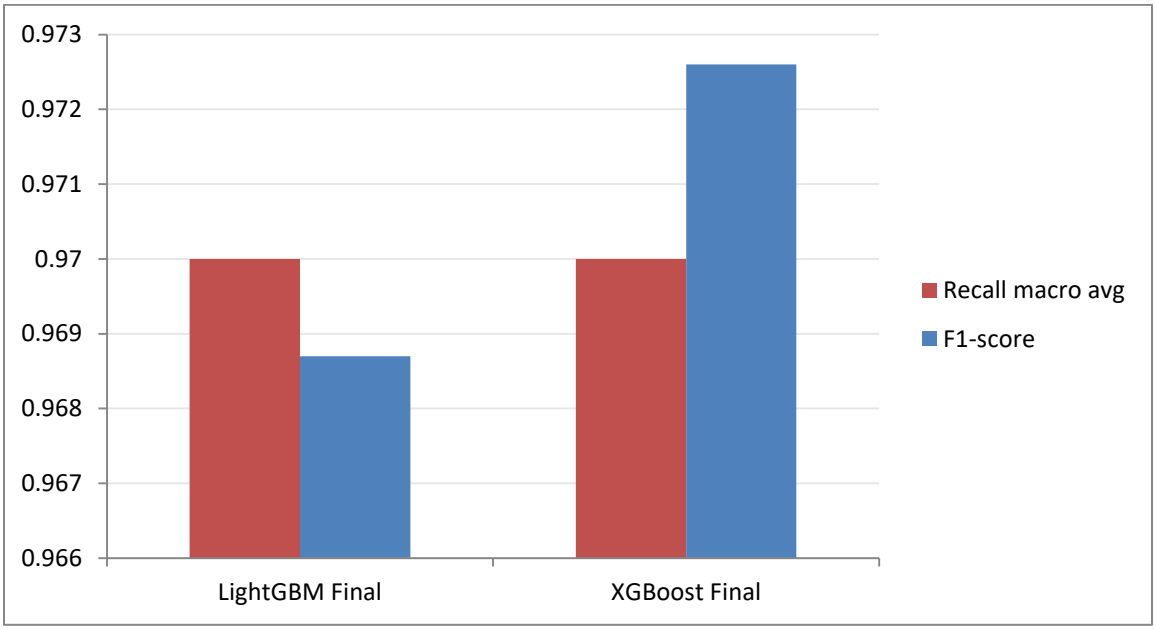
### 4.3.3 Τελικά Μοντέλα

Η τεχνική oversampling κατάφερε να αυξήσει το F1-score, το γενικό Recall και το Recall των τελικών μοντέλων σε όλες τις κατηγορίες, εκτός την SB όπου παρέμεινε σταθερό στο 99% και στα δύο μοντέλα.

Για να επιτευχθεί η βελτιστοποίηση των επιλεγμένων μοντέλων, εφαρμόστηκε η τεχνική oversampling. Μετά την εφαρμογή της, παρατηρείται ότι:

- Η μετρική F1-score στο LightGBM αυξήθηκε κατά 3.87%, το γενικό Recall περίπου κατά 3%.
- Αντίστοιχα στο XGBoost, το F1-score αυξήθηκε κατά 5% και το γενικό Recall περίπου κατά 3.5%.

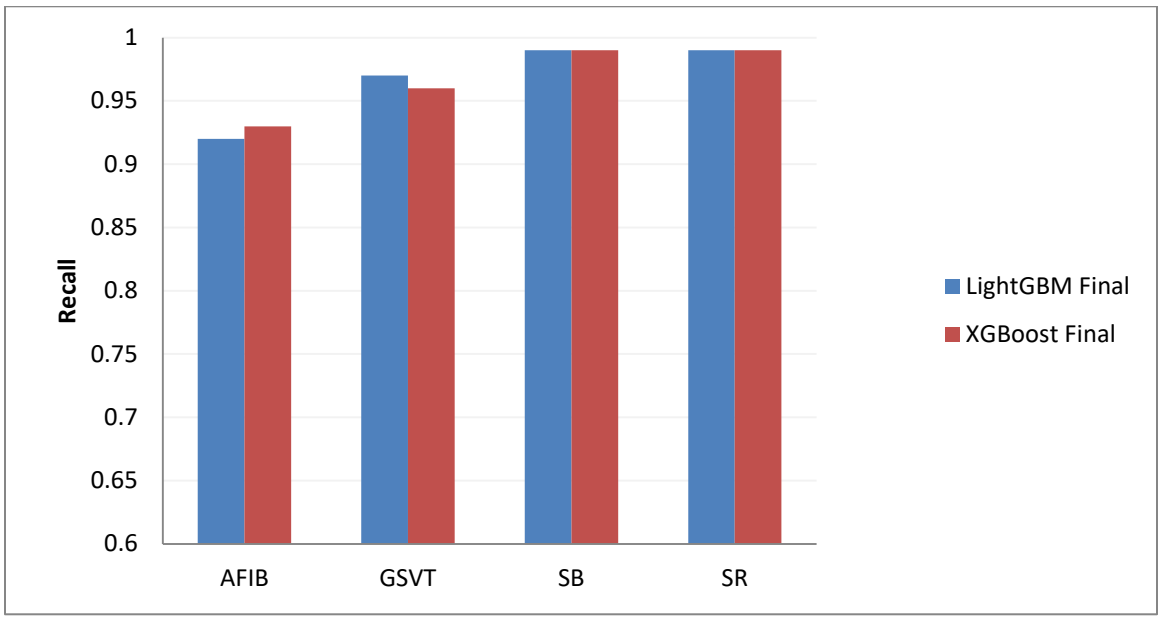
Ύστερα από σύγκριση των δύο μοντέλων, φαίνεται πως έχουν ίδιο γενικό Recall ίσο με 97% Επιπλέον, το F1-score του XGBoost είναι υψηλότερο από εκείνο του LightGBM κατά 0.39%, με τιμές 97.26% και 96.87% αντίστοιχα.




*Διάγραμμα 5: Επιδόσεις Τελικών Μοντέλων*

Συγκεκριμένα, ανά κατηγορία για εκάστοτε μοντέλο παρατηρείται ότι:

- Το Recall στην κατηγορία SB συνεχίζει να παραμένει σταθερό στο 99% και στα δύο μοντέλα.
- Η κατηγορία SR εμφανίζει Recall ίσο επίσης με 99% και στα δύο μοντέλα, παρουσιάζοντας αύξηση 3% F1-score και 4% για το XGBoost.
- Η κατηγορία AFIB στο LightGBM κατά 8% στο 92% και στο XGBoost αυξήθηκε κατά 11% στο 93%.
- Στην κατηγορία GSVT στο LightGBM αυξήθηκε κατά 5% στο 97% και στο XGBoost κατά 4% στο 96%.




*Διάγραμμα 6: Recall Τελικών Μοντέλων ανά κατηγορία*

Έχοντας υπόψη τα παραπάνω διαγράμματα και τις συναφείς παρατηρήσεις τους, καταλήγουμε ότι ο αλγόριθμος XGBoost μετά από oversampling είναι το πιο ικανοποιητικό μοντέλο.

# Επίλογος

Στην παρούσα διπλωματική εργασία παρουσιάστηκε πως η μηχανική μάθηση μπορεί να συμβάλει στον τομέα της υγείας. Συγκεκριμένα, η εφαρμογή αφορούσε την διάγνωση καρδιαγγειακών αρρυθμιών στους ασθενείς. Για τη δημιουργία της χρησιμοποιήθηκαν αλγόριθμοι μηχανικής μάθησης με επίβλεψη. Σκοπός ήταν η ανάπτυξη μοντέλων μηχανικής μάθησης, η βελτιστοποίησή τους και τέλος η εύρεση του πιο αποδοτικού μοντέλου.

Βασική προϋπόθεση για την ανάπτυξη της εφαρμογής ήταν η κατανόηση του θεωρητικού υποβάθρου, τόσο ως προς το κομμάτι της μηχανικής μάθησης όσο και ως προς την ιατρική φύση του προβλήματος. Για το λόγο αυτό αναλύθηκαν αρχικά οι βασικές αρχές της μηχανικής μάθησης και ύστερα συναφείς ιατρικές έννοιες με την εφαρμογή, όπως η ανατομία της καρδιάς, το ηλεκτροκαρδιογράφημα και οι καρδιαγγειακές αρρυθμίες. Επιπλέον, παρουσιάστηκαν ο τρόπος λειτουργίας και τα χαρακτηριστικά γνωρίσματα των αλγορίθμων μηχανικής μάθησης με επίβλεψη που χρησιμοποιήθηκαν.

Προκειμένου να αναπτυχθούν οι αλγόριθμοι, προηγήθηκε επεξεργασία και ανάλυση των δεδομένων βάσης. Καθώς οι αλγόριθμοι μπορούσαν να επεξεργαστούν μόνο αριθμητικά δεδομένα, τροποποιήθηκαν κατάλληλα τα χαρακτηριστικά εισόδου. Επίσης, ομαδοποιήσαμε σε ευρύτερες κατηγορίες τις καρδιαγγειακές αρρυθμίες, οι οποίες αποτελούν την επιθυμητή έξοδο της εφαρμογής. Η αλλαγή αυτή συνέβη στοχεύοντας να είναι πιο ισορροπημένη η αρχική βάση δεδομένων και ακολούθησε η αναζήτηση των ωφέλιμων χαρακτηριστικών.

Από τα αποτελέσματα των μοντέλων που αναπτύχθηκαν συμπεραίνουμε ότι το LightGBM και το XGBoost για την υλοποίηση της ιατρικής εφαρμογής είναι αρκετά ικανοποιητικά. Όμως υπήρχε περιθώριο βελτίωσης εκπαιδεύοντας τα μοντέλα με περισσότερα δεδομένα κυρίως για τις κατηγορίες που υπήρχε μικρότερος αριθμός δειγμάτων. Για τον λόγο αυτό εφαρμόστηκε η μέθοδος oversampling. Ύστερα από αυτή την τροποποίηση, τα τελικά μοντέλα παρουσίασαν σημαντική βελτίωση σε σύγκριση με τα προηγούμενα. Συγκεκριμένα, ο αλγόριθμος XGBoost μετά από oversampling είχε τις καλύτερες επιδόσεις τόσο ανά κατηγορία όσο και συνολικά.

Ένα βασικό πρόβλημα της βάσης δεδομένων ήταν το γεγονός ότι τα δείγματα δεν ήταν ισοκατανεμημένα σε όλες τις κατηγορίες εξόδου, το οποίο είχε επηρέασε τις επιδόσεις και την ακρίβεια των μοντέλων, οι οποίες θα μπορούσαν να είναι πιο υψηλές. Συνεπώς, μία πιθανή βελτίωση έχει ως εξής. Αν η βάση δεδομένων είχε μεγαλύτερη και όσο το δυνατό ισάριθμη ποσότητα δειγμάτων ανά κατηγορία, τα αποτελέσματα θα ήταν πιο ακριβή. Γενικότερα, όταν ο αριθμός δεδομένων μίας βάσης δεδομένων είναι μεγαλύτερος, τα αποτελέσματα είναι καλύτερα σε σύγκριση με μια μικρότερη.

Τέλος, τα χαρακτηριστικά ειδόδου απαιτούσαν περισσότερους υπολογιστικούς πόρους προκειμένου να εκπαιδευτούν οι αλγόριθμοι, ενώ πολλά χαρακτηριστικά είχαν μικρή έως και σχεδόν αμελητέα βαρύτητα και δεν συσχετιζόντουσαν ιδιαίτερα με την έξοδο. Αν η βάση δεδομένων είχε πιο συναφή χαρακτηριστικά εισόδου, η διαδικασία εκπαίδευσης των αλγορίθμων θα ήταν λιγότερο χρονοβόρα, καθώς θα απαιτούσαν λιγότερη υπολογιστική ισχύ.

# Παράρτημα Α: Ακρωνύμια και συντομογραφίες

| | |
|---|---|
| AF: | Atrial Flutter (Κολπικός Πτερυγισμός) |
| AFIB: | Atrial Fibrillation (Κολπική Μαρμαρυγή) |
| AI: | Artificial intelligence (Τεχνητή Νοημοσύνη) |
| AT: | Atrial Tachycardia (Κολπικά Ταχυκαρδία) |
| AVRT: | Atrioventricular Reentrant Tachycardia (Κολποκοιλιακή Ταχυκαρδία Επανεισόδου) |
| AVNRT: | Atrioventricular Node Reentrant Tachycardia (Κομβική Ταχυκαρδία) |
| CVD : | Cardiovascular Diseases (Καρδιαγγειακές Παθήσεις) |
| DL: | Deep learning (Βαθιά μάθηση) |
| ECG: | Electrocardiogram (ΗΚΓ / Ηλεκτροκαρδιογράφημα) |
| GPU: | Graphics Processing Unit (Κάρτα Γραφικών) |
| KNN: | K-Nearest Neighbors (Κ-Γείτονες) |
| ML: | Machine learning (Μηχανική Μάθηση) |
| SAAWR: | Sinus Atrium to Atrial Wandering Rhythm (Ρυθμός Του Καρδιακού Παλμού Που Εξελίσσεται σε Περιπλανώμενο Κολπικό Βηματοδότη) |
| SA: | Sinus Arrhythmia (Φλεβοκομβική Αρρυθμία) |
| SB: | Sinus Bradycardia (Φλεβοκομβική Βραχυκαρδία) |
| SI: | Sinus Irregularity (Φλεβοκομβική Ανωμαλία) |
| SR: | Sinus Rhythm (Φλεβοκομβικός Ρυθμός) |
| ST: | Sinus Tachycardia (Φλεβοκομβική Ταχυκαρδία) |
| SVT: | Supraventricular Tachycardia (Υπερκοιλιακή Ταχυκαρδία) |
| XGB: | Extreme Gradient Boosting |

# Παράρτημα Β: Κώδικας Python εύρεσης καρδιακής αρρυθμίας

## Προετοιμασία με φόρτωση απαραίτητων βιβλιοθηκών

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
%matplotlib inline
from sklearn.model_selection import train_test_split
from sklearn import tree
from sklearn.neighbors import KNeighborsClassifier
import xgboost
from xgboost import XGBClassifier
import lightgbm as ltb
from lightgbm import LGBMClassifier
from sklearn.model_selection import GridSearchCV
from sklearn.ensemble import RandomForestClassifier
from sklearn.metrics import accuracy_score
from sklearn.metrics import confusion_matrix
from sklearn.metrics import precision_score
from sklearn.metrics import recall_score
from sklearn.metrics import f1_score
from sklearn.metrics import classification_report
from sklearn.model_selection import cross_val_score
from sklearn.preprocessing import LabelEncoder
import imblearn
from imblearn.over_sampling import RandomOverSampler
```

## Εισαγωγή δεδομένων

```
data= pd.read_csv('C:/Users/riaga/Desktop/Diagnostics.csv')
```

## Πληροφορίες δεδομένων

*Εμφάνιση όλων των γραμμών και διαστάσεων πίνακα (γραμμές=δείγματα, στήλες=χαρακτηριστικά)*

```
data.head(10646)
```

*Αφαίρεση στήλης FileName, καθώς δεν προσδίδει χρήσιμη πληροφορία*

```
data=data.drop(data.columns[0], axis=1)

data.shape
```

*Εμφάνιση στατιστικών στοιχείων δεδομένων*

```
data.describe(include='all') #Οι μη αριθμητικές τιμές εμφανίζονται ως NaN
```

*Εμφάνιση ονομάτος στηλών (χαρακτηριστικών)*
```
data.columns
```

*Εμφάνιση δεικτών και αρίθμηση δειγμάτων*
```
data.index
```

*Πληροφορίες για τύπους δεδομένων*
```
data.info

data.dtypes
```

*Καταμέτρηση στοιχείων ανά χαρακτηριστικό*
```
data.count()
```

*Καταγραφή μοναδικών τιμών ανά χαρακτηριστικό*
```
cols=data.columns
for i in cols:
   print(i)
   print(data[i].unique())
```

*Πλήθος μοναδικών τιμών για κάθε χαρακτηριστικό*
```
for i in cols:
   print(i)
   print(data[i].value_counts())
```

*Έλεγχος για τιμές χωρίς καταγραφή*
```
data.isnull().any()

data.isnull().sum()
```

## Επεξεργασία δεδομένων

*Μετατροπή ταμπελών εξόδου*
```
data.Rhythm.value_counts()

data.Rhythm = data.Rhythm.replace(['SB', 'SR', 'AFIB', 'ST', 'SVT', 'AF', 'SA', 'AT', 'AVNRT',
'AVRT', 'SAAWR'],
                                  ['SB', 'SR', 'AFIB', 'GSVT', 'GSVT', 'AFIB', 'SR', 'GSVT', 'GSVT',
'GSVT', 'GSVT'])

data.Rhythm.value_counts()

data.Rhythm.head()
```

*Χρήση dummy για Gender*
```
dummy1=pd.get_dummies(data['Gender'])
dummy1.head
```

*Χρήση dummy για Beat*
```
dummy2=pd.get_dummies(data['Beat'])
dummy2.head
```

```
Ενσωμάτωση dummies στο dataframe
data= pd.concat([data, dummy1, dummy2], axis=1)

data=data.drop(['Gender', 'Beat'], axis=1)

data.shape

print(data.columns.tolist())

Ομαδοποίηση χαρακτηριστικών που θα χρησιμοποιήσουμε ως εισόδους και εξόδους στη μοντελοποίηση
feature1 = ['PatientAge', 'VentricularRate', 'AtrialRate', 'QRSDuration', 'QTInterval', 'QTCorrected',
        'RAxis','TAxis', 'QRSCount', 'QOnset', 'QOffset', 'TOffset'] #χαρακτηριστικά - είσοδοι με αριθμητικές τιμές
target = ['Rhythm']

test=data.columns

feature2 = []

for element in test:

   if element not in feature1:

      feature2.append(element)

feature2.remove("Rhythm")

features = feature1 + feature2
```

# Μελέτη δεδομένων

## Πίνακες

```
Πίνακας δεδομένων ατόμων που έχουν αυξημένο αρτηριακό ρυθμό ανάλογα με ηλικία (>μέσου όρου δειγμάτων)
data[(data.PatientAge>data.PatientAge.mean()) & (data.AtrialRate>=120)]

Πίνακας δεδομένων ατόμων που το QTIntervalείναι μεγαλύτερο από το QTCorrected
data[(data.QTInterval>data.QTCorrected) ]
```

## Ιστογράμματα & Γραφήματα

```
Ιστόγραμμα μεταβητών εισόδου (feature1)
data[feature1].hist(bins=25, figsize=(15, 15))

Διάγραμμα διασποράς ηλικίας με ρυθμών που θεωρητικά επηρεάζονται από αυτήν
data.plot.scatter(x='PatientAge',y='VentricularRate',title='Διάγραμμα διασποράς των PatientAge και VentricularRate')
data.plot.scatter(x='PatientAge',y='AtrialRate',title='Διάγραμμα διασποράς των PatientAge και AtrialRate')
```

*Διάγραμμα διασποράς ρυθμών*

```
data.plot.scatter(x='AtrialRate',y='VentricularRate',title='Διάγραμμα διασποράς των AtrialRate και VentricularRate')
```

*Διάγραμμα διασποράς offset και Qoffset-Qonset*

```
data.plot.scatter(x='QOffset',y='QOnset',title='Διάγραμμα διασποράς των QOffset και QOnset')
data.plot.scatter(x='QOffset',y='TOffset',title='Διάγραμμα διασποράς των QOffset και TOffset')
```

*Διάγραμμα QTInterval και QTCorrected*

```
data.plot.scatter(x='QTInterval',y='QTInterval',title='Διάγραμμα διασποράς των QTInterval και QTInterval')
```

# Εύρεση ωφέλιμων χαρακτηριστικών

## Διαχωρισμός σε δεδομένα εκπαίδευσης και δεδομένα ελέγχου

```
X=data[features]
X.shape
X.head()

y=data.Rhythm
y.shape
y.head(10)

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size = 0.2, random_state=4)

print("X_train διαστάσεις:", X_train.shape)
print("y_train διαστάσεις:", y_train.shape)
print("X_test διαστάσεις:", X_test.shape)
print("y_test διαστάσεις:", y_test.shape)
```

## Βαρύτητα μέσω των importances του Random Forest

*Βέλτιστο max_features*

```
R=[]
for i in range (1,757):
    rfc = RandomForestClassifier(n_estimators=20,max_features = i)
    rfc.fit(X_train, y_train)
    y_pred = rfc.predict(X_test)
    f1 = f1_score(y_test, y_pred, average='weighted')
    r = int(i)
    R.append(f1)
    print(r , f1)

max_features = R.index(max(R))
print('Μέγιστη τιμή έχουμε στο Features', max_features, ',άρα στο', max_features+1, 'χαρακτηριστικο με τιμή', max(R))
```

*Εφαρμογή αλγορίθμου*

```
rfc = RandomForestClassifier(n_estimators=10,max_features = max_features+1)

rfc.fit(X_train, y_train)

y_pred = rfc.predict(X_test)

rfc.feature_importances_

imp=rfc.feature_importances_
```

*Εύρεση πλήθους χαρακτηριστικών με βαρύτητα 0 (συμβ. 0.00000000e+00)*

```
count = np.count_nonzero(imp == 0.00000000e+00)
number=756-count

print('Η τιμή 0 εμφανίζεται σε', count, 'χαρακτηριστικά, άρα θα χρησιμοποιήσουμε', number,
'χαρακτηριστικά')
```

*Εμφάνιση index ωφέλιμων χαρακτηριστικών*

```
ind = (-imp).argsort()[:number]
print(ind)
```

*Εύρεση ταμπελών ωφέλιμων χαρακτηριστικών (στήλες)*

```
data.iloc[:, ind]

df=data.iloc[:, ind]

df.columns
```

*Ένωση λοιπών χαρακτηριστικών με βαρύτητα 0 ως στήλη Other*

```
df2 = data.drop(data.columns[ind],axis = 1)

df2.columns

def check_columns(df2):
  df2['OTHER'] = 0
  for index, row in df2.iterrows():
     for col in row:
     if col == 1:
       df2.at[index, 'OTHER'] = 1
       break

check_columns(df2)

df2.columns

df2 = df2.loc[:, ['OTHER']]
df2
```

*Ορισμός τελικών χαρακτηριστικών εισόδου*

```
test=pd.concat([df, df2], axis=1)

test=test.drop(['Rhythm'], axis=1)
```

```
feature= test.columns.tolist()
print(feature)
```

# Εύρεση καταλληλότερων αλγορίθμων

## Διαχωρισμός σε δεδομένα εκπαίδευσης και δεδομένα ελέγχου

```
X=test[feature]
X.shape
X.head()

y=data.Rhythm
y.shape
y.head()

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size = 0.2, random_state=4)

print("X_train διαστάσεις:", X_train.shape)
print("y_train διαστάσεις:", y_train.shape)
print("X_test διαστάσεις:", X_test.shape)
print("y_test διαστάσεις:", y_test.shape)

target_names = ['AFIB', 'GSVT', 'SB','SR']
```

## Μοντέλα

### Μοντέλο 1: Lightgbm

```
param_grid = {
    'learning_rate': [0.1, 0.01],
    'max_depth': [1, 5, 10],
    'n_estimators': [50,100,200]
}

lgb=ltb.LGBMClassifier()

grid_search = GridSearchCV(
    estimator=lgb,
    param_grid=param_grid,
    scoring='accuracy',
    cv=10,
    )

grid_search.fit(X_train, y_train)

grid_search.best_estimator_

lg=grid_search.best_estimator_

lg.fit(X_train, y_train)
```

*Αξιολόγηση μοντέλου με είσοδο τα δεδομένα ελέγχου*

```
y_pred = lg.predict(X_test)
```

```
# Αξιολόγηση μοντέλου με είσοδο τα δεδομένα ελέγχου
y_pred = lg.predict(X_test)

print("Ποσοστό επιτυχίας κατηγοριοποίησης:")
print(np.sum(y_pred == y_test) / float(len(y_test)))

scores = cross_val_score(lg, X_train, y_train, cv=10, scoring = "f1_weighted")
print("Scores:", scores)
print("Mean:", scores.mean())

# Μετρική Επίδοσης μοντέλου κατηγοριοποίησης 1
confusion_matrix(y_test, y_pred)

f1_score(y_test, y_pred, average=None)

print(classification_report(y_test, y_pred, target_names=target_names))

epidosi= cross_val_score(lg, X, y, cv=10)
print("Επιδόσεις με :", epidosi)

print("Μέση επίδοση με :",epidosi.mean())

a= f1_score(y_test, y_pred, average='weighted')
print (a)
```

### Μοντέλο 2: Δέντρο αποφάσεων

```
Ra=[]
for i in range (1,30):
    dtr = tree.DecisionTreeClassifier(criterion="entropy", max_depth=i)
    dtr.fit(X_train, y_train)
    y_pred = dtr.predict(X_test)
    f1 = f1_score(y_test, y_pred, average='weighted')
    A = int(i)
    Ra.append(f1)
    print(A , f1)

max_depth =  Ra.index(max(Ra))
print('Μέγιστη τιμή έχουμε στο Index', max_depth, ',άρα στους', max_depth+1, 'max_depth
με τιμή', max(Ra))

# Επιλογή βέλτιστου max_depth για εφαρμογή μοντέλου
dtr = tree.DecisionTreeClassifier(max_depth=max_depth+1)

dtr.fit(X_train, y_train)

# Οπτικοποίηση Δέντρου Απόφασης
plt.figure(figsize=(90, 90))
tree.plot_tree(dtr.fit(X_train, y_train))

# Αξιολόγηση μοντέλου με είσοδο τα δεδομένα ελέγχου
y_pred = dtr.predict(X_test)
```

```
print("Ποσοστό επιτυχίας κατηγοριοποίησης:")
print(np.sum(y_pred == y_test) / float(len(y_test)))

scores = cross_val_score(dtr, X_train, y_train, cv=10, scoring = "f1_weighted")
print("Scores:", scores)
print("Mean:", scores.mean())
```

*Μετρική Επίδοσης μοντέλου κατηγοριοποίησης 2*

```
confusion_matrix(y_test, y_pred)

f1_score(y_test, y_pred, average=None)

print(classification_report(y_test, y_pred, target_names=target_names))

epidosi = cross_val_score(dtr, X, y, cv=10)
print("Επιδόσεις με Cross-validation: ",epidosi)

print("Μέση επίδοση με cross-validation:",epidosi.mean())

b= f1_score(y_test, y_pred, average='weighted')
print (b)
```

### Μοντέλο 3: Random Forest

```
Rb=[]
for i in range (1,number):
    rfc = RandomForestClassifier(n_estimators=20,max_features = i)
    rfc.fit(X_train, y_train)
    y_pred = rfc.predict(X_test)
    f1 = f1_score(y_test, y_pred, average='weighted')
    B = int(i)
    Rb.append(f1)
    print(B , f1)

max_features = Rb.index(max(Rb))
print('Μέγιστη τιμή έχουμε στο Features', max_features, ',άρα στο', max_features+1,
'χαρακτηριστικο με τιμή', max(Rb))
```

*Επιλογή βέλτιστου max_features για εφαρμογή μοντέλου*

```
rfc = RandomForestClassifier(n_estimators=10,max_features = max_features+1)

rfc.fit(X_train, y_train)
```

*Αξιολόγηση μοντέλου με είσοδο τα δεδομένα ελέγχου*

```
y_pred = rfc.predict(X_test)

print("Ποσοστό επιτυχίας κατηγοριοποίησης:")
print(np.sum(y_pred == y_test) / float(len(y_test)))

scores = cross_val_score(rfc, X_train, y_train, cv=10, scoring = "f1_weighted")
print("Scores:", scores)
print("Mean:", scores.mean())
```

*Μετρική Επίδοσης μοντέλου κατηγοριοποίησης 3*

```
confusion_matrix(y_test, y_pred)

f1_score(y_test, y_pred, average=None)

print(classification_report(y_test, y_pred, target_names=target_names))

epidosi = cross_val_score(rfc, X, y, cv=10)
print("Επιδόσεις με Cross-validation: ",epidosi)

print("Μέση επίδοση με cross-validation:",epidosi.mean())

c= f1_score(y_test, y_pred, average='weighted')
print(c)
```

### Μοντέλο 4: K - Neighbors

```
Rc=[]
for i in range (1,30):
    Kn = KNeighborsClassifier(n_neighbors=i)
    Kn.fit(X_train, y_train)
    y_pred = Kn.predict(X_test)
    f1 = f1_score(y_test, y_pred, average='weighted')
    C = int(i)
    Rc.append(f1)
    print(C , f1)

max_index =  Rc.index(max(Rc))
print('Μέγιστη τιμή έχουμε στο Index', max_index, ',άρα στους', max_index+1, 'γείτονες με τιμή', max(Rc))
```

*Επιλογή βέλτιστου γείτονα για εφαρμογή μοντέλου*

```
knn = KNeighborsClassifier(n_neighbors=max_index+1)

knn.fit(X_train, y_train)
```

*Αξιολόγηση μοντέλου με είσοδο τα δεδομένα ελέγχου*

```
y_pred = knn.predict(X_test)

print("Ποσοστό επιτυχίας κατηγοριοποίησης:")
print(np.sum(y_pred == y_test) / float(len(y_test)))

scores = cross_val_score(knn, X_train, y_train, cv=10, scoring = "f1_weighted")
print("Scores:", scores)
print("Mean:", scores.mean())
```

*Μετρική Επίδοσης μοντέλου κατηγοριοποίησης 4*

```
confusion_matrix(y_test, y_pred)

f1_score(y_test, y_pred, average=None)

print(classification_report(y_test, y_pred, target_names=target_names))
```

```
epidosi= cross_val_score(knn, X, y, cv=10)
print("Επιδόσεις με :", epidosi)

print("Μέση επίδοση με :", epidosi.mean())

d= f1_score(y_test, y_pred, average='weighted')
print (d)
```

### Μοντέλο 5: XGBoost

```
params={
   'learning_rate' : [0.01, 0.1],
   'max_depth' : [2, 4, 6],
   'n_estimators' : [150, 200],
   'n_jobs': [2, 5]
}

xgboo=xgboost.XGBClassifier()

grid_search = GridSearchCV(
   estimator=xgboo,
   param_grid=params,
   scoring='accuracy',
   cv=10
   )

le = LabelEncoder()
y_train = le.fit_transform(y_train)
y_test=le.fit_transform(y_test)
y=le.fit_transform(y)

grid_search.fit(X_train, y_train)

grid_search.best_estimator_

xgb=grid_search.best_estimator_

xgb.fit(X_train, y_train)
```

*Αξιολόγηση μοντέλου με είσοδο τα δεδομένα ελέγχου*

```
y_pred = xgb.predict(X_test)

print("Ποσοστό επιτυχίας κατηγοριοποίησης:")
print(np.sum(y_pred == y_test) / float(len(y_test)))

scores = cross_val_score(xgb, X_train, y_train, cv=10, scoring = "f1_weighted")
print("Scores:", scores)
print("Mean:", scores.mean())
```

*Μετρική Επίδοσης μοντέλου κατηγοριοποίησης 5*

```
confusion_matrix(y_test, y_pred)

f1_score(y_test, y_pred, average=None)

print(classification_report(y_test, y_pred, target_names=target_names))
```

```
print("Επιδόσεις με :", epidosi)

print("Μέση επίδοση με :",epidosi.mean())

e= f1_score(y_test, y_pred, average='weighted')
print (e)
```

## Επιλογή μοντέλων προς βελτιστοποίηση

```
f1 = [a, b, c, d, e]
f1.sort(reverse=True)
print (f1)
print ('a =', a)
print ('b =', b)
print ('c =', c)
print ('d =', d)
print ('e =', e)
print ('Τα μέγιστο f1 είναι στο', max(f1),'και στο', f1[1],'.')
```

## Χρήση Oversampling για βελτιστοποίηση

```
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size = 0.2, random_state=4)
```

*Εύρεση βέλτιστου random_state για Oversampling για κάθε μοντέλο*

```
models = []
list_XBGM=[]
list_LBGM=[]
for i in range(100):
    ros = RandomOverSampler(random_state=i)
    X_resampled, y_resampled = ros.fit_resample(X, y)
    X_train, X_test, y_train, y_test = train_test_split(X_resampled, y_resampled, test_size
=0.2,  random_state=4)
    le = LabelEncoder()
    y_train = le.fit_transform(y_train)
    y_test=le.fit_transform(y_test)
    xgb_base = xgboo
    xgb_base.fit(X_train, y_train)
    models.append(('xgb_base with oversample for everything', xgb_base))
    y_pred=xgb_base.predict(X_test)
    b=np.sum(y_pred == y_test) / float(len(y_test))
    list_XBGM.append(b)
    lgb_base = lgb
    lgb_base.fit(X_train, y_train)
    models.append(('lgb_base1 with oversample for everything', lgb_base))
    y_pred=lgb_base.predict(X_test)
    v=np.sum(y_pred == y_test) / float(len(y_test))
    list_LBGM.append(v)
max_XBGM =  list_XBGM.index(max(list_XBGM))
print(f"the best XGBM is at index {max_XBGM} with value {max(list_XBGM)}")
max_LBGM =  list_LBGM.index(max(list_LBGM))
print(f"the best LGBM is at index {max_LBGM} with value {max(list_LBGM)}")
```

*Εφαρμογή βέλτιστου random_state για κάθε μοντέλο*

```
models = []
ros = RandomOverSampler(random_state=max_XBGM)
X_resampled, y_resampled = ros.fit_resample(X, y)
X_train, X_test, y_train, y_test = train_test_split(X_resampled, y_resampled, test_size = 0.2,
random_state=4)
le = LabelEncoder()
y_train = le.fit_transform(y_train)
y_test=le.fit_transform(y_test)
xgb_base = xgboo
xgb_base.fit(X_train, y_train)
y_pred= xgb_base.predict(X_test)
xgboos= f1_score(y_test, y_pred, average='weighted')
models.append(('xgb_base with oversampling', xgb_base))

ros = RandomOverSampler(random_state=max_LBGM)
X_resampled, y_resampled = ros.fit_resample(X, y)
X_train, X_test, y_train, y_test = train_test_split(X_resampled, y_resampled, test_size = 0.2,
random_state=4)
le = LabelEncoder()
y_train = le.fit_transform(y_train)
y_test=le.fit_transform(y_test)
lgb_base = lgb
lgb_base.fit(X_train, y_train)
y_pred= lgb_base.predict(X_test)
light= f1_score(y_test, y_pred, average='weighted')
models.append(('lgb_base1 with oversampling', lgb_base))
for name, model in models:
    print(name)
    y_pred=model.predict(X_test)
    print(classification_report(y_test, y_pred, target_names=target_names))
```

## Τελική επιλογή

```
f = [xgboos, light]
f.sort(reverse=True)
print (f)
print ('Τα μέγιστο f1 είναι το', max(f), '.')
```

# Βιβλιογραφία


1. Mitchell, Tom (1997). Machine Learning. New York: McGraw Hill. ISBN 0-07-042807-7. OCLC 36417892.
2. Mahesh, Batta. "Machine learning algorithms-a review." International Journal of Science and Research (IJSR). [Internet] 9 (2020): 381-386.
3. Cios, Krzysztof J., et al. "Unsupervised learning: association rules." Data Mining: A Knowledge Discovery Approach (2007): 289-306.
4. Παρτάλας, Ι. (2009). Μέθοδοι ενισχυτικής μάθησης σε συστήματα πρακτόρων (Doctoral dissertation, Αριστοτέλειο Πανεπιστήμιο Θεσσαλονίκης (ΑΠΘ). Σχολή Θετικών Επιστημών. Τμήμα Πληροφορικής).
5. Dayan, P., & Niv, Y. (2008). Reinforcement learning: the good, the bad and the ugly. Current opinion in neurobiology, 18(2), 185-196.
6. Graczyk, M., Lasota, T., Trawiński, B., & Trawiński, K. (2010). Comparison of bagging, boosting and stacking ensembles applied to real estate appraisal. In Intelligent Information and Database Systems: Second International Conference, ACIIDS, Hue City, Vietnam, March 24-26, 2010. Proceedings, Part II 2 (pp. 340-350). Springer Berlin Heidelberg.
7. Natekin, A., & Knoll, A. (2013). Gradient boosting machines, a tutorial. Frontiers in neurorobotics, 7, 21.
8. FEDESORIANO (2017, September 3). *Heart Failure Prediction Dataset*. Kaggle. https://www.kaggle.com/datasets/fedesoriano/heart-failure-prediction
9. Oyeleye M, Chen T, Titarenko S, Antoniou G. A Predictive Analysis of Heart Rates Using Machine Learning Techniques. Int J Environ Res Public Health. 2022 Feb 19;19(4):2417. doi: 10.3390/ijerph19042417. PMID: 35206603; PMCID: PMC8872524.
10. Chen, C. Ascent of machine learning in medicine. https://www.nature.com/articles/s41563-019-0360-1
11. The Texas Heart Institute .Heart Anatomy. https://www.texasheart.org/heart-health/heart-information-center/topics/heart-anatomy/
12. Cleveland clinic (2021, August 26). Heart. https://my.clevelandclinic.org/health/body/21704-heart
13. Upasana Mishra and Love Verma. Noise removal from ecg signal by thresholding with comparing different types of wavelet. International Journal of Application or Innovation in engineering and Management, 3(3), 2014.
14. Madona, P., Basti, R. I., & Zain, M. M. (2021). PQRST wave detection on ECG signals. Gaceta Sanitaria, 35, S364-S369.
15. Isin, A., & Ozdalili, S. (2017). Cardiac arrhythmia detection using deep learning. Procedia computer science, 120, 268-275.
16. C Saritha, V Sukanya, and Y Narasimha Murthy. Ecg signal analysis using wavelet transforms. Bulg. J. Phys, 35(1):68–77, 2008
17. Incardiology. Φλεβοκομβική βραδυκαρδία. http://incardiology.gr/arr_arrithmies_yperkoiliakes_flev_brady.html

18. Αθανασίου , Μ., & Βαχλιώτη , Β. (2018, June 22). *Τι είναι η κολπική μαρμαρυγή*. Doctoranytime. https://www.doctoranytime.gr/p/kolpiki-marmarygi
19. Πισσαρίδης, Κ. (2017, September 3). *ΤΑΧΥΚΑΡΔΙΑ (ΦΥΣΙΟΛΟΓΙΚΗ-ΠΑΘΟΛΟΓΙΚΗ)*. καρδιολογοσπισσαριδησκ. https://www.καρδιολογοσπισσαριδησκ.gr/2017/09/03/ταχυκαρδια-φυσιολογικη-παθολογικη/
20. World Health Organization. (2010). International statistical classification of diseases and related health problems (11th ed.). https://icd.who.int/browse10/2010/en
21. Πισσαρίδης, Κ. (2017, December 6). *ΟΙ ΑΡΡΥΘΜΙΕΣ*. καρδιολογοσπισσαριδησκ. https://www.καρδιολογοσπισσαριδησκ.gr/2017/12/06/οι-αρρυθμιεσ/
22. Δασκαλόπουλος , Δ. Α. (2014, January 1). *Υπερκοιλιακή ταχυκαρδία (SVT)*. Athenspedcard. https://www.athenspedcard.com/gia-goneis-and-astheneis/gnoseis-kai-plerophories/paidia-kai-arruthmia/tupoi-arruthmias-sta-paidia/takhukardia/uperkoiliake-takhukardia-svt
23. Kashou, Anthony H.; Basit, Hajira; Chhabra, Lovely (2021), "Physiology, Sinoatrial Node", *StatPearls*, Treasure Island (FL): PMID 29083608, retrieved 2021-11-17
24. Κοντογεώργης, Α. *Υπερκοιλιακή ταχυκαρδία*. Cardiorhythm. https://cardiorhythm.gr/υπερκοιλιακή-ταχυκαρδία/
25. Andreas C. Müller και Sarah Guido,(2020) Introduction to Machine Learning with Python.
26. Yadav, P. (2018, November 14). Decision Tree in Machine Learning. https://towardsdatascience.com/decision-tree-in-machine-learning-e380942a4c96
27. Chen, T., & Guestrin, C. (2016). XGBoost: A Scalable Tree Boosting System. In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (pp. 785–794). New York, NY, USA: ACM. https://doi.org/10.1145/2939672.2939785
28. Ke, G., Meng, Q., Finley, T., Wang, T., Chen, W., Ma, W., … Liu, T.-Y. (2017). Lightgbm: A highly efficient gradient boosting decision tree. Advances in Neural Information Processing Systems, 30, 3146–3154.
29. Kluyver, T., Ragan-Kelley, B., Fernando P'erez, Granger, B., Bussonnier, M., Frederic, J., … Willing, C. (2016). Jupyter Notebooks – a publishing format for reproducible computational workflows. In F. Loizides & B. Schmidt (Eds.), Positioning and Power in Academic Publishing: Players, Agents and Agendas (pp. 87–90).
30. Harris, C. R., Millman, K. J., van der Walt, S. J., Gommers, R., Virtanen, P., Cournapeau, D., … Oliphant, T. E. (2020). Array programming with NumPy. Nature, 585, 357–362. https://doi.org/10.1038/s41586-020-2649-2
31. McKinney, W., & others. (2010). Data structures for statistical computing in python. In Proceedings of the 9th Python in Science Conference (Vol. 445, pp. 51–56).

32. J. D. Hunter, "Matplotlib: A 2D Graphics Environment", Computing in Science & Engineering, vol. 9, no. 3, pp. 90-95, 2007.
33. Jianwei Zheng et al. 2020 A 12-lead electrocardiogram database for arrhythmia research covering more than 10,000 patients
34. Ling, Charles X., and Chenghui Li. "Data mining for direct marketing: Problems and solutions." Kdd. Vol. 98. 1998.
35. Devanshi. Types of Machine Learning Algorithms. Programsbuzz. https://www.programsbuzz.com/article/types-machine-learning-algorithms
36. Supervised Machine Learning. Avatpoint. https://www.javatpoint.com/supervised-machine-learning
37. Unsupervised Machine Learning. Avatpoint. https://www.javatpoint.com/unsupervised-machine-learning
38. Ensemble Methods. Corporatefinanceinstitute. https://corporatefinanceinstitute.com/resources/data-science/ensemble-methods/
39. Pickles, J. C., Stone, T. J., & Jacques, T. S. (2020). Methylation-based algorithms for diagnosis: experience from neuro-oncology. The Journal of Pathology, *250*(5), 510-517.
40. Ashik, K. (2020, March 2). *XGBoost Vs LightGBM*. Linkedin. https://www.linkedin.com/pulse/xgboost-vs-lightgbm-ashik-kumar/
41. 10- fold validation Talpur, A. (2017). *Congestion Detection in Software Defined Networks using Machine Learning* (Doctoral dissertation, PhD thesis, 02 2017).