Περιεχόμενα

[I. Εισαγωγή 3](#_Toc143361273)

[II. Βιβλιογραφική ανασκόπηση 4](#_Toc143361274)

[A. Αναφορά σχετικών ερευνών. 4](#_Toc143361275)

[B. Θεωρητικό πλαίσιο εργασίας. 5](#_Toc143361276)

[1) Αλγόριθμοι. 5](#_Toc143361277)

[2) Μετρικές και πίνακες. 10](#_Toc143361278)

[III. Μέθοδοι 14](#_Toc143361279)

[A. Χρήσιμα πακέτα. 14](#_Toc143361280)

[B. Κυρίως κώδικας. 15](#_Toc143361281)

[C. Παραγόμενα αποτελέσματα. 20](#_Toc143361282)

[D. Υπολογιστικό σύστημα. 21](#_Toc143361283)

[IV. Πειραματικά αποτελέσματα 21](#_Toc143361284)

[A. Ανάλυση ραβδογραμμάτων. 22](#_Toc143361285)

[B. Ανάλυση γραμμικών γραφημάτων. 24](#_Toc143361286)

[C. Ανάλυση μετρικών βέλτιστων μοντέλων. 25](#_Toc143361287)

[1) Ανάλυση για το σύνολο δεδομένων ‘breast cancer’. 25](#_Toc143361288)

[2) Ανάλυση για το σύνολο δεδομένων ‘Pima diabetes’. 26](#_Toc143361289)

[3) Ανάλυση για το σύνολο δεδομένων ‘wine color’. 28](#_Toc143361290)

[4) Σύνοψη. 29](#_Toc143361291)

[V. Συμπεράσματα 29](#_Toc143361292)

[VI. Προτάσεις για μελλοντική έρευνα 30](#_Toc143361293)

[VII. Βιβλιογραφία 31](#_Toc143361294)

[VIII. Παράρτημα 32](#_Toc143361295)

[**Εικόνα 1**.Συμψηφισμός των επιμέρους δένδρων στον RandomForestClassifier (Sinha, et al., 2020). 11](#_Toc143361367)

[**Εικόνα 2**.Δομή πίνακα σύγχυσης. 16](#_Toc143361368)

[**Εικόνα 3.** Διάγραμμα Venn του συνόλου προβλέψεων και των πραγματικών τιμών των δεδομένων. 17](file:///C:\Users\-%20Κωνσταντίνος%20-\OneDrive\Υπολογιστής\BAKAS(2).docx#_Toc143361369)

[**Εικόνα 4.** Πακέτα που χρησιμοποιήθηκαν για την υλοποίηση του κώδικα. 18](file:///C:\Users\-%20Κωνσταντίνος%20-\OneDrive\Υπολογιστής\BAKAS(2).docx#_Toc143361370)

[**Εικόνα 5.** Διάγραμμα ροής του κώδικα που παράχθηκε. 19](file:///C:\Users\-%20Κωνσταντίνος%20-\OneDrive\Υπολογιστής\BAKAS(2).docx#_Toc143361371)

[**Εικόνα 6.** Ορισμός συνάρτησης για την δημιουργία πινάκων σύγχυσης. 20](file:///C:\Users\-%20Κωνσταντίνος%20-\OneDrive\Υπολογιστής\BAKAS(2).docx#_Toc143361372)

[**Εικόνα 7.** Ορισμός χώρων αναζήτησης υπερπαραμέτρων. 20](file:///C:\Users\-%20Κωνσταντίνος%20-\OneDrive\Υπολογιστής\BAKAS(2).docx#_Toc143361373)

[**Εικόνα 8.** Ορισμός της λίστας μοντέλων. 21](file:///C:\Users\-%20Κωνσταντίνος%20-\OneDrive\Υπολογιστής\BAKAS(2).docx#_Toc143361374)

[**Εικόνα 9**. Ορισμός ποσοστού δεδομένων ελέγχου και λίστας συνόλων δεδομένων. 21](file:///C:\Users\-%20Κωνσταντίνος%20-\OneDrive\Υπολογιστής\BAKAS(2).docx#_Toc143361375)

[**Εικόνα 10**. Δημιουργία φακέλων και αρχείων. 21](file:///C:\Users\-%20Κωνσταντίνος%20-\OneDrive\Υπολογιστής\BAKAS(2).docx#_Toc143361376)

[**Εικόνα 11.** Φωλιασμένοι βρόχοι επανάληψης. 22](file:///C:\Users\-%20Κωνσταντίνος%20-\OneDrive\Υπολογιστής\BAKAS(2).docx#_Toc143361377)

[**Εικόνα 12**. Βήματα εξαγωγής πειραματικών αποτελεσμάτων. 23](file:///C:\Users\-%20Κωνσταντίνος%20-\OneDrive\Υπολογιστής\BAKAS(2).docx#_Toc143361378)

[**Εικόνα 13**. Ανανέωση αρχείων. 23](file:///C:\Users\-%20Κωνσταντίνος%20-\OneDrive\Υπολογιστής\BAKAS(2).docx#_Toc143361379)

[**Εικόνα 14.** Δημιουργία μέσων οπτικοποίησης. 24](file:///C:\Users\-%20Κωνσταντίνος%20-\OneDrive\Υπολογιστής\BAKAS(2).docx#_Toc143361380)

[**Εικόνα 15.** Φάκελοι πειραματικών αποτελεσμάτων. 25](file:///C:\Users\-%20Κωνσταντίνος%20-\OneDrive\Υπολογιστής\BAKAS(2).docx#_Toc143361381)

[**Εικόνα 16**. Όλοι οι υπερπαράμετροι των 27 βέλτιστων μοντέλων που εκπαιδεύτηκαν. 26](file:///C:\Users\-%20Κωνσταντίνος%20-\OneDrive\Υπολογιστής\BAKAS(2).docx#_Toc143361382)

[**Εικόνα 17**. Ραβδογράμματα υπολογιστικού χρόνου. 27](file:///C:\Users\-%20Κωνσταντίνος%20-\OneDrive\Υπολογιστής\BAKAS(2).docx#_Toc143361383)

[**Εικόνα 18.** Γραμμικά γραφήματα. 28](file:///C:\Users\-%20Κωνσταντίνος%20-\OneDrive\Υπολογιστής\BAKAS(2).docx#_Toc143361384)

[**Εικόνα 19**.Πίνακες σύγχυσης για το σύνολο δεδομένων ‘breast cancer’. 30](file:///C:\Users\-%20Κωνσταντίνος%20-\OneDrive\Υπολογιστής\BAKAS(2).docx#_Toc143361385)

[**Εικόνα 20.** Πίνακες σύγχυσης για το σύνολο δεδομένων ‘Pima diabetes’. 31](file:///C:\Users\-%20Κωνσταντίνος%20-\OneDrive\Υπολογιστής\BAKAS(2).docx#_Toc143361386)

[**Εικόνα 21.** Πίνακες σύγχυσης για το σύνολο δεδομένων ‘wine color’. 32](file:///C:\Users\-%20Κωνσταντίνος%20-\OneDrive\Υπολογιστής\BAKAS(2).docx#_Toc143361387)

[**Πίνακας 1.** Πίνακας μετρικών για τα μοντέλα gradient boosting. 37](file:///C:\Users\-%20Κωνσταντίνος%20-\OneDrive\Υπολογιστής\BAKAS(2).docx#_Toc143361636)

[**Πίνακας 2.** Πίνακας μετρικών για τα μοντέλα τυχαίων δασών. 38](file:///C:\Users\-%20Κωνσταντίνος%20-\OneDrive\Υπολογιστής\BAKAS(2).docx#_Toc143361637)

[**Πίνακας 3.** Πίνακας μετρικών για τα μοντέλα XGBoost. 39](file:///C:\Users\-%20Κωνσταντίνος%20-\OneDrive\Υπολογιστής\BAKAS(2).docx#_Toc143361638)

ΣΎΓΚΡΙΣΗ ΑΛΓΟΡΊΘΜΩΝ ΕΎΡΕΣΗΣ ΒΈΛΤΙΣΤΩΝ ΥΠΕΡΠΑΡΑΜΈΤΡΩΝ ΜΟΝΤΈΛΩΝ ΜΗΧΑΝΙΚΉΣ ΜΆΘΗΣΗΣ

ΜΙΑ ΈΡΕΥΝΑ ΤΗΣ ΑΠΟΔΟΤΙΚΌΤΗΤΑΣ ΑΛΓΟΡΊΘΜΩΝ ΒΕΛΤΙΣΤΟΠΟΊΗΣΗΣ ΥΠΕΡΠΑΡΑΜΈΤΡΩΝ

ΚΑΡΑΤΕΚΊΔΗΣ ΚΩΝΣΤΑΝΤΊΝΟΣ

ΕΠΙΒΛΈΠΩΝ ΚΑΘΗΓΗΤΉΣ ΝΙΚΌΛΑΟΣ ΜΠΆΚΑΣ

ΜΗΤΡΟΠΟΛΙΤΙΚΟ ΚΟΛΛΕΓΙΟ

ΘΕΣΣΑΛΟΝΙΚΗ, ΕΛΛΑΔΑ

kostaskaratekidis@gmail.com

*Περίληψη*— Οι αλγόριθμοι μηχανικής μάθησης συχνά απαιτούν των κατάλληλο ορισμό πολλών υπερπαραμέτρων, οι οποίες επηρεάζουν τις διαδικασίες λειτουργίας των μοντέλων με πολύπλοκους τρόπους. Λόγω του μεγάλου πλήθους πιθανών συνδυασμών, είναι συνηθισμένο να χρησιμοποιούνται μέθοδοι για την αυτοματοποιημένη εύρεση των κατάλληλων υπερπαραμέτρων που θα βελτιστοποιήσουν την απόδοση των μοντέλων μηχανικής μάθησης. Η παρούσα εργασία διερευνά την αποτελεσματικότητα τριών σημαντικών μεθόδων συντονισμού υπερπαραμέτρων που υλοποιούνται από τους RandomizedSearchCV, TPOTClassifier και BayesSearchCV, που εφαρμόζονται για την εύρεση βέλτιστων υπερπαραμέτρων πάνω σε τρεις αλγόριθμους μηχανικής μάθησης που έχουν ως βάση τα δένδρα αποφάσεων και υλοποιούνται από τους XGBoostClassifier, RandomForestClassifier και GradientBoostingClassifier. Πρωταρχικός στόχος της εργασίας αυτής είναι η σύγκριση της απόδοσης αυτών των μεθόδων σε διάφορα σύνολα δεδομένων, παρέχοντας πληροφορίες σχετικά με την αποτελεσματικότητά τους και τον αντίκτυπό τους στην ακρίβεια των μοντέλων που παράγουν. Για την επίτευξη αυτού του στόχου παράχθηκε κατάλληλος κώδικας στην γλώσσα προγραμματισμού Python για την εκτέλεση πειραμάτων του οποίου η λειτουργία και δομή παρουσιάζονται διεξοδικά. Για όλους τους αλγόριθμους ,τις μετρικές που χρησιμοποιούνται και τα σύνολα δεδομένων παρέχεται μία εξήγηση. Η ανάλυση περιλαμβάνει τη διεξαγωγή πολλαπλών πειραμάτων για κάθε συνδυασμό αλγορίθμου, μεθόδου συντονισμού υπερπαραμέτρων και δεδομένων και την σύγκριση οπτικών και αριθμητικών δεδομένων που προκύπτουν κατά την διαδικασία εκτέλεσης του πειράματος. Λαμβάνοντας υπόψιν την ταχύτητα εκτέλεσης, την μεθοδικότητα κάθε διαδικασίας αναζήτησης και την σύγκριση των βέλτιστων μοντέλων που βρέθηκαν για κάθε μέθοδο αναζήτησης βέλτιστων υπερπαραμέτρων καταλήξαμε στο συμπέρασμα ότι οι πιο αποδοτικοί αλγόριθμοι στα πλαίσια της εργασίας φάνηκαν να είναι οι TPOTClassifier και BayesSearchCV, με τον BayesSearchCV να είναι αργός στην εκτέλεσή του μα ο πιο αποδοτικός στην εύρεση των βέλτιστων μοντέλων και ο TPOTClassifier να είναι και αποδοτικός και σαφώς πιο γρήγορος από τον BayesSearchCV ενώ ο RandomizedSearchCV βρέθηκε να μην είναι τόσο συνεπής στην εύρεση των βέλτιστων υπερπαραμέτρων αν και ήταν ο γρηγορότερος στην εκτέλεσή του.

Abstract— Machine learning algorithms often require the appropriate specification of multiple hyperparameters, which intricately affect the operational processes of the models in complex ways. Due to the large number of possible combinations, it is common to employ methods for the automated discovery of suitable hyperparameters that will optimize the performance of machine learning models. This study investigates the effectiveness of three significant hyperparameter tuning methods, namely RandomizedSearchCV, TPOTClassifier, and BayesSearchCV. These methods are implemented to find optimal hyperparameters for three tree-based machine learning algorithms that are implemented by the XGBoostClassifier, RandomForestClassifier, and GradientBoostingClassifier functions. The primary objective of this study is to compare the performance of these methods across various datasets, offering insights into their efficiency and impact on the accuracy of the generated models. To achieve this goal, appropriate Python code was developed to execute experiments. The code’s function and structure are explained in detail. A comprehensive analysis involves conducting multiple experiments for each algorithm-hyperparameter tuning method combination and dataset. Visual and numerical data resulting from the experimentation process are compared. Taking into consideration execution speed, methodological robustness, and the comparison of optimal models found by each search method, we conclude that the most efficient algorithms within the scope of this study are TPOTClassifier and BayesSearchCV. BayesSearchCV, although slow in execution, is the most efficient in finding optimal models, while TPOTClassifier is both efficient and significantly faster than BayesSearchCV. On the other hand, RandomizedSearchCV is found to be less consistent in finding optimal hyperparameters, despite being the fastest in execution.

Keywords—machine learning, hyperparameter tuning, algorithms, comparison

# Εισαγωγή

Οι αλγόριθμοι μηχανικής μάθησης είναι σύνηθες να περιέχουν πολλές υπερπαραμέτρους των οποίων οι τιμές επηρεάζουν την απόδοση των μοντέλων που περιγράφουν με περίπλοκους τρόπους. Λόγω του μεγάλου αριθμού συνδυασμών που μπορεί να προκύψουν από όλες αυτές τις υπερπαραμέτρους είναι συχνή η πρακτική εφαρμογής αλγορίθμων για την εύρεση των κατάλληλων υπερπαραμέτρων που θα εξασφαλίσουν την καλύτερη απόδοση του αλγορίθμου μηχανικής μάθησης. Ωστόσο, δεν είναι ξεκάθαρο το πώς θα μπορούσε να εξερευνηθεί αποτελεσματικά αυτός ο τεράστιος χώρος. Είναι αναπόφευκτο να δημιουργηθούν απορίες όπως, ποιοι είναι οι καλύτεροι αλγόριθμοι βελτιστοποίησης για την αντιμετώπιση ενός συγκεκριμένου προβλήματος, πότε και πώς πρέπει να τους χρησιμοποιούμε για να έχουμε τα καλύτερα δυνατά αποτελέσματα και ακόμη είναι σημαντικοί για την καλύτερη απόδοση των μοντέλων μηχανικής μάθησης τόσο ως προς την ακρίβεια των προβλέψεων όσο και στον χρόνο εκτέλεσής τους; Σε αυτή την εργασία παρέχετε μια προσέγγιση για τη διερεύνηση του συντονισμού υπερπαραμέτρων από τρείς διαφορετικούς αλγορίθμους συντονισμού υπερπαραμέτρων πάνω σε τρεις αλγορίθμους μηχανικής μάθησης που θα εφαρμοστούν σε τρία διαφορετικά αναφορικά σύνολα δεδομένων για την κατηγοριοποίηση τους.

Οι αλγόριθμοι συντονισμού που επιλέχτηκαν είναι οι RandomizedSearchCV, BayesSearchCV και TPOTClassifier που υλοποιούν αλγόριθμους τυχαίας αναζήτησης, μπεϋζιανό και γενετικό αλγόριθμο συντονισμού υπερπαραμέτρων αντίστοιχα. Αυτοί οι αλγόριθμοι επιλέχτηκαν ως αντιπροσωπευτικοί των κατηγοριών αλγορίθμων αναζήτησης πλέγματος, μπευζιανής αναζήτησης και εξελικτικών αλγόριθμων. Επίσης επιλέχτηκαν από την βιβλιοθήκη "XGBoost" και "scikit-learn" οι αλγόριθμοι μηχανικής μάθησης XGBoostClassifier, RandomForestClassifier και GradientBoostingClassifier ως αλγόριθμοι με καλές επιδόσεις σε προβλήματα κατηγοριοποίησης (Mantovani, et al., 2019) βασισμένοι σε δένδρα αποφάσεων και τεχνικές υλοποίησης ενίσχυσης κλίσης που διαθέτουν πλήθος υπερπαραμέτρων πάνω στο οποίο μπορούν να εφαρμοστούν αλγόριθμοι εύρεσης βέλτιστων υπερπαραμέτρων. Τα σύνολα δεδομένων που επιλέχτηκαν είναι το “Pima Indians Diabetes Database” (LEARNING, 2016) που είναι ένα σύνολο δεδομένων που προέρχεται από το Εθνικό Ινστιτούτο Διαβήτη και Πεπτικών και Νεφρικών ασθενειών στο οποίο στόχος είναι να προβλεφθεί, με βάση συγκεκριμένες διαγνωστικές μετρήσεις, εάν ένας ασθενής έχει διαβήτη ή όχι.. Ακόμη επιλέχτηκε το “Breast Cancer Wisconsin Diagnostic Data Set” (LEARNING, 2016) όπου βάση ορισμένων χαρακτηριστικών από δείγματα όγκων μαστών σκοπός είναι να προβλεφθεί αν ο όγκος είναι καλοήθης ή κακοήθης. Τέλος, επιλέχτηκε το “Wine Quality Data Set (Red & White Wine)” (RUTHGN, 2021) στο οποίο επιλέχτηκε να γίνει πρόβλεψη ως προς το κόκκινο ή λευκό χρώμα του κρασιού βάση όλων των άλλων χαρακτηριστικών του. Με αυτά τα σύνολα δεδομένων πραγματοποιήθηκαν πειράματα συντονισμού των υπερπαραμέτρων των τριών μοντέλων μηχανικής μάθησης χρησιμοποιώντας τους τρείς αλγορίθμους εύρεσης παραμέτρων, έτσι γίνεται λόγος για την μελέτη 27 διαφορετικών πειραματικών αποτελεσμάτων με σκοπό την σύγκριση απόδοσης των αλγορίθμων.

Πολλοί αλγόριθμοι Μηχανικής Μάθησης (ML) είναι ικανοί να αντιμετωπίσουν εργασίες ταξινόμησης όπως παρουσιάζεται στην βιβλιογραφία. Σε αυτά τα μοντέλα η ακρίβεια είναι το πιο συχνά χρησιμοποιούμενο μέτρο για την αξιολόγηση τους αλλά σε πολλές εφαρμογές, η εύκολη ερμηνεία των μοντέλων είναι επίσης σημαντική απαίτηση. Η καλή απόδοση σε ακρίβεια και η εύκολη ερμηνεία του μοντέλου είναι κοινό χαρακτηριστικό των αλγορίθμων μηχανικής μάθησης που έχουν ως βασική τους αρχή τα δέντρα αποφάσεων (Mantovani, et al., 2019). Το μοντέλο Random Forest (RF) (Pal, 2005) είναι μια μέθοδος εκμάθησης συνόλου που συνδυάζει πολλαπλά δέντρα αποφάσεων για να κάνει προβλέψεις. Σε κάθε δέντρο του δάσους γίνεται εκπαίδευση σε ένα τυχαία επιλεγμένο υποσύνολο των δεδομένων εκπαίδευσης και χρησιμοποιείται ένα τυχαίο υποσύνολο χαρακτηριστικών. Η τελική πρόβλεψη προκύπτει από τη συγκέντρωση των προβλέψεων όλων των δέντρων. Το μοντέλο XGBoost (XGB) είναι ένα πλαίσιο ενίσχυσης κλίσης που κάνει χρήση ενός συνόλου αδύναμων μοντέλων δέντρων αποφάσεων προσθέτοντας διαδοχικά νέα δέντρα για να διορθώσει τα λάθη που έγιναν από τα προηγούμενα δέντρα (Chen & Guestrin, 2016). Το XGBoost χρησιμοποιεί τεχνικές ενίσχυσης κλίσης για να ελαχιστοποιήσει μια καθορισμένη συνάρτηση απώλειας. Το μοντέλο Gradient Boosting Classifier (GBC) είναι μια άλλη εφαρμογή του αλγόριθμου ενίσχυσης κλίσης, παρόμοια με το XGB η οποία δημιουργεί κι αυτή ένα σύνολο δέντρων αποφάσεων διαδοχικά, ώστε κάθε καινούριο δέντρο να εκπαιδεύεται ώστε να κάνει διορθώσεις στα λάθη των προηγούμενων δέντρων. Το GBC χρησιμοποιεί βελτιστοποίηση ενίσχυσης κλίσης για να βρει τις καλύτερες παραμέτρους για τα δέντρα απόφασης ( Chakrabarty, et al., 2018). Συνοπτικά, τα RF, XGB και GBC χρησιμοποιούν όλα τα δέντρα απόφασης ως βασικά τους μοντέλα, αλλά διαφέρουν στον τρόπο κατασκευής και συνδυασμού των δέντρων για να κάνουν προβλέψεις. Αυτές οι διαφορές και ομοιότητες στην λειτουργία τους τα καθιστούν άξια χρήσης τους για την σύγκρισή τους αλλά και για την σύγκριση των αλγορίθμων εύρεσης παραμέτρων.

Ο συντονισμός υπερπαραμέτρων είναι ζωτικής σημασίας επειδή η επιλογή κατάλληλων τιμών υπερπαραμέτρων μπορεί να επηρεάσει σημαντικά την απόδοση ενός μοντέλου μηχανικής μάθησης. Διαφορετικοί αλγόριθμοι βελτιστοποίησης υπερπαραμέτρων είναι δυνατό να δώσουν διαφορετικά αποτελέσματα ανάλογα με την μορφή του πλέγματος υπερπαραμέτρων στο οποίο γίνεται η αναζήτηση και του τρόπου με τον οποίο γίνεται η αναζήτηση από τον κάθε αλγόριθμο. Το TPOTClassifier είναι μία συνάρτηση που βασίζεται σε γενετικό προγραμματισμό. Η προσέγγιση του γενετικού προγραμματισμού είναι εμπνευσμένη από τα δαρβινικά εξελικτικά μοντέλα και μπορεί να είναι πολύτιμη για τον εντοπισμό του πιο αποτελεσματικού συνδυασμού υποσυνόλων χαρακτηριστικών για τη δημιουργία μοντέλων κατηγοριοποίησης υψηλής απόδοσης (Abu Syeed , et al., 2021). Η RandomizedSearchCV είναι μία συνάρτηση η οποία ορίζοντας έναν καθορισμένο αριθμό τυχαίων αναζητήσεων δοκιμάζει τυχαίες δυνατότητες εντός ενός συγκεκριμένου εύρους τιμών για κάθε υπερπαράμετρο με αποτέλεσμα συγκρίνοντας τα εκπαιδευμένα μοντέλα βάση μίας μετρικής απόδοσης να βρίσκει μία πρόταση βέλτιστων υπερπαραμέτρων πολύ πιο γρήγορα από την συνάρτηση GridsearchCV. Αυτή, μπορεί να φανεί χρήσιμη για την εξαγωγή γρήγορων αποτελεσμάτων όμως όχι απαραίτητα και των αποδοτικότερων λόγο της τυχαιότητας με την οποία λειτουργεί (Yang & Shami, 2020) (Datta & Faroughi, 2022). Στην BayesSearchCV συνάρτηση η βασική φιλοσοφία είναι να χρησιμοποιηθούν όλες οι διαθέσιμες πληροφορίες από προηγούμενα μοντέλα που αξιολογήθηκαν ώστε να βρεθούν διαδοχικά μοντέλα που διαρκώς θα δίνουν μικρότερες τιμές μέχρι να βρεθούν οι υπερπαράμετροι που δίνουν την μικρότερη, στην περίπτωση αυτή, αρνητική τιμή της ακρίβειας (Snoek, et al., 2012). Γίνεται λοιπόν φανερή η διαφορετικότητα των τριών αλγορίθμων.

Συγκρίνοντας διάφορους αλγόριθμους για βελτιστοποίηση υπερπαραμέτρων σε διαφορετικά μοντέλα μηχανικής μάθησης και σε διάφορα σύνολα δεδομένων μπορούμε να συμβάλλουμε στην εμπειρική γνώση των τεχνικών μηχανικής μάθησης με τους ακόλουθους τρόπους:

* Με τη δοκιμή των αλγορίθμων σε διαφορετικά μοντέλα μηχανικής μάθησης και σύνολα δεδομένων, θα αξιολογηθεί εάν ένας αλγόριθμος έχει σταθερά καλή απόδοση σε διάφορους τομείς προβλημάτων ή εάν η αποτελεσματικότητά του ποικίλλει ανάλογα με τα ειδικά χαρακτηριστικά των δεδομένων ή του μοντέλου. Η προσπάθεια γενίκευσης των αποτελεσμάτων των ποικίλων αλγορίθμων είναι υψηλής σημασίας για τη δημιουργία βέλτιστων πρακτικών και προτάσεων στον τομέα της μηχανικής μάθησης.
* Εκτός από την επίδοση, μπορεί να αξιολογηθεί και η υπολογιστική απαίτηση κάθε αλγορίθμου. Αυτό περιλαμβάνει την αξιολόγηση του χρόνου που απαιτείται για την εύρεση των βέλτιστων υπερπαραμέτρων με κάθε αλγόριθμο για διαφορετικά μοντέλα και σύνολα δεδομένων. Τέτοιες πληροφορίες μπορούν να βοηθήσουν στη λήψη αποφάσεων, ειδικά σε εργασίες με σύνολα δεδομένων μεγάλης κλίμακας ή εφαρμογές ευαίσθητες στον υπολογιστικό χρόνο, όπου η υπολογιστική ισχύς είναι καθοριστική.

Συνεπώς η συγκεκριμένη εργασία παρουσιάζει ιδιαίτερο ερευνητικό ενδιαφέρον και είναι φανερό πως απασχολεί την σύγχρονη επιστημονική και επαγγελματική κοινότητα του χώρου.

Η εργασία διατηρεί μία σειρά παρουσίασης του θέματος. Πρώτα, θα παρουσιαστούν τα ευρήματα της βιβλιογραφικής ανασκόπησης που έγινε για την εύρεση του θέματος ,δηλαδή θα γίνει αναφορά σε σχετικές έρευνες, μαζί με μία πιο θεωρητική και αναλυτική περιγραφή των αλγορίθμων που θα χρησιμοποιηθούν. Στην συνέχεια θα γίνει λόγος για τις μεθόδους που χρησιμοποιήθηκαν κατά το πειραματικό στάδιο και για πιο λόγο επιλέχτηκαν. Έπειτα, θα γίνει περιγραφή της πειραματικής διαδικασίας και των αποτελεσμάτων. Τέλος, θα γίνει η ερμηνεία των πειραματικών αποτελεσμάτων και θα δοθούν τα συμπεράσματα.

# Βιβλιογραφική ανασκόπηση

Πραγματοποιήθηκε μια ευρεία αναζήτηση σε ακαδημαϊκές βάσεις δεδομένων τις οποίας τα συμπεράσματα παρουσιάζονται στην παράγραφο II.A. Επίσης, μέσω αυτής της έρευνας δημιουργήθηκε και ένα θεωρητικό πλαίσιο πάνω στο οποίο στηρίχτηκε η εργασία το οποίο παρουσιάζεται στην παράγραφο II.B.

## Αναφορά σχετικών ερευνών.

Όπως αναφέρθηκε, το πρόβλημα εύρεσης των βέλτιστων παραμέτρων πολύπλοκων μοντέλων μηχανικής μάθησης απασχολεί ιδιαίτερα την επιστημονική κοινότητα με αποτέλεσμα να υπάρχει πλήθος ερευνών πάνω στην σύγκριση μεθόδων συντονισμού των υπερπαραμέτρων μοντέλων μηχανικής μάθησης αλλά και εφαρμογή τέτοιων αλγορίθμων για την επίλυση συγκεκριμένων προβλημάτων που βασίζονται στην υψηλής ακρίβειας πρόβλεψη ενός μοντέλου μηχανικής μάθησης. Η ανάγκη για να γίνουν οι αλγόριθμοι μηχανικής μάθησης ανεξάρτητοι με την έννοια του αυτόματου συντονισμού των υπερπαραμέτρων τους γίνεται ιδιαίτερα φανερή στην επιστημονική κοινότητα καθώς κρίνονται σημαντικές οι στρατηγικές για την αυτοματοποιημένη βελτιστοποίηση των μοντέλων καθώς και ανάπτυξη μοντέλων με λιγότερες υπερπαραμέτρους όπως αναφέρετε στην βιβλιογραφία (Hutter, et al., 2015). Άλλα παραδείγματα ερευνών που δείχνουν την σημασία των αλγορίθμων εύρεσης βέλτιστων υπερπαραμέτρων είναι στο πεδίο της εξερεύνησης για πόρους πετρελαίου όπου προσδιορίζεται με ακρίβεια η λιθολογία μίας περιοχής κάνοντας χρήση διάφορων μοντέλων μηχανικής μάθησης στα οποία για την εύρεση των παραμέτρων τους χρησιμοποιήθηκαν αλγόριθμοι συντονισμού υπερπαραμέτρων για την βελτίωση της ακρίβειας τους (Xie, et al., 2018). Ακόμη, παρόμοιο πρόβλημα παρουσιάζεται στον τομέα του «computer vision» οι σχετικοί αλγόριθμοι διαθέτουν πλήθος υπερπαραμέτρων που πρέπει να συντονιστούν για να υπάρξει η βέλτιστη απόδοση, μάλιστα γίνεται λόγος για την δυσκολία των μη αυτόματων μεθόδων που χρησιμοποιούνται για να γίνει αυτό καθώς και για την δυσκολία σύγκρισης της καταλληλόλητας δύο μοντέλων καθώς είναι μερικές φορές δύσκολο να εντοπιστεί αν ένα μοντέλο είναι πραγματικά καλύτερο, ή απλά έχει καλύτερα συντονισμένες τις υπερπαραμέτρους που το διέπουν (Bergstra, et al., 2013). Ένας εύλογος χώρος για την χρήση αλγορίθμων εύρεσης βέλτιστων υπερπαραμέτρων είναι και τα βαθιά νευρωνικά δίκτυα. Εκεί, παρά τις σημαντικές προόδους που έχουν σημειωθεί στον τομέα, η διαδικασία σχεδιασμού και εκπαίδευσης νευρωνικών δικτύων παραμένει δύσκολη και συχνά αναφέρεται ως αλχημεία, λόγο της μυστηριώδους και αβέβαιης φύση της ( Yu & Zhu, 2020) ενώ δεν είναι λίγες οι φορές που γίνεται χρήση αλγορίθμων ωμής βίας για την εύρεση τους, πράγμα χρονοβόρο και ενεργοβόρο, κάτι που μπορεί να αποφευχθεί με την χρήση πιο αποδοτικών και λιγότερο δαπανηρών αλγορίθμων (Wu, et al., 2019). Συνεπώς η χρήση αλγορίθμων συντονισμού υπερπαραμέτρων είναι πλέον αναπόσπαστο κομμάτι του ερευνητικού ενδιαφέροντος και φαίνεται να έχει πληθώρα εφαρμογών.

Από τα παραπάνω διαφαίνεται η σημασία του συντονισμού υπερπαραμέτρων και γίνεται και λόγος για την σύγκριση μοντέλων μηχανικής μάθησης κατόπιν συντονισμού των υπερπαραμέτρων τους ώστε να γίνεται μία δίκαιη σύγκριση. Όμως, δεν γίνεται ιδιαίτερος λόγος για το ποιοι αλγόριθμοι εύρεσης υπερπαραμέτρων είναι οι καλύτεροι για την εύρεση των υπερπαραμέτρων ενός μοντέλου μηχανικής μάθησης, δηλαδή για την εύρεση του βέλτιστου μοντέλου που με την εφαρμογή του σε ένα σύνολο δεδομένων θα είχαμε την καλύτερη πρόβλεψη. Αυτό είναι εύλογο καθώς η εύρεση των βέλτιστων υπερπαραμέτρων ενός μοντέλου που εκπαιδεύεται πάνω σε ένα σύνολο δεδομένων εξαρτώνται τόσο από την δομή του μοντέλου όσο και από το είδος των δεδομένων καθώς και από το κατά πόσο η επιλογή του συγκεκριμένου μοντέλου είναι η κατάλληλη για το συγκεκριμένο σύνολο δεδομένων. Είναι, λοιπόν μία χαοτική και σε μεγάλο μέρος αχαρτογράφητη περιοχή της μηχανικής μάθησης.

Παρόλα αυτά, έχουν υπάρξει μελέτες για την σύγκριση τέτοιων αλγορίθμων για ορισμένα μοντέλα και σύνολα δεδομένων. Ένα τέτοιο παράδειγμα έρευνας συγκρίνει διαφορετικούς αλγόριθμους συντονισμού παραμέτρων που εφαρμόστηκαν πάνω σε αλγόριθμους δένδρων αποφάσεων που εκπαιδεύτηκαν σε πληθώρα συνόλων δεδομένων κατηγοριοποίησης (Mantovani, et al., 2019). Επίσης γίνονται προσπάθειες για την παρουσίαση αλγορίθμων που είναι πιο αποδοτική στην εύρεση υπερπαραμέτρων όπως στο άρθρο ( Bergstra, et al., 2011) που παρουσιάζει δύο αλγορίθμους βελτιστοποίησης υπερπαραμέτρων και κάνει φανερό ότι επιτυγχάνουν ή υπερβαίνουν την ανθρώπινη απόδοση και την απόδοση μιας τυχαίας αναζήτησης «Random Search» σε δύο δύσκολες εργασίες βελτιστοποίησης υπερπαραμέτρων που αφορούν «Deep Belief Networks» που είναι μοντέλα με μεγάλο πλήθος υπερπαραμέτρων. Ακόμη, επιλέγοντας ένα μοντέλο μηχανικής μάθησης εξετάζεται ποιος αλγόριθμος συντονισμού υπερπαραμέτρων μπορεί να δώσει τις πιο αποδοτικές για τις προβλέψεις υπερπαραμέτρους. Έτσι για παράδειγμα στην βιβλιογραφία εξετάζεται ποιος είναι ο βέλτιστος, μεταξύ ενός συνόλου αλγορίθμων, αλγόριθμος συντονισμού των υπερπαραμέτρων του μοντέλου τυχαίων δασών πάνω σε πληθώρα συνόλων δεδομένων που σκοπό έχει να καταδείξει, αν υπάρχει, μία συνέπεια στην συμπεριφορά των αλγορίθμων για κάθε σύνολο δεδομένων ( Probst, et al., 2019). Παρόμοια, γίνεται σύγκριση τεσσάρων διαφορετικών αλγορίθμων συντονισμού υπερπαραμέτρων για την προσαρμογή των υπερπαραμέτρων του J48 που είναι ένας αλγόριθμος δένδρου αποφάσεων (Mantovani, et al., 2016). Συνολικά, για τον J48 έγινε πειραματική ανάλυση χρησιμοποιώντας 102 ετερογενή σύνολα δεδομένων για την εκπαίδευση των επαγόμενων, από την διαδικασία συντονισμού υπερπαραμέτρων, μοντέλων (Mantovani, et al., 2016). Αναφορικά, μπορεί να γίνει λόγος και σε άλλες παρόμοιες έρευνες όπου είτε επιλέγεται ένα μοντέλο μηχανικής μάθησης και ένα σύνολο δεδομένων πάνω στα οποία στηρίζεται η σύγκριση ενός συνόλου αλγορίθμων εύρεσης υπερπαραμέτρων (Panichella, 2021) (Alibrahim & Ludwig, 2021) είτε επιλέγονται διαφορετικά μοντέλα και ένα σύνολο δεδομένων ( Zahedi, et al., 2021) ( Bacanin, et al., 2023) ή γίνεται επιλογή πολλών συνόλων δεδομένων και διαφορετικών μοντέλων (Turner, et al., 2021) (Franceschi, et al., 2017) ανάλογα με τον σκοπό της σύγκρισης κάθε φορά. Σε κάθε περίπτωση γίνεται μία προσπάθεια να εξασφαλισθεί μία δίκαιη και συνεπή σύγκριση των αλγορίθμων λαμβάνοντας υπόψιν τα μοντέλα και τα σύνολα δεδομένων πάνω στα οποία εφαρμόζονται.

Από τα παραπάνω γίνεται φανερό πως δεν υπάρχει ολοκληρωμένη σύγκριση μεταξύ των αλγορίθμων εύρεσης βέλτιστων υπερπαραμέτρων παρά μόνο για συγκεκριμένα μοντέλα και σύνολα δεδομένων. Επίσης, γίνεται φανερό πως για να εξεταστεί η συνέπεια της αποδοτικότητας ενός τέτοιου αλγορίθμου, αυτός πρέπει να εξεταστεί πάνω σε πλήθος διαφορετικών μοντέλων και συνόλων δεδομένων με διαφορετικά κριτήρια. Ακόμη παρατηρεί κανείς πως οι περισσότερες σχετικές έρευνες επικεντρώνονται στα βαθιά νευρωνικά δίκτυα ενώ υπάρχουν λιγότερες έρευνες για μοντέλα δένδρων αποφάσεων. Ακόμη πιο πολύ μικραίνει το πλήθος των ερευνών αν κάνουμε λόγο για σύνολα δεδομένων που αφορούν προβλήματα κατηγοριοποίησης καθώς και αν απαιτήσουμε να γίνεται σύγκριση αλγορίθμων διαφορετικών από αλγορίθμους ωμής βίας σε διαφορετικά μοντέλα και σύνολα δεομένων. Τέλος, παρατηρείται πως οι βασικότεροι τρόποι σύγκρισης της αποδοτικότητας των αλγορίθμων είναι κάνοντας λόγο για το μέγεθος του χρονικού διαστήματος της διαδικασίας εύρεσης των παραμέτρων για κάποιο μοντέλο, μιλώντας για την ακρίβεια των μοντέλων που εκπαιδεύτηκαν και κατά πόσο αυτές οι συγκρίσεις παραμένουν συνεπής όταν αλλάζουν τα μοντέλα και τα σύνολα δεδομένων. Σε κάθε περίπτωση η δημιουργία ενός δίκαιου πλαισίου για την σύγκριση αλγορίθμων εύρεσης βέλτιστων υπερπαραμέτρων είναι δύσκολο να επιτευχθεί.

Σε αυτή την εργασία, γίνεται μία προσπάθεια σύγκρισης των αλγορίθμων RandomizedSearchCV, BayesSearchCV και TPOTClassifier πάνω σε διαφορετικά μοντέλα κατηγοριοποίησης RF, XGB και GBC και σε διαφορετικά σύνολα δεδομένων που αφορούν προβλήματα κατηγοριοποίησης λαμβάνοντας υπόψιν όλα τα παραπάνω. Η έρευνα αυτή κυρίως διαφοροποιείται από τις άλλες ως προς την επιλογή των αλγορίθμων για τους οποίους θα γίνει σύγκριση καθώς και την επιλογή αυτή η σύγκριση να γίνει σε διαφορετικά μοντέλα κατηγοριοποίησης και σε διαφορετικά σύνολα δεδομένων. Αυτό συμβαίνει διότι θεωρήθηκε πως είναι μία πιο ολοκληρωμένη μέθοδος σύγκρισης των συμπεριφορών των αλγορίθμων εύρεσης βέλτιστων υπερπαραμέτρων καθώς διαφαίνεται η συμπεριφορά τους πάνω σε διαφορετικά μοντέλα μηχανικής μάθησης και σύνολα δεδομένων.

## Θεωρητικό πλαίσιο εργασίας.

Εδώ, γίνεται μία πιο αναλυτική παρουσίαση των αλγορίθμων που θα χρησιμοποιηθούν καθώς και των μετρικών πάνω στις οποίες θα βασιστεί η σύγκριση των αλγορίθμων εύρεσης βέλτιστων υπερπαραμέτρων.

### Αλγόριθμοι.

#### XGBoostClassifier.

Ο αλγόριθμος XGBoost ή Extreme Gradient Boosting ήταν μία προσπάθεια ερευνητικού έργου που έγινε στο πανεπιστήμιο της Ουάσιγκτον από τους Tianqi Chen και Carlos Guestrin. Ο XGBoost είναι ένας αλγόριθμος που υλοποιεί την gradient boosted decision trees μέθοδο και έχει σχεδιαστεί με σκοπό την βελτίωση της ταχύτητας και της απόδοσης ( Afifah, et al., 2021). Το XGBoost έχει την δυνατότητα να κάνει την βέλτιστη δυνατή διαχείριση των πόρων, όπως χρόνου και μνήμης, και να λύνει προβλήματα ανισορροπίας στα δεδομένα. Ο αλγόριθμος XGBoost λειτουργεί ως ένα σύνολο δέντρων απόφασης που βασίζεται σε μεθόδους ενίσχυσης κλίσης ( Afifah, et al., 2021). Η ενίσχυση κλίσης επιτρέπει τη συνδυαστική χρήση της προβλεπτικής ισχύος πολλαπλών μοντέλων μάθησης. Κατά τη διαδικασία της ενίσχυσης, τα δέντρα δημιουργούνται σε ακολουθία, με κάθε επόμενο δέντρο να στοχεύει στη μείωση των σφαλμάτων των προηγούμενων δέντρων ( Afifah, et al., 2021). Τα επόμενα δέντρα διαμορφώνονται και προκύπτουν συναρτήσει των σφαλμάτων των προηγούμενων δέντρων και με σκοπό την ελάττωση αυτών των σφαλμάτων. Με αυτή την διαδικασία τα δέντρα που προκύπτουν στηρίζονται στην μάθηση των προηγούμενων. Οι βασικοί μαθητές στην ενίσχυση είναι αδύναμοι μαθητές με υψηλό «bias» και κάθε ένας από αυτούς τους αδύναμους μαθητές συνεισφέρει δίνοντας πληροφορίες για την τελική πρόβλεψη, επιτρέποντας στην τεχνική της ενίσχυσης να δημιουργήσει ένα ισχυρό μοντέλο συνδυάζοντας αποτελεσματικά αυτούς τους αδύναμους μαθητές ( Afifah, et al., 2021). Συνολικά, το XGBoost είναι ένας ισχυρός αλγόριθμος που αξιοποιεί την ενίσχυση κλίσης και τα δέντρα αποφάσεων για την επίτευξη προβλέψεων υψηλής απόδοσης. Προσφέρει αποτελεσματική χρήση πόρων, αντιμετωπίζει ανισορροπίες στα δεδομένα και επωφελείται από τη συνδυασμένη προσέγγιση των αδύναμων μαθητών για τη δημιουργία ενός ισχυρού γενικού μοντέλου ( Afifah, et al., 2021).

Δίνεται μία γενικευμένη μαθηματική περιγραφή του αλγορίθμου. Αν είναι τα δεδομένα εκπαίδευσης, yi το σύνολο που θέλουμε να προβλέψουμε, t το πλήθος των επαναλήψεων αυτής της διαδικασίας και η συνάρτηση του νέου μαθητή τότε ο XGBoostClassifier για να βρει την τελική πρόβλεψη ιt κάνει χρήση του τύπου:

*Όπου είναι η προηγούμενη πρόβλεψη και είναι η νέα πρόβλεψη η οποία προστίθεται για την βελτίωση της τελικής πρόβλεψης* ( Afifah, et al., 2021)*. Για την εύρεση μοντέλου με τον XGBoost πρέπει να γίνει ελαχιστοποίηση της αντικειμενικής συνάρτησης.* *Αυτή αντιπροσωπεύει τη συνάρτηση απώλειας στην επανάληψη t. Προσδιορίζει πόσο μακριά είναι οι προβλεπόμενες τιμές από τις πραγματικές τιμές. Διαφορετικοί αλγόριθμοι μπορεί να χρησιμοποιούν διαφορετικές συναρτήσεις απώλειας ανάλογα με το πρόβλημα, όπως το μέσο τετραγωνικό σφάλμα για παλινδρόμηση ή απώλεια καταγραφής για ταξινόμηση:*

*Η οποία περιέχει την l που είναι ουσιαστικά το άθροισμα των τωρινών και προηγούμενων αθροιζόμενων δένδρων και τον όρο . Αυτός ο όρος αντιπροσωπεύει έναν όρο κανονικοποίησης που εφαρμόζεται στο νέο μοντέλο στην επανάληψη t. Η κανονικοποίηση βοηθά στην αποφυγή υπερβολικού overfitting και προωθεί απλούστερα μοντέλα. Η επιλογή της κανονικοποίησης μπορεί να ποικίλλει ανάλογα με τον συγκεκριμένο αλγόριθμο ενίσχυσης κλίσης που χρησιμοποιείται. Έτσι, τελικά προκύπτει ο τύπος:*

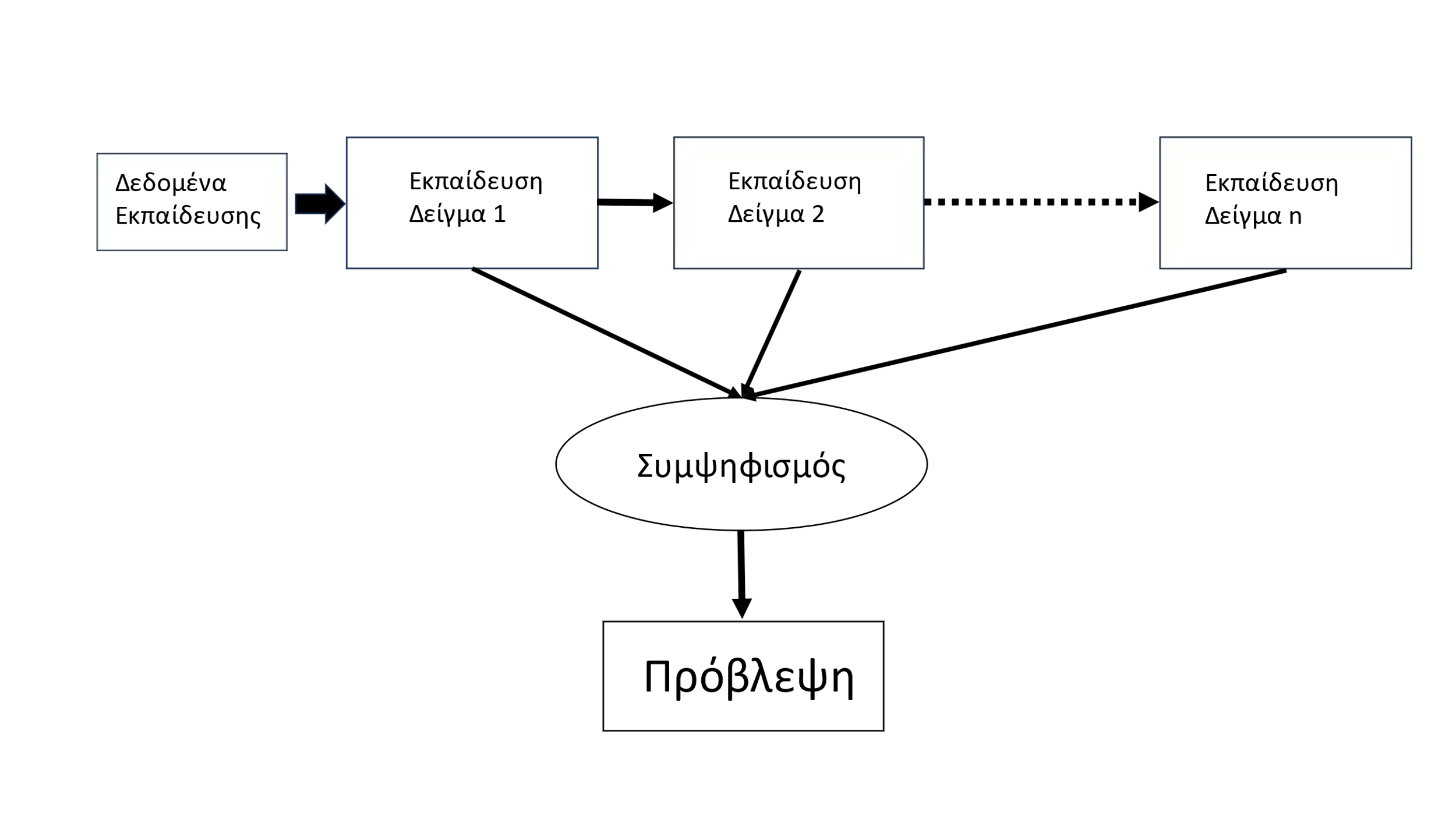
*Η συνάρτηση απώλειας δείχνει το κατά πόσο το μοντέλο που εκπαιδεύτηκε είναι συμβατό με τα δεδομένα ενώ ο όρος κανονικοποίησης αναπαριστά την πολυπλοκότητα των δέντρων* ( Afifah, et al., 2021)*.*

Ο αλγόριθμος αυτός διαθέτει αρκετές υπερπαραμέτρους (Chen & Guestrin, 2016), (@trivialfis, 2023) μερικές από τις οποίες είναι:

* max\_depth: Το μέγιστο βάθος κάθε δένδρου αποφάσεων.
* learning\_rate: Συρρίκνωση μεγέθους βήματος που χρησιμοποιείται στην ενίσχυση. Οι μικρότερες τιμές κάνουν το μοντέλο πιο συντηρητικό.
* n\_estimators: O αριθμός των δένδρων που θα φτιαχτούν.
* subsample: Ρυθμίζει την ύπαρξη overfitting κάνοντας τυχαία επιλογή ενός υποσυνόλου δεδομένων.
* colsample\_bytree: Ρυθμίζει, αν υπάρχει, το overfitting με την επιλογή ομάδας χαρακτηριστικών κατά τυχαίο τρόπο.
* colsample\_bylevel: Είναι μια υπερπαράμετρος που ελέγχει την δειγματοληψία χαρακτηριστικών σε κάθε επίπεδο του δέντρου κατά την κατασκευή δέντρων ενίσχυσης κλίσης. Καθορίζει τον λόγο των χαρακτηριστικών που πρόκειται να δειγματοληφθούν τυχαία για κάθε επίπεδο του δέντρου.
* min\_child\_weight: Ελάχιστο άθροισμα βάρους (hessian) που απαιτείται σε ένα παιδί. Ελέγχει το βάθος του δέντρου και βοηθά στην αποφυγή υπερβολικής προσαρμογής.
* max\_leaves: Μέγιστος αριθμός φύλλων για κάθε δέντρο.

Αυτές είναι μερικές μόνο από τις υπερπαραμέτρους που είναι διαθέσιμες σε αυτό τον αλγόριθμο. Ανάλογα με την υλοποίηση ή τη βιβλιοθήκη που χρησιμοποιείται, ενδέχεται να υπάρχουν πρόσθετες υπερπαράμετροι.

#### RandomForestClassifier.

 Ο αλγόριθμος Random Forest Classifier είναι μια μέθοδος μηχανικής μάθησης που κατασκευάζει πολλά δέντρα αποφάσεων και λαμβάνει την τελική απόφαση βασιζόμενος στην πλειοψηφία των δέντρων που συνθέτουν το «δάσος». Πρόκειται για ένα διάγραμμα δεντρικής δομής που χρησιμοποιείται για τη λήψη αποφάσεων. Κάθε κλαδί του δέντρου αναπαριστά μια πιθανή απόφαση, περίπτωση ή αντίδραση (Sinha, et al., 2020). Ένα από τα βασικά πλεονεκτήματα του αλγορίθμου Random Forest είναι ότι μειώνει τον κίνδυνο overfitting, σε σχέση με άλλους classifiers, και τον απαιτούμενο χρόνο εκπαίδευσης στο σύνολο δεδομένων. Επιπλέον, προσφέρει υψηλή ακρίβεια. Λειτουργεί αποδοτικά σε μεγάλες βάσεις δεδομένων και παράγει σχεδόν ακριβείς προβλέψεις προσεγγίζοντας την πιθανή απουσία δεδομένων (Sinha, et al., 2020). Ένα σχεδιάγραμμα σχετικό με τον τρόπο λειτουργίας του αλγορίθμου αυτού φαίνεται στην **Εικόνα 1**. Εδώ φαίνεται η βασική μέθοδος με την οποία λειτουργεί ο RandomForestClassifier.

**Εικόνα 1**.Συμψηφισμός των επιμέρους δένδρων στον RandomForestClassifier (Sinha, et al., 2020).

Ο αλγόριθμος αυτός διαθέτει πλήθος υπερπαραμέτρων που χρήζουν συντονισμού. Μερικοί από αυτούς είναι οι:

* n\_estimators: O αριθμός των δένδρων αποφάσεων που θα εκπαιδευτεί για την δημιουργία του δάσους.
* criterion: Μία μετρική που εκφράζει την ποιότητα των αποφάσεων.
* max\_depth: Ο μέγιστος αριθμός επιπέδων κάθε δένδρου αποφάσεων στο δάσους.
* min\_samples\_split: Το μικρότερο πλήθος δειγμάτων για να γίνει διάσπαση ενός κόμβου μέσο μίας διαδικασίας απόφασης.
* min\_samples\_leaf: Το μικρότερο πλήθος δειγμάτων στα φύλλα ενός δένδρου αποφάσεων.
* max\_features: Το πλήθος των χαρακτηριστικών που θα χρησιμοποιηθούν για να γίνει η επιλογή της κατάλληλης διαδικασίας απόφασης.

Αυτές είναι μερικές μόνο από τις υπερπαραμέτρους που είναι διαθέσιμες σε αυτό τον αλγόριθμο. Ανάλογα με την υλοποίηση ή τη βιβλιοθήκη που χρησιμοποιείται, ενδέχεται να υπάρχουν πρόσθετες υπερπαράμετροι.

#### GradientBoostingClassifier.

Η μέθοδος της gradient boosting αποτελεί μια ισχυρή τεχνική συνδυασμού μοντέλων προκειμένου να δημιουργηθεί ένα μοντέλο που είναι πιο ακριβές από κάθε μοναδικό απλό μοντέλο ( Gevorkyan, et al., 2020). Μία ομάδα από μοντέλα που χαρακτηρίζονται από απλότητα συντελεί το σύνολο εκμάθησης (ensemble), και κάνοντας χρήση του όρου boosting γίνεται λόγος για μία επαναλαμβανόμενη διαδικασία κατασκευής απλών μοντέλων ( Gevorkyan, et al., 2020). Ο αλγόριθμος gradient boosting είναι ένας ευρέως χρησιμοποιούμενος αλγόριθμος μηχανικής μάθησης. Εδώ, δίνεται μια σύντομη και απλουστευμένη περιγραφή του αλγορίθμου αυτού. Με κάθε εκτέλεση της αλγοριθμικής διαδικασίας του gradient boosting, επιλέγεται και γίνεται ελαχιστοποίηση σε μια συνάρτηση απώλειας μέσω της μεθόδου κλίσης η οποία συνάρτηση απώλειας φτιάχνεται για κάθε ένα από τα επιμέρους απλοϊκά μοντέλο. Συνήθως, η επιλογή αυτού του απλού μοντέλου είναι ένα μοντέλο δέντρου αποφάσεων. Κατασκευάζοντας το επόμενο μοντέλο συνυπολογίζονται και τα λάθη του προηγούμενου μοντέλου. Αυτό γίνεται με το να εξακριβώνονται τα δεδομένα που δεν εξηγούνται καλά από τα προηγούμενα μοντέλα και κατασκευάζοντας τα επόμενα μοντέλα ώστε να εξηγούν αυτά τα υπολειπόμενα δεδομένα ( Gevorkyan, et al., 2020). Κατά την προετοιμασία του αλγορίθμου, δίνεται και η τιμή του μέγιστου πλήθους δέντρων στο σύνολο εκμάθησης, και όταν επιτευχθεί η εκπαίδευση αυτού του πλήθους δέντρων ολοκληρώνεται και η εκτέλεση του αλγόριθμου. Κάθε μοντέλο στο σύνολο εκμάθησης αντιστοιχεί σε ένα συγκεκριμένο βάρος και οι προβλέψεις τους συνδυάζονται για να παράγουν μια γενικευμένη πρόβλεψη ( Gevorkyan, et al., 2020). Η βιβλιοθήκη sklearn παρέχει μια συνάρτηση που υλοποιεί τον αλγόριθμο gradient boosting με το όνομα GradientBoostingClassifier.

Ο αλγόριθμος αυτός είναι παρόμοιος με τον XGBoostClassifier όμως υπάρχουν ορισμένες σημαντικές διαφορές βασικές εκ των οποίων είναι:

* Η υλοποίηση του αλγορίθμου. Ο XGBoostClassifier είναι μια βελτιστοποιημένη υλοποίηση της μεθόδου ενίσχυσης κλίσης, σχεδιασμένη για να βελτιώσει την ταχύτητα και την απόδοση. Χρησιμοποιεί έναν παραλληλισμένο αλγόριθμο κατασκευής δέντρων και άλλες προηγμένες τεχνικές για να ενισχύσει την αποδοτικότητα. Από την άλλη πλευρά, ο GradientBoostingClassifier είναι μέρος της βιβλιοθήκης scikit-learn και παρέχει μια πιο απλή-γραμμική υλοποίηση της μεθόδου ενίσχυσης κλίσης.
* Η προσαρμογή των υπερπαραμέτρων. Ο XGBoostClassifier παρέχει περισσότερες υπερπαραμέτρους για την αρμονική ρύθμιση του μοντέλου, επιτρέποντας έτσι μεγαλύτερο έλεγχο στη διαδικασία ενίσχυσης κλίσης. Επίσης, προσφέρει μια ευρεία γκάμα παραμέτρων για την κατασκευή δέντρων. Ο GradientBoostingClassifier παρέχει επίσης υπερπαραμέτρους για τη ρύθμιση του μοντέλου, αλλά ο χώρος των υπερπαραμέτρων είναι σχετικά μικρότερος.

Γενικά, οι δύο αυτοί αλγόριθμοι έχουν ομοιότητες και διαφορές, και η επιλογή ανάμεσά τους εξαρτάται από το πρόβλημα που αντιμετωπίζετε και τις συγκεκριμένες απαιτήσεις.

Ο GradientBoostingClassifier έχει ως υπερπαραμέτρους τα εξής:

* n\_estimators: O αριθμός των δένδρων που θα φτιαχτούν.
* learning\_rate: Ο ρυθμός μάθησης συρρικνώνει τη συμβολή κάθε δέντρου στο σύνολο. Ένας χαμηλότερος ρυθμός εκμάθησης απαιτεί συνήθως να προστεθούν περισσότερα δέντρα στο σύνολο.
* max\_depth: Το μέγιστο βάθος κάθε δένδρου αποφάσεων. Βοηθά στην αποφυγή του overfitting.
* subsample: Το μέρος των δεδομένων που θα χρησιμοποιηθεί για την εκπαίδευση κάθε μεμονωμένου δέντρου.
* min\_samples\_split: O μικρότερος αριθμός απαιτούμενων δειγμάτων για την διάσπαση ενός κόμβου μέσο μίας διαδικασίας απόφασης.
* max\_features: Ο αριθμός των χαρακτηριστικών που θα ληφθούν υπόψιν κατά την επιλογή της διαδικασίας απόφασης.
* min\_samples\_leaf: O μικρότερος αριθμός απαιτούμενων δειγμάτων των φύλλων ενός δένδρου αποφάσεων.

Αυτές είναι μερικές μόνο από τις υπερπαραμέτρους που είναι διαθέσιμες για τον GradientBoostingClassifier. Όμως είναι σημαντικό για την συγκεκριμένη έρευνα να γίνει η παρατήρηση πως αυτά τα μοντέλα μηχανικής μάθησης παρουσιάζουν ομοιότητες στις υπερπαραμέτρους τους που οφείλονται κυρίως στο ότι κάνουν χρήση δένδρων αποφάσεων για την τελική πρόβλεψη.

#### RandomizedSearchCV.

Ο αλγόριθμος τυχαίας αναζήτησης είναι ένας αλγόριθμος που χρησιμοποιείται κατά κόρον για την εύρεση βέλτιστων υπερπαραμέτρων σε μοντέλα μηχανικής μάθησης (Rahmadayana & Sibaroni, 2021). Στην συγκεκριμένη εργασία επιλέχτηκε να γίνει χρήση της υλοποίησης που προσφέρει η scikit-learn με τoν RandomizedSearchCV. Αυτός ο αλγόριθμος εκτελεί την αναζήτηση δημιουργώντας τυχαίους συνδυασμούς υπερπαραμέτρων από τον ορισμένο χώρο αναζήτησης και αξιολογεί την απόδοση του μοντέλου χρησιμοποιώντας διασταυρούμενη επικύρωση (cross-validation). Εκπαιδεύει τον εκτιμητή (estimator) με κάθε συνδυασμό υπερπαραμέτρων, υπολογίζει το score βάσει της καθορισμένης μετρικής και καταγράφει τον καλύτερο συνδυασμό υπερπαραμέτρων και το αντίστοιχο score που έχει επιτευχθεί μέχρι στιγμής. Στην συνέχεια, επιλέγεται ο καλύτερος συνδυασμός υπερπαραμέτρων που παρήγαγε το υψηλότερο score βάσει της καθορισμένης μετρικής και τέλος εκπαιδεύεται ξανά το μοντέλο με ολόκληρο το σύνολο εκπαίδευσης χρησιμοποιώντας τις βέλτιστες υπερπαραμέτρους.

Ο RandomizedSearchCV έχει παραμέτρους για τη ρύθμιση της διαδικασίας εύρεσης υπερπαραμέτρων του αλγορίθμου που εξετάζεται και αυτές είναι (ARAUNA, 2021):

* Ο εκτιμητής (estimator), ο οποίος ουσιαστικά είναι το μοντέλο μηχανικής μάθησης για το οποίο θα γίνει η αναζήτηση των βέλτιστων υπερπαραμέτρων. Δημιουργείται για κάθε συνδυασμό υπερπαραμέτρων που εξετάζεται.
* Οι πιθανές τιμές των υπερπαραμέτρων (param\_distributions), που είναι ένα λεξικό (dictionary) με λέξεις κλειδιά τις υπερπαραμέτρους του μοντέλου μηχανικής μάθησης και τιμές μία λίστα τιμών για τις υπερπαραμέτρους αυτές. Αυτός είναι ο χώρος αναζήτησης των υπερπαραμέτρων. Η τυχαία δειγματοληψία των συνδυασμών των υπερπαραμέτρων γίνεται με ομοιόμορφο τρόπο.
* Ο αριθμός των ταυτόχρονων εργασιών (n\_jobs) ο οποίος καθορίζει τον αριθμό των ταυτόχρονων εργασιών που θα εκτελεστούν κατά τη διάρκεια της βελτιστοποίησης των υπερπαραμέτρων. Με την τιμή n\_jobs=1, οι εργασίες εκτελούνται σειριακά, ενώ με μεγαλύτερη τιμή (π.χ. 4) εκτελούνται παράλληλα, εκμεταλλευόμενες τυχόν πολλαπλούς πυρήνες της CPU.
* Η αναλυτικότητα της εκτέλεσης (verbose) ελέγχει το επίπεδο αναλυτικότητας κατά την εκτέλεση της βελτιστοποίησης. Μεγαλύτερες τιμές προκαλούν την εμφάνιση περισσότερων μηνυμάτων κατά τη διάρκεια του υπολογισμού
* Η μέθοδος αξιολόγησης (scoring) καθορίζει τη στρατηγική αξιολόγησης της απόδοσης του μοντέλου μετά την αναζήτηση των υπερπαραμέτρων. Οι τιμές που παίρνει είναι συνήθως μία γραμματοσειρά που περιγράφει την μετρική που θα χρησιμοποιηθεί.
* Ο δείκτης διασταυρούμενης επικύρωσης (cv) που καθορίζει πόσα βήματα θα έχει η διαδικασία της επικύρωσης κάθε μοντέλου που θα δημιουργείται καθώς και το πλήθος των κομματιών στα οποία θα χωριστούν τα δεδομένα εκπαίδευσης για να γίνει η επικύρωση.
* Ο αριθμός του τυχαίου δείγματος συνδυασμών υπερπαραμέτρων (n\_iter) που καθορίζει το πλήθος των μοντέλων που θα συγκριθούν για την εύρεση του βέλτιστου συνδυασμού.

Όλες τους σημαντικές για την απόδοση του αλγορίθμου.

#### BayesSearchCV.

Σε αυτή τη μελέτη χρησιμοποιείται η τεχνική Bayes search cross-validation (CV) με την υλοποίηση που παρέχεται από τη βιβλιοθήκη scikit-optimize. Ο αλγόριθμος BayesSearchCV, αντίθετα από τον GridsearchCV που εξερευνά όλους τους συνδυασμούς υπερπαραμέτρων, χρησιμοποιεί έναν Bayesian αλγόριθμο όπου η αναζήτηση γίνεται με χρήση ενός σταθερού αριθμού περιπτώσεων υπερπαραμέτρων που η δειγματοληψία τους γίνεται από μια κατανομή τους. Πιο συγκεκριμένα, ο αλγόριθμος ξεκινά με την κατασκευή ενός αρχικού μοντέλου πρόβλεψης, συνήθως χρησιμοποιώντας μια γκαουσιανή διαδικασία ή ένα μοντέλο βασισμένο σε δέντρα ( Brochu, et al., 2010). Αυτό το μοντέλο πρόβλεψης προσεγγίζει την άγνωστη αντικειμενική συνάρτηση και παρέχει μια εκτίμηση για τη συμπεριφορά της στον χώρο αναζήτησης ( Brochu, et al., 2010). Στην συνέχεια, γίνεται επιλογή του επόμενου σημείου στο δείγμα των υπερπαραμέτρων που αναμένεται να παρουσιάσει την καλύτερη απόδοση βάσει του μοντέλου πρόβλεψης ( Brochu, et al., 2010). Αυτή η επιλογή γίνεται βάσει μιας συνάρτηση απόκτησης η οποία ισορροπεί μεταξύ εξερεύνησης (δειγματοληψία σε ανεξερεύνητες περιοχές) και εκμετάλλευσης (δειγματοληψία σε ελπιδοφόρες περιοχές) του χώρου αναζήτησης. Έπειτα, γίνεται η αξιολόγηση της αντικειμενικής συνάρτησης που βρέθηκε για το σημείο του χώρου αναζήτησης που επιλέχτηκε ( Brochu, et al., 2010). Η αντικειμενική συνάρτηση παρέχει την πραγματική τιμή απόδοσης του συστήματος για αυτήν την συγκεκριμένη ομάδα υπερπαραμέτρων. Κατόπιν αυτού ακολουθεί η ενημέρωση του μοντέλου πρόβλεψης βάσει των νέων δεδομένων που προέκυψαν δηλαδή του σημείου δείγματος και της αντίστοιχης τιμής απόδοσης έτσι μέσω μιας αναδρομικής διαδικασίας, το μοντέλο πρόβλεψης βελτιώνεται σταδιακά, προσαρμόζοντας τις υπερπαραμέτρους του για να προσεγγίσει ακριβέστερα την πραγματική αντικειμενική συνάρτηση ( Brochu, et al., 2010). Τα βήματα αυτά επαναλαμβάνονται για έναν προκαθορισμένο αριθμό επαναλήψεων ή μέχρι να επιτευχθεί μια συγκεκριμένη συνθήκη τερματισμού, όπως η σύγκλιση του μοντέλου.

Ο αλγόριθμος αυτός απαιτεί τον καθορισμό του χώρου αναζήτησης των υπερπαραμέτρων, του αριθμού των βημάτων για την διασταυρούμενη επικύρωση και το μοντέλο που θα βελτιστοποιηθεί. Οι παράμετροι που ρυθμίζουν όλα αυτά είναι οι ίδιοι που αναφέρθηκαν και για τον RandomizedSearchCV στην παράγραφο 2) με μόνη διαφορά στην ονομασία της παραμέτρου (param\_distributions) η οποία για τον BayesSearchCV ονομάζεται (search\_spaces).

#### TPOTClassifier.

Ο συντονισμός υπερπαραμέτρων των μοντέλων μηχανικής μάθησης κάνοντας χρήση γενετικών αλγορίθμων είναι μια μέθοδος που συνδυάζει μια ενημερωμένη αναζήτηση με μία από τις βασικότερες αρχές της γενετικής, την επιβίωση του ισχυρότερου. Η διαδικασία συντονισμού των υπερπαραμέτρων ξεκινά με τη δημιουργία μερικών μοντέλων που αποτελούν την πρώτη γενιά, στην συνέχεια γίνεται η επιλογή των καλύτερων από αυτά και έπειτα η δημιουργία νέων μοντέλων που είναι παρόμοια με αυτά που επιλέχτηκαν ως τα καλύτερα από την προηγούμενη γενιά κάτι που γίνεται τροποποιώντας ελάχιστα τα χαρακτηριστικά της προηγούμενης γενιάς για να προκύψει η νέα (Özdemir, et al., 2023). Στο τέλος, αφού έχουν δημιουργηθεί τα μοντέλα όλων των γενεών γίνεται η αξιολόγηση τους βάσει μίας μετρικής για να προκύψει το μοντέλο με τις βέλτιστες υπερπαραμέτρους. Αυτή η διαδικασία συνεχίζεται είτε για πεπερασμένο αριθμό γενεών ή μέχρι να επιτευχθεί το επιθυμητό αποτέλεσμα δηλαδή να εκπαιδευτεί μοντέλο με ικανοποιητική επίδοση. Η βιβλιοθήκη TPOT (jay-m-dev, 2023) αποτελεί ένα εργαλείο που μπορεί να χρησιμοποιηθεί για την πρόβλεψη των βέλτιστων τιμών υπερπαραμέτρων χρησιμοποιώντας έναν εξελικτικό αλγόριθμο. Ο αλγόριθμος αυτός επιλέγει το καλύτερο μοντέλο μετά από μάθηση από προηγούμενες επαναλήψεις. Ωστόσο, είναι σημαντικό να σημειωθεί ότι η εκτέλεση της TPOT μπορεί να απαιτεί σημαντικούς υπολογιστικούς πόρους. Σε αυτήν τη μελέτη, ο TPOTClassifier χρησιμοποιήθηκε για την εύρεση των βέλτιστων μοντέλων σε κάθε σύνολο δεδομένων αντίστοιχα. Είναι ένας αλγόριθμος που ενισχύει όχι μόνο η διαδικασία επιλογής του καλύτερου μοντέλου με βάση την υπολογισμένη ακρίβεια, αλλά και την ρύθμιση των παραμέτρων του μοντέλου με τη χρήση γενετικού αλγορίθμου (Özdemir, et al., 2023).

Τα βήματα εκτέλεσης του TPOTClassifier ρυθμίζονται από τις διάφορες παραμέτρους που τον απαρτίζουν. Σημαντικές εκ το οποίων είναι οι (jay-m-dev, 2023):

* generations: Αναπαριστά τον αριθμό των γενεών (επαναλήψεων) που θα εκτελέσει ο αλγόριθμος TPOTClassifier. Κάθε γενιά περιλαμβάνει την αξιολόγηση και την εξελικτική διαδικασία ενός πληθυσμού μοντέλων.
* template: Αυτή η παράμετρος δίνει την δυνατότητα να καθοριστεί ένα συγκεκριμένο πρότυπο για τον αγωγό (pipeline) μηχανικής εκμάθησης που θα εξελίξει και θα βελτιστοποιήσει ο TPOTClassifier κατά τη διαδικασία αναζήτησης. Για τις ανάγκες της συγκεκριμένης εργασίας δόθηκε η τιμή 'Classifier' ώστε να γίνει η διαδικασία της βελτιστοποίησης υπερπαραμέτρων για τα ζητούμενα μοντέλα μηχανικής μάθησης.
* population\_size: Καθορίζει τον αριθμό των ατόμων (μοντέλων) σε κάθε γενιά. Ένας μεγαλύτερος αριθμός ατόμων μπορεί να οδηγήσει σε καλύτερες λύσεις, αλλά απαιτεί περισσότερους υπολογιστικούς πόρους.
* offspring\_size: Καθορίζει τον αριθμό των απογόνων που παράγονται από τα κορυφαία μοντέλα κάθε γενιάς. Οι απόγονοι είναι νέα μοντέλα που δημιουργούνται εφαρμόζοντας γενετικούς τελεστές, όπως generations: Αναπαριστά τον αριθμό των γενεών (επαναλήψεων) που θα εκτελέσει ο αλγόριθμος TPOT. Κάθε γενιά περιλαμβάνει την αξιολόγηση και την εξέλιξη μιας πληθυσμού μοντέλων.
* verbosity: Ελέγχει το επίπεδο λεπτομέρειας των πληροφοριών που δίνει ο TPOTClassifier ως έξοδο. Υψηλότερες τιμές παρέχουν περισσότερες λεπτομέρειες σχετικά με τη διαδικασία βελτιστοποίησης και τα μοντέλα που προέκυψαν.
* early\_stop: Καθορίζει τον αριθμό των γενεών χωρίς βελτίωση στη διαδικασία βελτιστοποίησης που θα προκαλέσει έγκαιρο τερματισμό. Αν ο αλγόριθμος δεν βρει καλύτερα μοντέλα για τον καθορισμένο αριθμό γενεών, θα σταματήσει την αναζήτηση.
* config\_dict: Είναι ένα λεξικό που καθορίζει τον χώρο αναζήτησης των υπερπαραμέτρων για κάποιο μοντέλο μηχανικής μάθησης από τον TPOTClassifier. .
* cv: Καθορίζει τον αριθμό των διασταυρούμενων επαναλήψεων που χρησιμοποιούνται κατά την αξιολόγηση κάθε μοντέλου. Η διασταύρωση επιτρέπει την εκτίμηση της απόδοσης των μοντέλων σε μη εκπαιδευμένα δεδομένα και μειώνει τον κίνδυνο overfitting.
* scoring: Καθορίζει τη μετρική αξιολόγησης που χρησιμοποιείται για την εκτίμηση της απόδοσης κάθε μοντέλου. Σε αυτήν την μελέτη θα χρησιμοποιείται η μετρική "ακρίβεια" (accuracy) για την αξιολόγηση των προβλέψεων του μοντέλου.

Αυτές είναι οι βασικές παράμετροι του TPOTClassifier που περιγράφονται στην συγκεκριμένη εργασία. Βάσει αυτών των παραμέτρων θα τεθεί το πλαίσιο στο οποίο θα γίνει η διαδικασία της βελτιστοποίησης για κάθε σύνολο δεδομένων από τον TPOTClassifier.

### Μετρικές και πίνακες.

#### Accuracy.

Μία από τις μετρικές που θα χρησιμοποιηθούν για την σύγκριση των αλγορίθμων είναι και η ακρίβεια. Ποιο συγκεκριμένα θα χρησιμοποιηθούν η ακρίβεια επικύρωσης που προκύπτει κατά την διάρκεια εκπαίδευσης ενός μοντέλου, η ακρίβεια πρόβλεψης του μοντέλου στα δεδομένα εκπαίδευσης και η ακρίβεια πρόβλεψης του μοντέλου στα δεδομένα ελέγχου.

Η ακρίβεια επικύρωσης προκύπτει από την διαδικασία της διασταυρούμενης επικύρωσης γνωστή και ως μεθοδολογία περιστροφής, στην οποία το σύνολο δεδομένων D χωρίζεται τυχαία σε k μη επικαλυπτόμενα υποσύνολα (folds) D1, D2, ..., Dk περίπου ίσου μεγέθους ( Kohavi, 1995). Το μοντέλο εκπαιδεύεται k φορές, κάθε φορά εκπαιδεύοντας το στο D\Dt όπου και ελέγχοντας το στο Dt, έτσι η εκτίμηση της ακρίβειας στην διασταυρούμενη επικύρωση είναι το συνολικό πλήθος σωστών ταξινομήσεων, διαιρούμενο με τον αριθμό των παραδειγμάτων που χρησιμοποιήθηκαν από το σύνολο δεδομένων ( Kohavi, 1995). Έστω ότι D(i) είναι το σύνολο δοκιμής για το xi = (ui , yi) όπου ui το διάνυσμα των χαρακτηριστικών και yi ο στόχος που θέλουμε να προβλέψουμε και αν i = j ενώ αν i ≠ j τότε . Η ακρίβεια επικύρωσης τότε θα υπολογίζεται ως εξής ( Kohavi, 1995):

*όπου Μ είναι το μοντέλο που εξετάζεται και n το πλήθος των υποσυνόλων. Αυτή η μέθοδος παρέχει μία πιο βέβαιη μετρική για το πόσο καλά συμπεριφέρεται ένα μοντέλο στην πρόβλεψη των δεδομένων καθώς περιγράφει τον μέσο όρο απόδοσης σε διαφορετικά σύνολα δεδομένων πάνω σε ένα πρόβλημα.*

*Όσον αφορά την ακρίβεια των προβλέψεων στα δεδομένα εκπαίδευσης και στα δεδομένα ελέγχου αντίστοιχα αυτές υπολογίζονται από τον τύπο:*

*όπου TP, TN είναι το πλήθος των σωστών προβλέψεων του μοντέλου για τις δύο αντίστοιχες κατηγορίες για τις οποίες έπρεπε να γίνει διαχωρισμός και FP, FN είναι το πλήθος των λανθασμένων προβλέψεων για τις δύο κατηγορίες* (Sinha, et al., 2020)*.*

#### MAE.

Το μέσο απόλυτο σφάλμα (MAE) δεν χρησιμοποιείται συνήθως για μοντέλα ταξινόμησης. Το MAE χρησιμοποιείται κυρίως σε εργασίες παλινδρόμησης για τη μέτρηση της μέσης απόλυτης διαφοράς μεταξύ των προβλεπόμενων και των πραγματικών τιμών. Στις εργασίες ταξινόμησης, η εστίαση είναι στην πρόβλεψη των κατηγοριών και όχι στην πρόβλεψη συνεχών τιμών. Παρόλα αυτά, πραγματοποιήθηκε ο υπολογισμός και αυτής της μετρικής ως μία δευτερεύουσα μορφή σύγκρισης των μοντέλων που προέκυψαν.

Το μέσο απόλυτο σφάλμα υπολογίζεται από τον τύπο ( Chai & Draxler, 2014):

όπου το σφάλμα κάθε φοράκαι n το πλήθος των δειγμάτων.

#### R2 και R.

Ο υπολογισμός του R2 γίνεται από τους μαθηματικούς τύπους (Bakas, et al., 2023):

, , ,

με m το πλήθος των δειγμάτων.

Πρώτα υπολογίζεται ο μέσος του Υtr που είναι ένα διάνυσμα που περιέχει τις τιμές του χαρακτηριστικού που θέλουμε να προβλέψουμε και στην συνέχεια υπολογίζονται το SSres που είναι το άθροισμα των σφαλμάτων για κάθε πρόβλεψη στο τετράγωνο και το SStot που είναι το άθροισμα των υψωμένων στο τετράγωνο διαφορών του Ytr με τον μέσο του και τέλος υπολογίζεται το R2.

Το R2 είναι ένας αριθμός μεταξύ του 0 και 1 και αυτό που περιγράφει είναι το κατά πόσο η καμπύλη της παλινδρόμησης προσαρμόζεται στα δεδομένα. Όταν είναι κοντά στην τιμή 1 σημαίνει πως υπάρχει πολύ καλή προσαρμογή του μοντέλου στα δεδομένα ενώ όταν βρίσκεται κοντά στο 0 δεν υπάρχει καλή προσαρμογή στα δεδομένα. Ωστόσο, είναι σημαντικό να σημειωθεί ότι το R2 θα πρέπει να ερμηνεύεται με προσοχή και να χρησιμοποιείται σε συνδυασμό με άλλες μετρικές αξιολόγησης. Μπορεί να επηρεαστεί από τον αριθμό των παραγόντων του μοντέλου και μπορεί να μην είναι η καταλληλότερη μέτρηση σε ορισμένες περιπτώσεις. Στις εργασίες ταξινόμησης, το R2 δεν χρησιμοποιείται συνήθως ως μέτρηση αξιολόγησης όμως στην περίπτωση αυτής της εργασίας χρησιμοποιείται με την ίδια λογική με το μέσο απόλυτο σφάλμα, δηλαδή ως μία δευτερεύουσα μορφή σύγκρισης των μοντέλων.

Το R ή συντελεστής συσχέτισης Pearson ( PLEVRIS, et al., 2022) υπολογίζεται χρησιμοποιώντας τον τύπο:

όπου είναι οι επιμέρους όροι των διανυσμάτων x,y και οι μέση όροι των επιμέρους όρων των διανυσμάτων.

Ο συντελεστής συσχέτισης Pearson είναι ένα μέτρο της γραμμικής σχέσης μεταξύ δύο μεταβλητών ( PLEVRIS, et al., 2022). Κυμαίνεται από -1 έως 1, όπου το -1 υποδηλώνει μια τέλεια αρνητική γραμμική σχέση, το 1 δείχνει μια τέλεια θετική γραμμική σχέση και το 0 δείχνει καμία γραμμική σχέση. Στα πλαίσια της εργασίας θα χρησιμοποιηθεί κι αυτός ως μία μορφή σύγκρισης των μοντέλων. Τόσο το R όσο και το R2 είναι ικανά να υποδείξουν την ύπαρξη σφαλμάτων σε προβλήματα κατηγοριοποίησης αν ερμηνευτούν κατάλληλα.

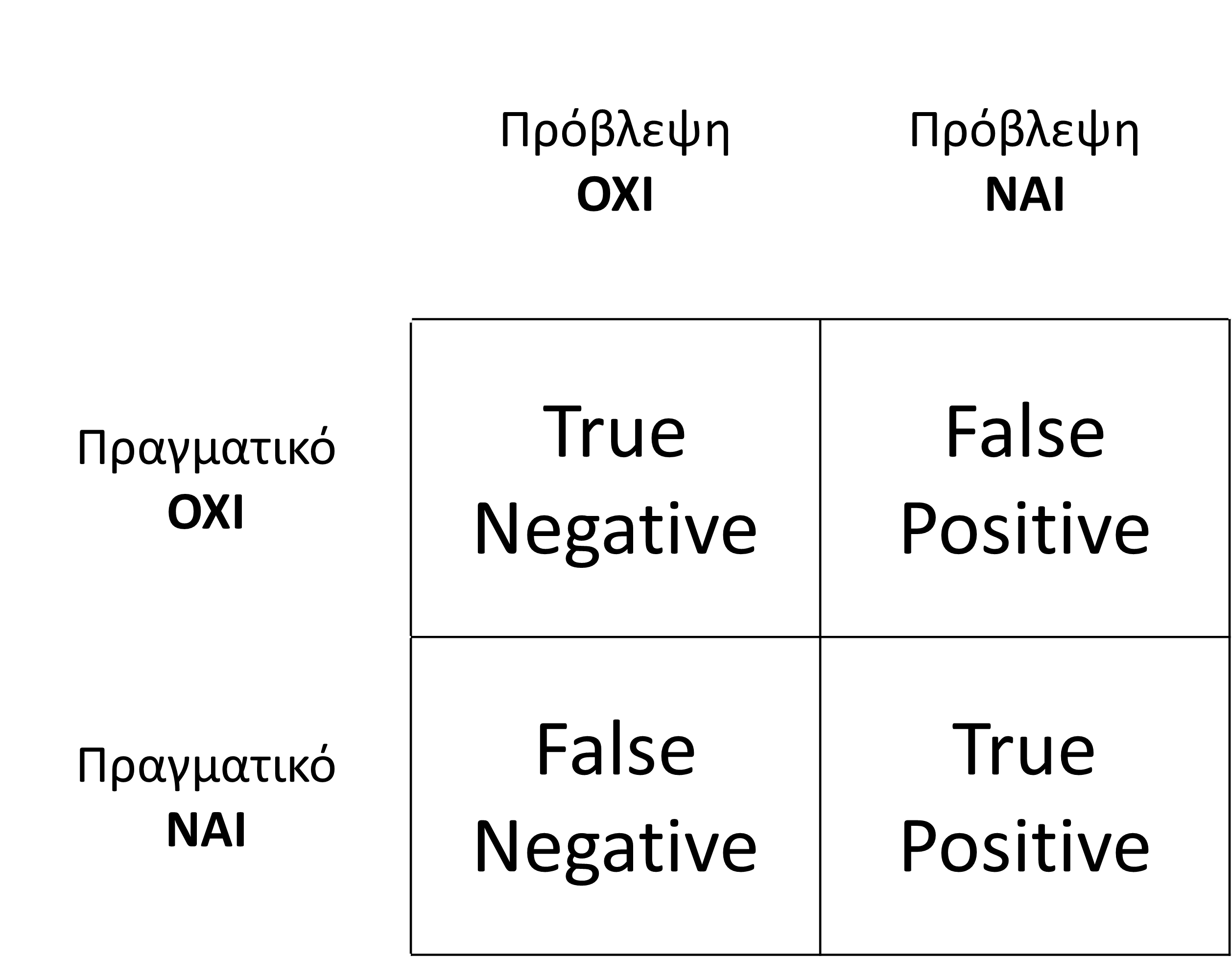
#### P-value.

Επίσης στην εργασία αυτή γίνεται χρήση του P-value που η τιμή του είναι ενδεικτική της σχέσης δύο μεταβλητών (Thisted, 1998), μίας εξαρτημένης και μίας ανεξάρτητης. Στην περίπτωσή μας η ανεξάρτητη μεταβλητή είναι οι πραγματικές τιμές του στόχου στο σύνολο ελέγχου και η εξαρτημένη είναι οι προβλέψεις των τιμών του στόχου που έκανε ένα μοντέλο μηχανικής μάθησης. Υποδηλώνει μια στατιστική συσχέτιση μεταξύ των μεταβλητών. Εάν το p-value είναι μεγάλο, υποδηλώνει ότι η παρατηρούμενη συσχέτιση μπορεί να οφείλεται σε τυχαία πιθανότητα και δεν παρέχει ισχυρές ενδείξεις συσχέτισης. Η αριθμητική τιμή αυτού του δείκτη στα πλαίσια αυτής της εργασίας θα είναι μία ένδειξη, όταν οι τιμή είναι μικρή, ότι οι προβλέψεις που δόθηκαν από τα μοντέλα είναι σχετικές με τις πραγματικές τιμές δηλαδή η πρόβλεψη σαφώς δεν γίνεται τυχαία ενώ αν είναι μεγάλη, με σημαντικότητα σφάλματος α=0,05, θα συμβαίνει το αντίθετο δηλαδή οι προβλέψεις του μοντέλου θα θεωρούνται χειρότερες από αυτές μίας τυχαίας πρόβλεψης (Thisted, 1998).

#### Πίνακες σύγχυσης.

Ο πίνακας σύγχυσης (confusion matrix) είναι ένας πίνακας που χρησιμοποιείται συχνά για την αξιολόγηση της απόδοσης ενός μοντέλου ταξινόμησης ( Visa, et al., 2011). Απεικονίζει τον αριθμό των σωστών και λανθασμένων προβλέψεων που έκανε το μοντέλο σε ένα σύνολο δεδομένων ελέγχου. Ένας τυπικός πίνακας σύγχυσης έχει δύο διαστάσεις, τις πραγματικές ετικέτες κλάσεων και τις προβλεπόμενες ετικέτες κλάσεων. Οι γραμμές αντιπροσωπεύουν τις πραγματικές ετικέτες κλάσεων των δεδομένων, ενώ οι στήλες αντιπροσωπεύουν τις ετικέτες κλάσεων που προβλέφθηκαν από το μοντέλο. Οι τέσσερις περιοχές ή τεταρτημόρια του πίνακα σύγχυσης ορίζονται ως εξής **Εικόνα 2**:

1. Αληθώς Θετικό (True Positive, TP): Το μοντέλο προέβλεψε σωστά τη θετική κλάση-ομάδα.
2. Αληθώς Αρνητικό (True Negative, TN): Το μοντέλο προέβλεψε σωστά την αρνητική κλάση-ομάδα.
3. Ψευδώς Θετικό (False Positive, FP): Το μοντέλο προέβλεψε λανθασμένα τη θετική κλάση-ομάδα όταν η πραγματική κλάση-ομάδα ήταν αρνητική (Σφάλμα τύπου Ι).
4. Ψευδώς Αρνητικό (False Negative, FN): Το μοντέλο προέβλεψε λανθασμένα την αρνητική κλάση-ομάδα όταν η πραγματική κλάση-ομάδα ήταν θετική (Σφάλμα τύπου ΙΙ).

Με τους όρους θετική και αρνητική στην περίπτωση της εργασίας εννοούνται οι ετικέτες 0 και 1 αντίστοιχα. Οι τιμές στα κελιά του πίνακα σύγχυσης αντιπροσωπεύουν τον αριθμό ή τη συχνότητα των δειγμάτων που ανήκουν σε κάθε κατηγορία. Μέσω της ανάλυσης του πίνακα σύγχυσης, μπορούν να προκύψουν διάφορες μετρικές απόδοσης, όπως η ακρίβεια (accuracy), η ορθότητα (precision), η ανάκληση (recall) και το F1-score, που παρέχουν πληροφορίες για την απόδοση του μοντέλου όσον αφορά την ικανότητά του να ταξινομεί σωστά διάφορες κατηγορίες. Ο πίνακας σύγχυσης είναι ένα χρήσιμο εργαλείο για την κατανόηση των δυνατοτήτων και αδυναμιών ενός μοντέλου ταξινόμησης και μπορεί να βοηθήσει στη λήψη αποφάσεων για την προσαρμογή του μοντέλου ή τον καθορισμό κατάλληλων ορίων κατά την διαδικασία εκπαίδευσης του μοντέλου βάσει των συγκεκριμένων απαιτήσεων της εφαρμογής.

**Εικόνα 2**.Δομή πίνακα σύγχυσης.

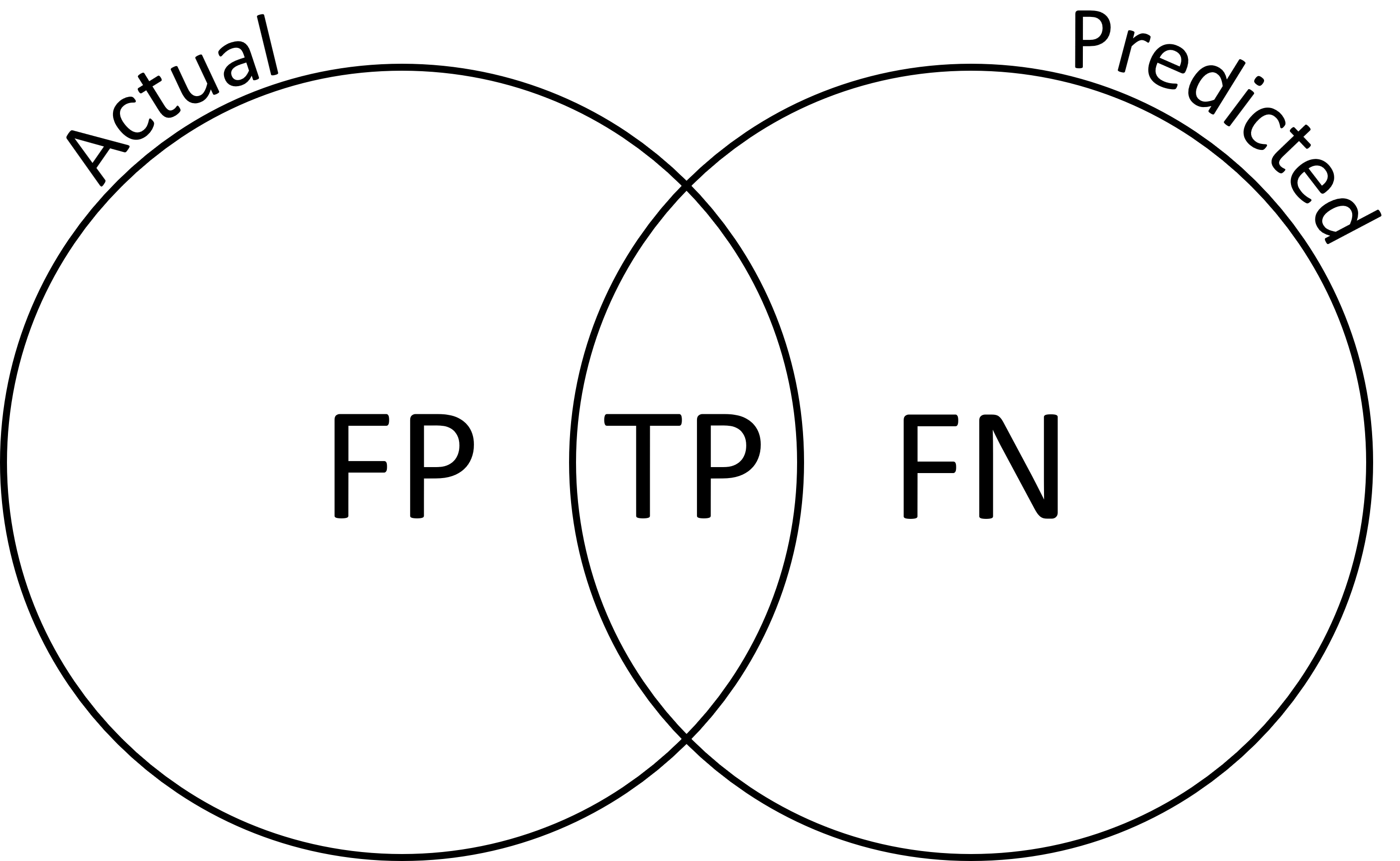
#### Precision and Recall.

Η ορθότητα (precision) και η ανάκληση (recall) είναι βασικές μετρικές αξιολόγησης που χρησιμοποιούνται στη μηχανική μάθηση, ιδιαίτερα σε κατηγοριοποίηση δυαδικών προβλημάτων, για να αξιολογηθεί η απόδοση ενός μοντέλου. Παρέχουν αξιόλογες πληροφορίες για την ικανότητα του μοντέλου να πραγματοποιεί ακριβείς προβλέψεις και να αναγνωρίζει σωστά τις θετικές περιπτώσεις. Ενώ η ορθότητα επικεντρώνεται στην ελαχιστοποίηση των ψευδών θετικών (false positive), η ανάκληση επικεντρώνεται στην ελαχιστοποίηση των ψευδών αρνητικών (false negative). Η κατανόηση και η βελτιστοποίηση και των δύο μετρικών είναι ουσιώδης για τη δημιουργία αξιόπιστων και αποτελεσματικών μοντέλων κατηγοριοποίησης. Ας εξετάσουμε όμως πιο αναλυτικά την ορθότητα και την ανάκληση, καθώς και τον τρόπο υπολογισμού και ερμηνείας τους.

Η ορθότητα, επίσης γνωστή ως θετική προβλεπτική αξία (positive predictive value), μετρά το ποσοστό των σωστά προβλεπόμενων θετικών περιπτώσεων μεταξύ όλων των περιπτώσεων που προβλέπονται ως θετικές από το μοντέλο. Αξιολογεί την ικανότητα του μοντέλου να ελαχιστοποιεί τις ψευδείς θετικές και να αποφεύγει την κατηγοριοποίηση των αρνητικών περιπτώσεων ως θετικές. Για τον υπολογισμό της ορθότητας, λαμβάνουμε υπόψη τα ακόλουθα στοιχεία:

* Αληθώς Θετικά (TP): Ο αριθμός των περιπτώσεων που είναι πραγματικά θετικές και προβλέπονται σωστά ως θετικές από το μοντέλο.
* Ψευδώς Θετικά (FP): Ο αριθμός των περιπτώσεων που είναι πραγματικά αρνητικές, αλλά προβλέπονται λανθασμένα ως θετικές από το μοντέλο.

Έχοντας αυτά κατά νου και προσεγγίζοντας τον ορισμό βάσει θεωρίας συνόλων **Εικόνα 3**, η ορθότητα υπολογίζεται με τον ακόλουθο τύπο ( Fränti & Mariescu-Istodor, 2023):



**Εικόνα 3.** Διάγραμμα Venn του συνόλου προβλέψεων και των πραγματικών τιμών των δεδομένων.

Ένα υψηλό ποσοστό ακρίβειας υποδεικνύει ότι το μοντέλο έχει χαμηλό ρυθμό ψευδών θετικών, δείχνοντας ότι όταν προβλέπει μια περίπτωση ως θετική, είναι πιθανό να είναι σωστή. Αντίθετα, ένα χαμηλό ποσοστό ακρίβειας υποδεικνύει υψηλό αριθμό ψευδών θετικών, υποδεικνύοντας ότι το μοντέλο μπορεί να κατηγοριοποιεί λανθασμένα αρνητικές περιπτώσεις ως θετικές.

Η ανάκληση, γνωστή επίσης ως ευαισθησία (sensitivity) ή ποσοστό σωστών θετικών, μετρά την ικανότητα ενός μοντέλου να αναγνωρίζει σωστά τις θετικές περιπτώσεις ελαχιστοποιώντας τα ψευδή αρνητικά. Αξιολογεί το ποσοστό των αληθινών θετικών (true positives) που το μοντέλο καταφέρνει να ανιχνεύσει από όλες τις πραγματικές θετικές περιπτώσεις. Για τον υπολογισμό της ανάκλησης, λαμβάνουμε υπόψη τα ακόλουθα στοιχεία:

* Αληθώς Θετικά (TP): Ο αριθμός των περιπτώσεων που είναι πραγματικά θετικές και προβλέπονται σωστά ως θετικές από το μοντέλο.
* Ψευδώς Αρνητικά (FN): Ο αριθμός των περιπτώσεων που είναι πραγματικά θετικές, αλλά προβλέπονται λανθασμένα ως αρνητικές από το μοντέλο.

Η ανάκληση συνεπώς υπολογίζεται με τον ακόλουθο τύπο:

Ένα υψηλό ποσοστό ανάκλησης υποδεικνύει ότι το μοντέλο είναι αποτελεσματικό στο να ανιχνεύει τις περισσότερες θετικές περιπτώσεις, ελαχιστοποιώντας τον κίνδυνο των ψευδών αρνητικών. Με άλλα λόγια, το μοντέλο έχει χαμηλό ρυθμό αποτυχίας στην αναγνώριση των θετικών περιπτώσεων. Από την άλλη πλευρά, ένα χαμηλό ποσοστό ανάκλησης υποδεικνύει υψηλό αριθμό ψευδών αρνητικών, υποδηλώνοντας ότι το μοντέλο μπορεί να χάνει μια σημαντική ποσότητα θετικών περιπτώσεων.

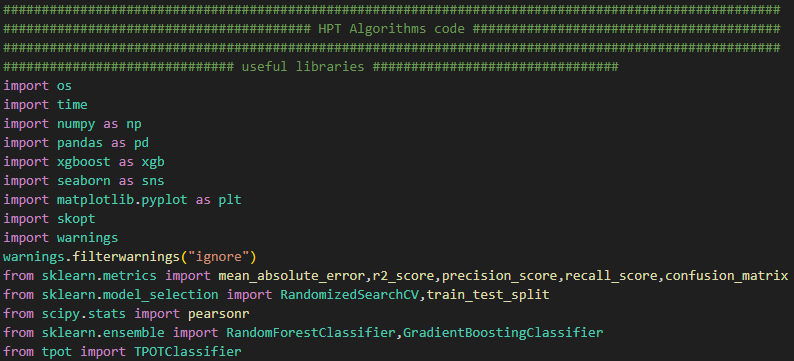
Η ορθότητα και η ανάκληση συχνά είναι αντίστροφα συνδεδεμένες, δημιουργώντας μια αντιστάθμιση μεταξύ των δύο μετρικών. Η βελτίωση της ακρίβειας μπορεί να οδηγήσει σε μείωση της ανάκλησης και αντίστροφα. Αυτή η αντιστάθμιση επηρεάζεται κυρίως από το όριο λήψης αποφάσεων του μοντέλου. Ρυθμίζοντας το όριο, μπορούμε να αλλάξουμε τον τρόπο με τον οποίο το μοντέλο προβλέπει θετικές και αρνητικές περιπτώσεις.

Συμπερασματικά, η ορθότητα και η ανάκληση είναι σημαντικές μετρικές αξιολόγησης για την απόδοση ενός μοντέλου μηχανικής μάθησης. Η ισορροπία μεταξύ των δύο μετρικών εξαρτάται από το πρόβλημα και τις απαιτήσεις του. Σε ορισμένες περιπτώσεις, η ορθότητα μπορεί να είναι πιο σημαντική, ενώ σε άλλες περιπτώσεις η ανάκληση μπορεί να έχει μεγαλύτερη σημασία. Είναι σημαντικό να εξετάζονται και οι δύο μετρικές για μια πλήρη αξιολόγηση της απόδοσης του μοντέλου.

# Μέθοδοι

Εδώ περιγράφονται ο κώδικας που χρησιμοποιήθηκε κατά το πειραματικό στάδιο, η ροή των εργασιών που εκτελεί, το αποτέλεσμα που παράγει και ο τρόπος με τον οποίο επιλέχτηκε να γίνει η ανάλυση των αποτελεσμάτων δηλαδή η σύγκριση των αλγορίθμων.

## Χρήσιμα πακέτα.



**Εικόνα 4.** Πακέτα που χρησιμοποιήθηκαν για την υλοποίηση του κώδικα.

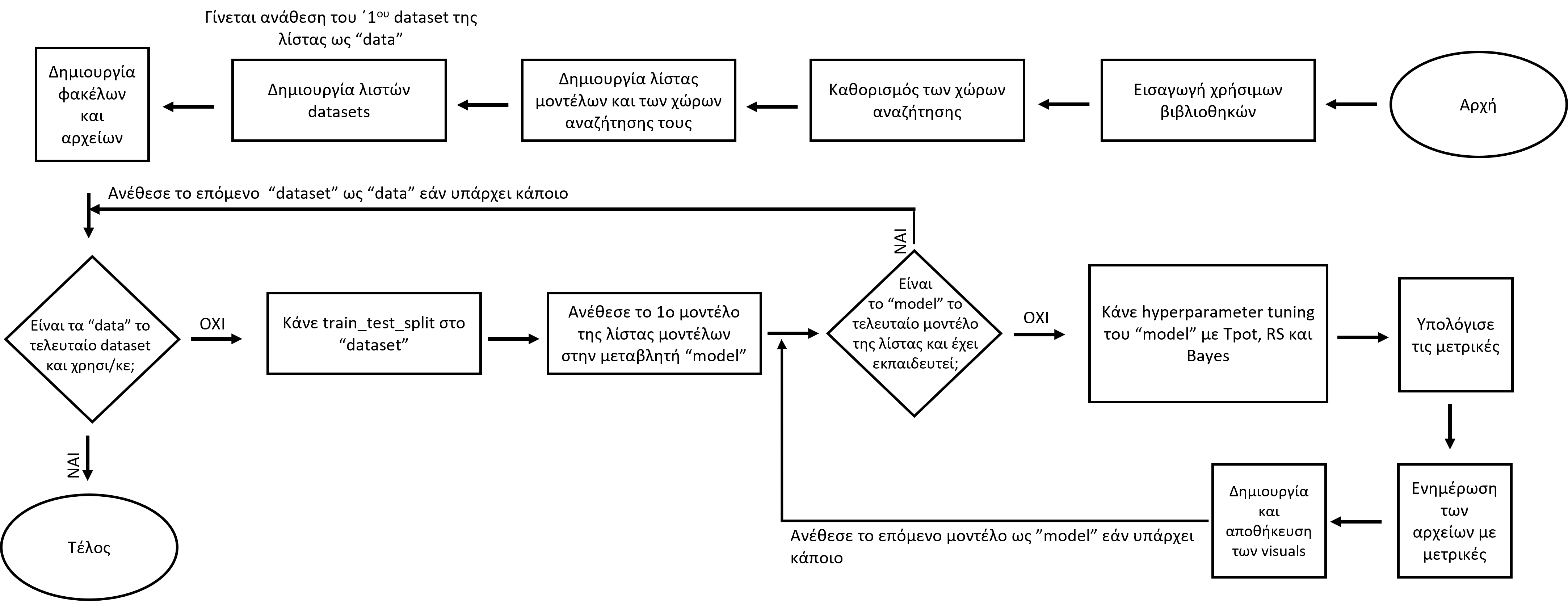
Για την υλοποίηση του κώδικα που εκτέλεσε την διαδικασία αναζήτησης βέλτιστων μοντέλων με τους αλγορίθμους TPOTClassifier, RandomizedSearchCV και BayesSearchCV για κάθε ένα από τα σύνολα δεδομένων χρησιμοποιήθηκε μία λίστα από πακέτα και λειτουργικές μονάδες (modules) συμβατά με την γλώσσα προγραμματισμού Python **Εικόνα 4**. Αυτά τα πακέτα και οι δυνατότητές τους περιγράφονται ως εξής:

* os: Στην Python, το os είναι μια λειτουργική μονάδα που παρέχει έναν τρόπο αλληλεπίδρασης με το λειτουργικό σύστημα. Δίνει την δυνατότητα να εκτελούνται διάφορες λειτουργίες που σχετίζονται με το χειρισμό αρχείων και φακέλων, τη διαχείριση διεργασιών, τις μεταβλητές περιβάλλοντος και άλλα. Η os λειτουργική μονάδα είναι μέρος της βιβλιοθήκης Python, επομένως δεν χρειάστηκε η εγκατάσταση κανενός εξωτερικού πακέτου για την χρησιμοποίηση της.
* time: Η time είναι μια άλλη λειτουργική μονάδα στην βιβλιοθήκη της Python η οποία παρέχει διάφορες συναρτήσεις για την εργασία πάνω σε διαδικασίες που σχετίζονται με το χρόνο, συμπεριλαμβανομένης της μέτρησης του χρόνου, της καθυστέρησης εκτέλεσης και της μορφοποίησης τιμών χρόνου.
* numpy: Η NumPy (Numerical Python) είναι μια ισχυρή βιβλιοθήκη στην Python για αριθμητικούς υπολογισμούς. Παρέχει υποστήριξη για μεγάλες, πολυδιάστατες συστοιχίες (arrays) και πίνακες (matrices), μαζί με μια συλλογή μαθηματικών συναρτήσεων για αποτελεσματική επέμβαση πάνω σε αυτούς τους πίνακες και επεξεργασία τους.
* pandas: Η Pandas είναι μία βιβλιοθήκη ανοιχτού κώδικα στην Python για την επεξεργασία και ανάλυση δεδομένων. Παρέχει εύκολες στη χρήση δομές δεδομένων (panda dataframe) και εργαλεία ανάλυσης δεδομένων για την αποτελεσματική εργασία με δομημένα δεδομένα, όπως πίνακες δεδομένων, χρονοσειρές και άλλα.
* xgboost: Η βιβλιοθήκη XGBoost είναι μια βιβλιοθήκη μηχανικής μάθησης ανοιχτού κώδικα που παρέχει αποτελεσματική υλοποίηση του αλγορίθμου XGBoost. Είναι γραμμένη σε C++ και προσφέρει διεπαφές για πολλές γλώσσες προγραμματισμού, συμπεριλαμβανομένων των Python, R, Java και Scala. Το πακέτο της Python xgboost είναι μια δημοφιλής επιλογή για τη χρήση του XGBoost αλγόριθμου στην Python.
* seaborn: H Seaborn είναι μια βιβλιοθήκη οπτικοποίησης δεδομένων στην Python χτισμένη πάνω στην Matplotlib. Παρέχει μια διεπαφή υψηλού επιπέδου για τη δημιουργία στατιστικών γραφημάτων. Η Seaborn έχει σχεδιαστεί ειδικά για εργασίες με δομημένα δεδομένα, όπως πλαίσια δεδομένων (dataframe) από την Pandas, και συνεργάζεται καλά με άλλες βιβλιοθήκες ανάλυσης δεδομένων όπως οι Pandas και NumPy. Στην συγκεκριμένη περίπτωση χρησιμοποιήθηκε για την δημιουργία πινάκων σύγχυσης.
* scipy: H scipy.stats είναι μια λειτουργική μονάδα της βιβλιοθήκης SciPy, η οποία χρησιμοποιείται από την επιστημονική κοινότητα με την γλώσσα προγραμματισμού Python. Το scipy.stats παρέχει ένα ευρύ φάσμα στατιστικών συναρτήσεων και κατανομών πιθανοτήτων χρήσιμες στην ανάλυση και διαχείριση δεδομένων.
* pyplot: Η Pyplot είναι μια λειτουργική μονάδα στη βιβλιοθήκη Matplotlib που παρέχει την δυνατότητα δημιουργίας διαφόρων τύπων γραφημάτων και απεικονίσεων στην Python. Είναι μια ευρέως χρησιμοποιούμενη βιβλιοθήκη σχεδίασης για οπτικοποίηση δεδομένων. Στα πλαίσια της εργασίας φάνηκε ιδιαίτερα χρήσιμη στην δημιουργία ραβδογραμμάτων και γραμμικών διαγραμμάτων.
* skopt: Η Scikit-Optimize παρέχει μεθόδους υλοποίησης των διαδικασιών βελτιστοποίησης υπερπαραμέτρων καθώς και βελτιστοποίησης μαύρου κουτιού (black-box optimization). Παρέχει ακόμη ένα σύνολο αλγορίθμων για τη βελτιστοποίηση των παραμέτρων διάφορων μοντέλων μηχανικής μάθησης και την επίλυση προβλημάτων βελτιστοποίησης σε διάφορους τομείς. Επίσης, παρέχει την δυνατότητα εύρεσης βέλτιστων υπερπαραμέτρων ενός μοντέλου μηχανικής μάθησης μέσο της συνάρτησης BayesSearchCV.
* warnings: Η λειτουργική μονάδα warnings στην Python παρέχει έναν τρόπο ελέγχου της εμφάνισης και του χειρισμού των προειδοποιήσεων που δημιουργούνται κατά την εκτέλεση του κώδικα. Επιτρέπει την προσαρμογή του τρόπου εμφάνισης των προειδοποιήσεων καθώς και την δυνατότητα φιλτραρίσματος τους.
* sklearn: H Scikit-learn είναι μια βιβλιοθήκη μηχανικής μάθησης στην Python η οποία παρέχει ένα ευρύ φάσμα εργαλείων και αλγορίθμων για διαδικασίες όπως οι ταξινόμηση, παλινδρόμηση, ομαδοποίηση, μείωση διαστάσεων και επιλογή κατάλληλου μοντέλου. Μέσο αυτής της βιβλιοθήκης μπορεί να γίνει χρήση των RandomForestClassifier, GradientBoostingClassifier και RandomizedSearchCV καθώς και συναρτήσεων για την αξιολόγηση και εκπαίδευση των μοντέλων μηχανικής μάθησης.
* tpot: Η TPOT είναι μια αυτοματοποιημένη βιβλιοθήκη μηχανικής μάθησης. Σχεδιάστηκε για να αυτοματοποιήσει τη διαδικασία κατασκευής και βελτιστοποίησης μοντέλων μηχανικής μάθησης για ένα σύνολο δεδομένων. Μέσο αυτής μπορεί να γίνει χρήση του TPOTClassifier.

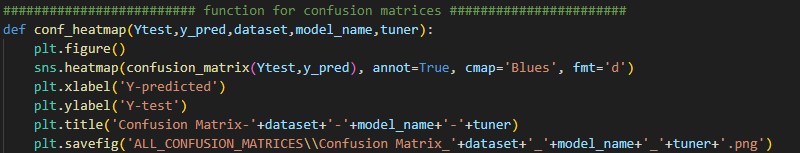
Όλα τα παραπάνω συνέβαλαν εξίσου στην ανάπτυξη του απαραίτητου για το πειραματικό στάδιο κώδικα μηχανικής μάθησης.

## Κυρίως κώδικας.

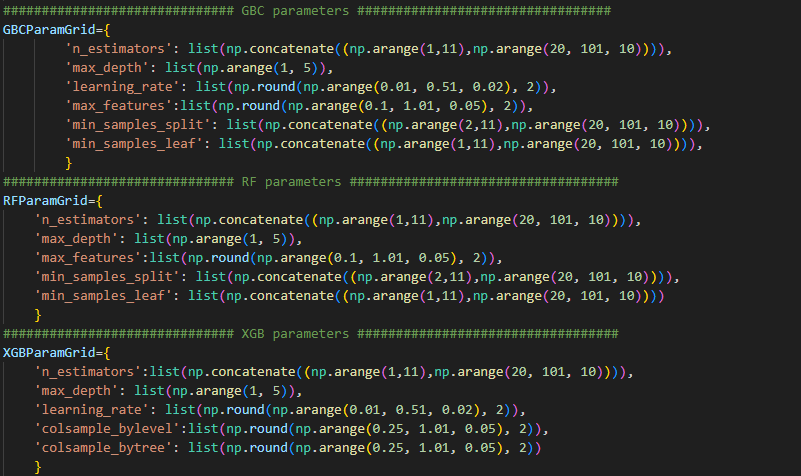
Αφού γίνει η φόρτωση των πακέτων πάνω στα οποία θα στηριχθεί η υλοποίηση του κώδικα, οι διαδικασίες που ακολουθούν περιγράφονται σχηματικά στο διάγραμμα ροής **Εικόνα 5**.



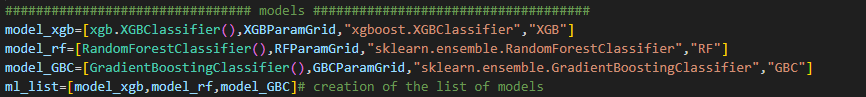
**Εικόνα 5.** Διάγραμμα ροής του κώδικα που παράχθηκε.

Πιο αναλυτικά, πρώτα γίνεται ο ορισμός μίας χρήσιμης συνάρτησης για την δημιουργία των πινάκων σύγχυσης **Εικόνα 6**. Αυτή δέχεται ως όρισμα τις προβλέψεις που έκανε ένα μοντέλο και τις πραγματικές τιμές που είχαν αυτές , το όνομα του συνόλου δεδομένων που χρησιμοποιήθηκε και το όνομα του μοντέλου και του αλγορίθμου βελτιστοποίησης που χρησιμοποιήθηκαν για την επίλυση του προβλήματος. Αυτή θα χρησιμοποιηθεί στο εσωτερικό του δεύτερου βρόγχου για κάθε βέλτιστο μοντέλο που προκύπτει.

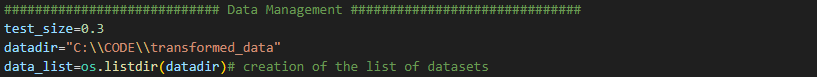
**Εικόνα 6.** Ορισμός συνάρτησης για την δημιουργία πινάκων σύγχυσης.

Μετά από αυτό, γίνεται ο ορισμός των χώρων αναζήτησης **Εικόνα 7** όπου για κάθε μοντέλο δημιουργείται ένα λεξικό με λέξεις κλειδιά τα ονόματα των υπερπαραμέτρων του μοντέλου και για κάθε λέξη κλειδί μία λίστα με ένα σύνολο δυνατών τιμών για αυτή. Για τoν XGBoostClassifier χρησιμοποιήθηκαν οι υπερπαράμετροι 'n\_estimators', 'max\_depth', 'learning\_rate', 'colsample\_bylevel' και 'colsample\_bytree' οι οποίες δέχονται ως τιμές ακέραιους ή πραγματικούς αριθμούς και κρίθηκαν ως οι σημαντικότερες για το συγκεκριμένο μοντέλο. Με αντίστοιχα κριτήρια για τον RandomForestClassifier επιλέχθηκαν οι 'n\_estimators', 'max\_depth', 'max\_features', 'min\_samples\_split' και 'min\_samples\_leaf' και για τον GradientBoostingClassifier έχουμε τις 'n\_estimators', 'max\_depth', 'learning\_rate', 'max\_features', 'min\_samples\_split' και 'min\_samples\_leaf'. Το τι ρυθμίζουν αυτές οι υπερπαράμετροι στο κάθε μοντέλο δίνεται στο II.B.1). Η επιλογή αυτών των χώρων αναζήτησης έγινε μελετώντας την δοσμένη βιβλιογραφία και μελετώντας τον κώδικα στο (Bakas, et al., 2023) όπου ορίζονταν αντίστοιχες λίστες τιμών.

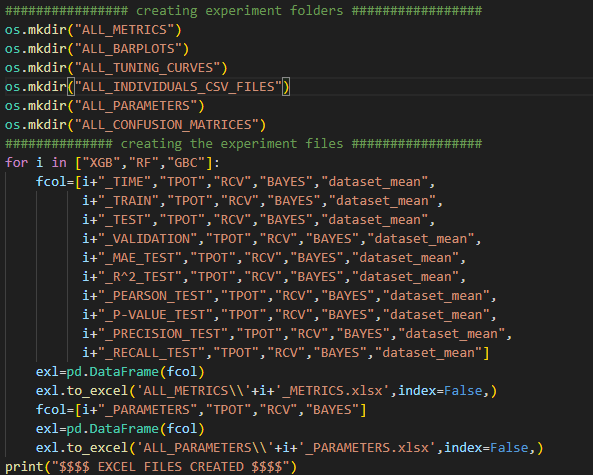
**Εικόνα 7.** Ορισμός χώρων αναζήτησης υπερπαραμέτρων.

Στην συνέχεια, γίνεται ο ορισμός της λίστας των μοντέλων μηχανικής μάθησης βάσει των οποίων θα γίνει η αναζήτηση των βέλτιστων μοντέλων κάνοντας χρήση των αλγορίθμων εύρεσης βέλτιστων υπερπαραμέτρων **Εικόνα 8**. Ο τρόπος με τον οποίο ορίστηκε αυτή η λίστα είναι βοηθητικός ως προς την εκτέλεση των εντολών που βρίσκονται εντός των βρόγχων επανάληψης του κώδικα. Κάθε αντικείμενο της λίστας μοντέλων είναι μία λίστα που περιέχει το όρισμα του μοντέλου κάνοντας χρήση της αντίστοιχης βιβλιοθήκης, τον χώρο αναζήτησης των υπερπαραμέτρων του σε μορφή λεξικού, την ακριβή τοποθεσία της μεθόδου που υλοποιεί το μοντέλο σε μορφή συμβολοσειράς (string) η οποία θα χρησιμοποιηθεί στον TPOTClassifier και το όνομά του ως μία συμβολοσειρά.

**Εικόνα 8.** Ορισμός της λίστας μοντέλων.

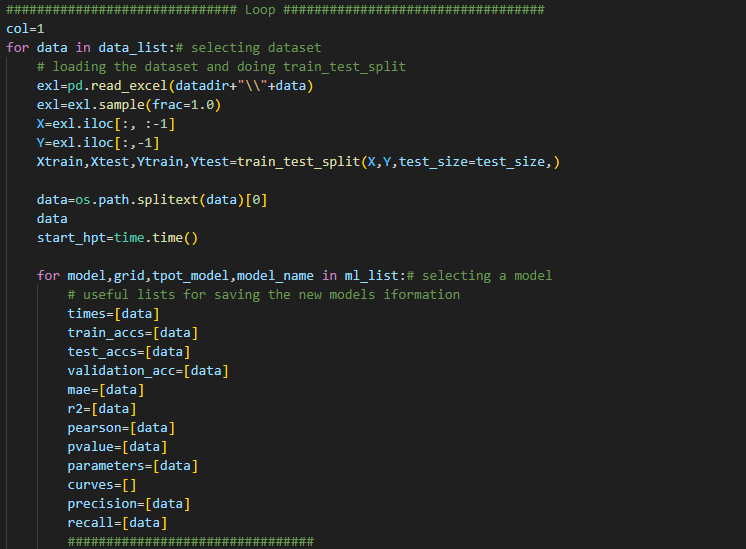
Ακολουθεί ο ορισμός του ποσοστού των δεδομένων που θα χρησιμοποιηθούν κατά την διαδικασία ελέγχου για τα βέλτιστα μοντέλα που θα προκύψουν καθώς ταυτόχρονα και το ποσοστό των δεδομένων πάνω στο οποίο θα γίνει η διαδικασία της εκπαίδευσης. Το ποσοστό αυτό επιλέχτηκε να είναι το 30% που είναι μία αρκετά συνήθης επιλογή κατά την εκπόνηση εργασιών μηχανικής μάθησης. Ακόμα, γίνεται ορισμός της λίστας δεδομένων που περιέχει όλα τα ονόματα των αρχείων των συνόλων δεδομένων που θα χρησιμοποιηθούν για την εκπαίδευση των μοντέλων. Αυτά θα φορτωθούν από τον φάκελο “transformed\_data” όπου έχουν αποθηκευτεί σε μορφή αρχείων .xlsx κατόπιν σχετικής επεξεργασίας. Όλα αυτά συμβαίνουν στην **Εικόνα 9**.

**Εικόνα 9**. Ορισμός ποσοστού δεδομένων ελέγχου και λίστας συνόλων δεδομένων.

Για την πιο εύκολη και οργανωμένη πρόσβαση στα πειραματικά αποτελέσματα αλλά και για την εύκολη διαχείριση τους στον κώδικα δημιουργείται πρώτα μία σειρά από φακέλους και μέσα σε ορισμένους από αυτούς μία σειρά από αρχεία μορφής .csv και .xlsx στα οποία γίνεται και η αποθήκευση των πειραματικών αποτελεσμάτων **Εικόνα 10**.

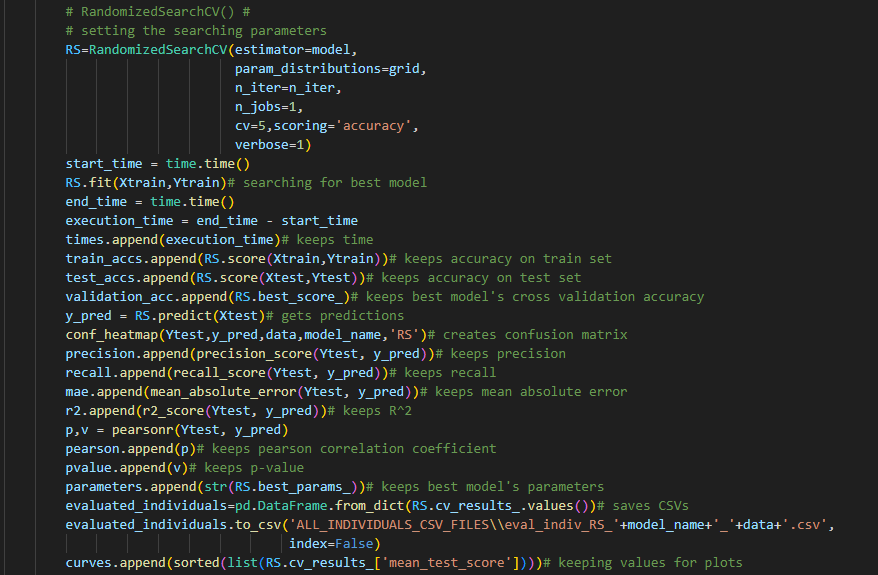
**Εικόνα 10**. Δημιουργία φακέλων και αρχείων.

Μετά την εκτέλεση όλων των παραπάνω βημάτων ο κώδικας περνάει στην εφαρμογή μίας επαναληπτικής διαδικασίας που αποτελείται από δύο φωλιασμένους βρόχους **Εικόνα 11**. Ο εξωτερικός βρόχος είναι υπεύθυνος για την επιλογή του συνόλου δεδομένων πάνω στο οποίο θα εφαρμοστούν τα μοντέλα μηχανικής μάθησης. Ποιο αναλυτικά, γίνεται η επιλογή ενός ονόματος από την λίστα δεδομένων που δημιουργήθηκε πριν, το οποίο στην συνέχεια χρησιμοποιείται για την φόρτωση των δεδομένων από το αντίστοιχο αρχείο. Τα δεδομένα που φορτώνονται ανακατεύονται με τυχαίο τρόπο και αφού χωριστούν τα δεδομένα στόχου από τα υπόλοιπα γίνεται η επιλογή του ποσοστού των δεδομένων ελέγχου που ορίστηκε πιο πάνω με την εφαρμογή της συνάρτησης “train\_test\_split” που επιστρέφει τα σύνολα δεδομένων εκπαίδευσης, ελέγχου και τα σύνολα των πραγματικών τιμών της μεταβλητής στόχου για τα αντίστοιχα σύνολα δεδομένων εκπαίδευσης και ελέγχου.

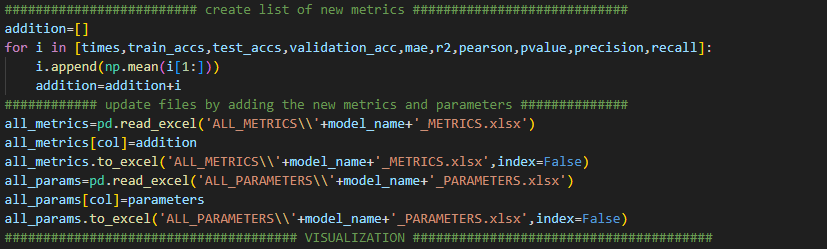
Ο δεύτερος βρόχος είναι υπεύθυνος για την επιλογή του μοντέλου για το οποίο θα εφαρμοστούν οι μέθοδοι εύρεσης βέλτιστων υπερπαραμέτρων στο σύνολο δεδομένων που επιλέχτηκε **Εικόνα 11**. Εντός αυτού του βρόχου εφαρμόζονται οι μέθοδοι εύρεσης βέλτιστων υπερπαραμέτρων, η αποθήκευση των αποτελεσμάτων και η οπτικοποίησή τους με την σειρά που αναφέρθηκαν, πριν όμως γίνουν αυτά ορίζεται ένα πλήθος βοηθητικών λιστών για την αποθήκευση των αποτελεσμάτων κάθε μεθόδου εύρεσης βέλτιστου μοντέλου, οι οποίες στο τέλος του εσωτερικού βρόχου συνενώνονται και αποθηκεύονται στο αντίστοιχο αρχείο.

**Εικόνα 11.** Φωλιασμένοι βρόχοι επανάληψης.

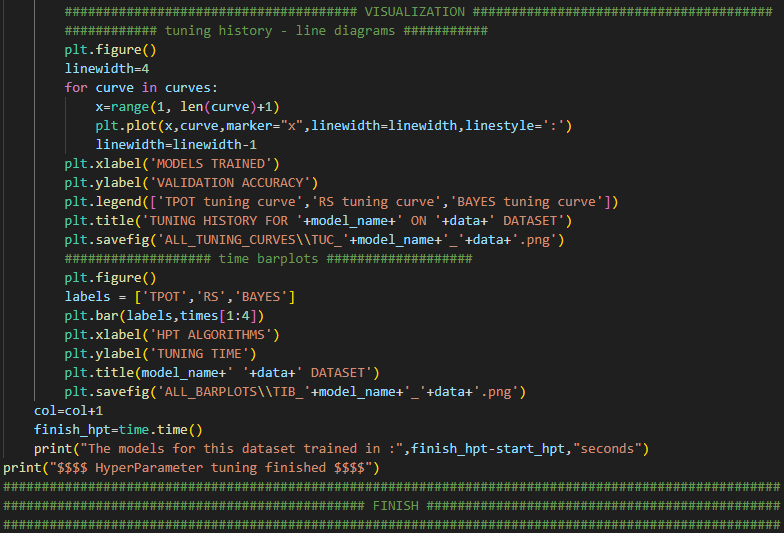
Για κάθε μέθοδο εύρεσης βέλτιστων υπερπαραμέτρων εκτελείται ένα σύνολο βημάτων για την αποθήκευση των κατάλληλων μετρικών και άλλων πληροφοριών βάσει των οποίων θα γίνει η σύγκριση των επιδόσεων των αλγορίθμων. Στην **Εικόνα 12** γίνεται λόγος για τα βήματα που εκτελούνται για την περίπτωση του RandomizedSearchCV τα οποία εκτελούνται κατά παρόμοιο τρόπο και για τους άλλους δύο αλγορίθμους. Πρώτα, ορίζονται οι παράμετροι του αλγορίθμου αναζήτησης, δηλαδή δίνονται τιμές για τις παραμέτρους των αλγορίθμων που ειπώθηκαν στην παράγραφο II.B.1).Εδώ είναι σημαντικό να αναφερθούν οι τιμές των παραμέτρων για κάθε αλγόριθμο εύρεσης βέλτιστου μοντέλου. Για τον TPOTClassifier δόθηκαν η τιμή 5 στην παράμετρο generations, 30 στην population\_size, 15 στην offspring\_size και 2 στην verbosity. Για τον BayesSearchCV δίνεται η τιμή του πλήθους των μοντέλων που εκπαιδεύονται από τον TPOTClassifier στην παράμετρο n\_iter και το ίδιο συμβαίνει και για τον RandomizedSearchCV. Για την μεταβλητή cv δίνεται σε όλους η τιμή 5 που κρίθηκε η καταλληλότερη και για τα σύνολα δεδομένων αλλά και ως κοινή πρακτική κατά την εκπαίδευση μοντέλων κάνοντας χρήση διασταυρούμενης επικύρωσης. Επίσης για την scoring δόθηκε η συμβολοσειρά “accuracy” και στους τρεις ενώ σε όλους δόθηκε και η τιμή 1 για την μεταβλητή n\_jobs.Τέλος, στις παραμέτρους που αφορούν τους χώρους αναζήτησης των υπερπαραμέτρων δίνεται σε όλα το λεξικό που αφορά το αντίστοιχο μοντέλο που χρησιμοποιείται κατά την αναζήτηση των βέλτιστων μοντέλων.

Επιστρέφοντας στην **Εικόνα 12**, μετά τον ορισμό των παραμέτρων ακολουθούν βήματα για προσαρμογή στα δεδομένα και εξαγωγή αποτελεσμάτων. Ποιο συγκεκριμένα γίνεται η διαδικασία αναζήτησης του βέλτιστου μοντέλου η οποία χρονομετρείται. Έπειτα, γίνεται ο υπολογισμός της ακρίβειας του βέλτιστου μοντέλου στα δεδομένα εκπαίδευσης και ελέγχου ενώ αποθηκεύεται και η ακρίβεια που προέκυψε από την διαδικασία διασταυρούμενης επικύρωσης. Στην συνέχεια το μοντέλο χρησιμοποιείται για να προβλέψει την κλάση στόχο για τα δεδομένα ελέγχου, οι οποίες προβλέψεις χρησιμοποιούνται για να υπολογιστεί ο πίνακας σύγχυσης, η ορθότητα (precision), η ανάκληση, το μέσο απόλυτο σφάλμα, το R2 , ο συντελεστής συσχέτισης του Pearson και το p-value. Κατόπιν, αποθηκεύονται οι παράμετροι του μοντέλου και δημιουργούνται αρχεία μορφής .csv τα οποία περιέχουν πληροφορίες για τα μοντέλα μέσα από τα οποία επιλέχτηκε το βέλτιστο για μία ολοκληρωμένη εικόνα της διαδικασίας που εκτελέστηκε. Κατά την εκτέλεση αυτών των βημάτων οι χρήσιμες πληροφορίες αποθηκεύονται σε προσωρινές λίστες που αναφέρθηκαν πιο πάνω.

**Εικόνα 12**. Βήματα εξαγωγής πειραματικών αποτελεσμάτων.

Έπεται η αποθήκευση των πληροφορίων για τα μοντέλα που προέκυψαν στα αντίστοιχα αρχεία **Εικόνα 13**. Αυτό γίνεται συνενώνοντας τις προσωρινές λίστες και χρησιμοποιώντας την λίστα που προκύπτει για την ανανέωση των αντίστοιχων αρχείων φορτώνοντάς τα και προσθέτοντας ως στήλη την λίστα με τα νέα δεδομένα.

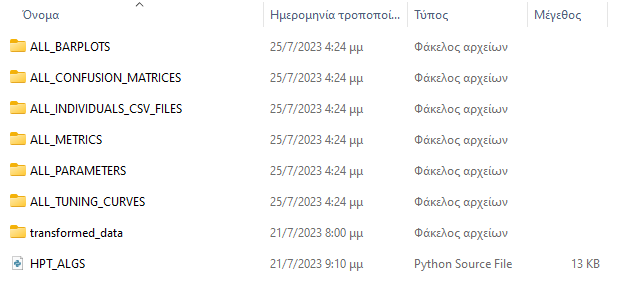
**Εικόνα 13**. Ανανέωση αρχείων.

Τέλος, γίνεται η δημιουργία ορισμένων χρήσιμων μέσων οπτικοποίησης **Εικόνα 14**. Αυτά είναι ένα σύνολο από γραμμικά διαγράμματα που σκοπό έχουν να περιγράψουν την διαδικασία εύρεσης των βέλτιστων υπερπαραμέτρων για ένα μοντέλο κάθε φορά από τις τρεις μεθόδους αναζήτησης και ένα σύνολο ραβδογραμμάτων με σκοπό να γίνει σύγκριση των χρονικών δεδομένων που αντλήθηκαν κατά τις διαδικασίες αναζήτησης που εκτελέστηκαν.

**Εικόνα 14.** Δημιουργία μέσων οπτικοποίησης.

## Παραγόμενα αποτελέσματα.

Ο κώδικας του οποίου η περιγραφή δόθηκε στην παράγραφο III.B παράγει όπως αναφέρθηκε ένα σύνολο αρχείων και φακέλων στο working directory που έγινε η εκτέλεση του. Εδώ θα δοθεί μία σύντομή περιγραφή των περιεχομένων των φακέλων αυτών **Εικόνα 15**. Στον φάκελο “ALL\_BARPLOTS” βρίσκονται 9 ραβδογράμματα για την σύγκριση του χρόνου εκτέλεσης των αλγορίθμων εύρεσης βέλτιστων υπερπαραμέτρων. Κάθε τέτοιο ραβδόγραμμα απεικονίζει το χρονικό διάστημα αναζήτησης υπερπαραμέτρων για κάθε αλγόριθμο αναζήτησης για ένα μοντέλο μηχανικής μάθησης που εφαρμόστηκε σε ένα σύνολο δεδομένων. Ο φάκελος “ALL\_CONFUSION\_MATRICES” περιέχει 27 πίνακες σύγχυσης, έναν για κάθε βέλτιστο μοντέλο που εκπαιδεύτηκε. Αυτοί οι πίνακες είναι χρήσιμοι για την ανάλυση της επίδοσης των μοντέλων που προέκυψαν. Ο φάκελος “ALL\_INDIVDUAL\_CSV\_FILES” περιέχει 27 αρχεία csv που περιέχουν πληροφορίες για όλες τις διαδικασίες αναζήτησης που εκτελέστηκαν κατά την ροή του κώδικα. Ο φάκελος “ALL\_METRICS” περιέχει χρήσιμα για την σύγκριση απόδοσης των μοντέλων αρχεία τύπου xlsx στα οποία αποθηκεύονται οι μετρικές που υπολογίστηκαν. Ο φάκελος “ALL\_PARAMETERS” περιέχει αρχεία xlsx με πληροφορίες για τις υπερπαραμέτρους των βέλτιστων μοντέλων που βρέθηκαν και τέλος ο φάκελος “ALL\_TUNING\_CURVES” περιέχει ένα σύνολο από γραμμικά διαγράμματα τα οποία σχηματίζονται τοποθετώντας κατά αύξουσα σειρά τις τιμές ακρίβειας της διασταυρούμενης επικύρωσης των μοντέλων που συγκρίνονται κατά την διαδικασία αναζήτησης για την εύρεση του βέλτιστου μοντέλου. Έτσι για κάθε εκτέλεση ενός αλγορίθμου αναζήτησης προκύπτει μία καμπύλη συντονισμού. Συνολικά δημιουργήθηκαν 27 τέτοιες καμπύλες οι οποίες για λόγους εύκολης σύγκρισης παρουσιάζονται στα διαγράμματα ανά τριάδες όπου κάθε τριάδα αφορά το ίδιο μοντέλο μηχανικής μάθησης και το ίδιο σύνολο δεδομένων.



**Εικόνα 15.** Φάκελοι πειραματικών αποτελεσμάτων.

## Υπολογιστικό σύστημα.

Σε αυτή την εργασία μηχανικής μάθησης, έγινε χρήση ενός υπολογιστή εξοπλισμένου με επεξεργαστή Intel(R) Core(TM) i5 M540, ο οποίος λειτουργεί με βασική ταχύτητα επεξεργαστή 2,53 GHz. Αυτός ο επεξεργαστής διπλού πυρήνα, με τέσσερα νήματα που ενεργοποιούνται από την τεχνολογία Hyper-Threading, προσφέρει μια ισορροπημένη απόδοση κατάλληλη για το χειρισμό διαφόρων εργασιών σχετικών με την εκπαίδευση μοντέλων και την ανάλυση δεδομένων. Η αρχιτεκτονική 64-bit του επεξεργαστή διασφαλίζει ότι μπορεί να γίνει επεξεργασία και αποτελεσματικός χειρισμός σχετικά μεγάλων συνόλων δεδομένων, επιτρέποντας την χρήση εξελιγμένων και πολύπλοκων μοντέλων.

Οι αλγόριθμοι μηχανικής μάθησης που χρησιμοποιούνται σε αυτή την εργασία επωφελούνται από τις βιβλιοθήκες Pandas και NumPy, αξιοποιώντας πλήρως την ιεραρχία της κρυφής μνήμης της CPU για αποτελεσματικό χειρισμό δεδομένων και υπολογισμούς. Οι κρυφές μνήμες L1 και L2 του επεξεργαστή Intel Core i5 M540 παρέχουν γρήγορη πρόσβαση σε δεδομένα με συχνή πρόσβαση, ενώ η κρυφή μνήμη L3 των 3 MB υποστηρίζει μεγαλύτερα σύνολα δεδομένων και λειτουργίες.

Η βιβλιοθήκη XGBoost αξιοποιεί τις δυνατότητες πολλαπλών νημάτων του επεξεργαστή, επιτρέποντας ταχύτερη εκπαίδευση μοντέλων και συντονισμό υπερπαραμέτρων, καθοριστικής σημασίας για την επίτευξη υψηλότερης ακρίβειας πρoβλέψεων. Επιπλέον, χρησιμοποιήσαμε τη βιβλιοθήκη TPOT, χρησιμοποιώντας τις δυνατότητες της αυτοματοποιημένης μηχανικής μάθησης για την αποτελεσματική αναζήτηση σε ένα ευρύ φάσμα υπερπαραμέτρων των μοντέλων μηχανικής μάθησης που χρησιμοποιήθηκαν. Ο γενετικός προγραμματισμός που χρησιμοποιείται στο TPOT βελτιώνει περαιτέρω τη διαδικασία επιλογής μοντέλων, επιτρέποντάς την εξερεύνηση διάφορων μοντέλων μηχανικής μάθησης χωρίς κόπο.

Επιπλέον, οι βιβλιοθήκες Seaborn και Matplotlib χρησιμοποιήθηκαν για την οπτικοποίηση και ανάλυση των αποτελεσμάτων των μοντέλων μηχανικής εκμάθησης. Οι πλούσιες δυνατότητες οπτικοποίησης που προσφέρουν αυτές οι βιβλιοθήκες μας βοήθησαν να αποκτήσουμε γνώσεις για την απόδοση διαφορετικών μοντέλων, επιτρέποντάς μας να λαμβάνουμε αποφάσεις βάσει δεδομένων και να τελειοποιούμε περαιτέρω τα μοντέλα κάτι το οποίο ήταν εφικτό μέσο της χρήσης αυτού του συστήματος.

Συνολικά, ο συνδυασμός του επεξεργαστή Intel Core i5 M540, της άφθονης μνήμης και των ισχυρών βιβλιοθηκών μηχανικής εκμάθησης αποδείχθηκε σημαντικό πλεονέκτημα σε αυτή την εργασία. Η αποτελεσματικότητα του συστήματός που χρησιμοποιήθηκε έκανε επιτρεπτή την πραγματοποίηση εκτενούς συντονισμού υπερπαραμέτρων και την αξιολόγηση μοντέλων μηχανικής μάθησης, διασφαλίζοντας με σιγουριά την δυνατότητα να εξαγάγουμε τη γραμμή μηχανικής μάθησης με την καλύτερη απόδοση για τα συγκεκριμένα σύνολα δεδομένων και τον τομέα προβλημάτων που αντιμετωπίστηκαν. Ως αποτέλεσμα, έγινε επίτευξη υψηλού επιπέδου ακρίβειας πρόβλεψης και βεβαιώθηκε η αξιοπιστία των αποτελεσμάτων για την διεκπεραίωση της ζητούμενης σύγκρισης των αλγορίθμων αναζήτησης.

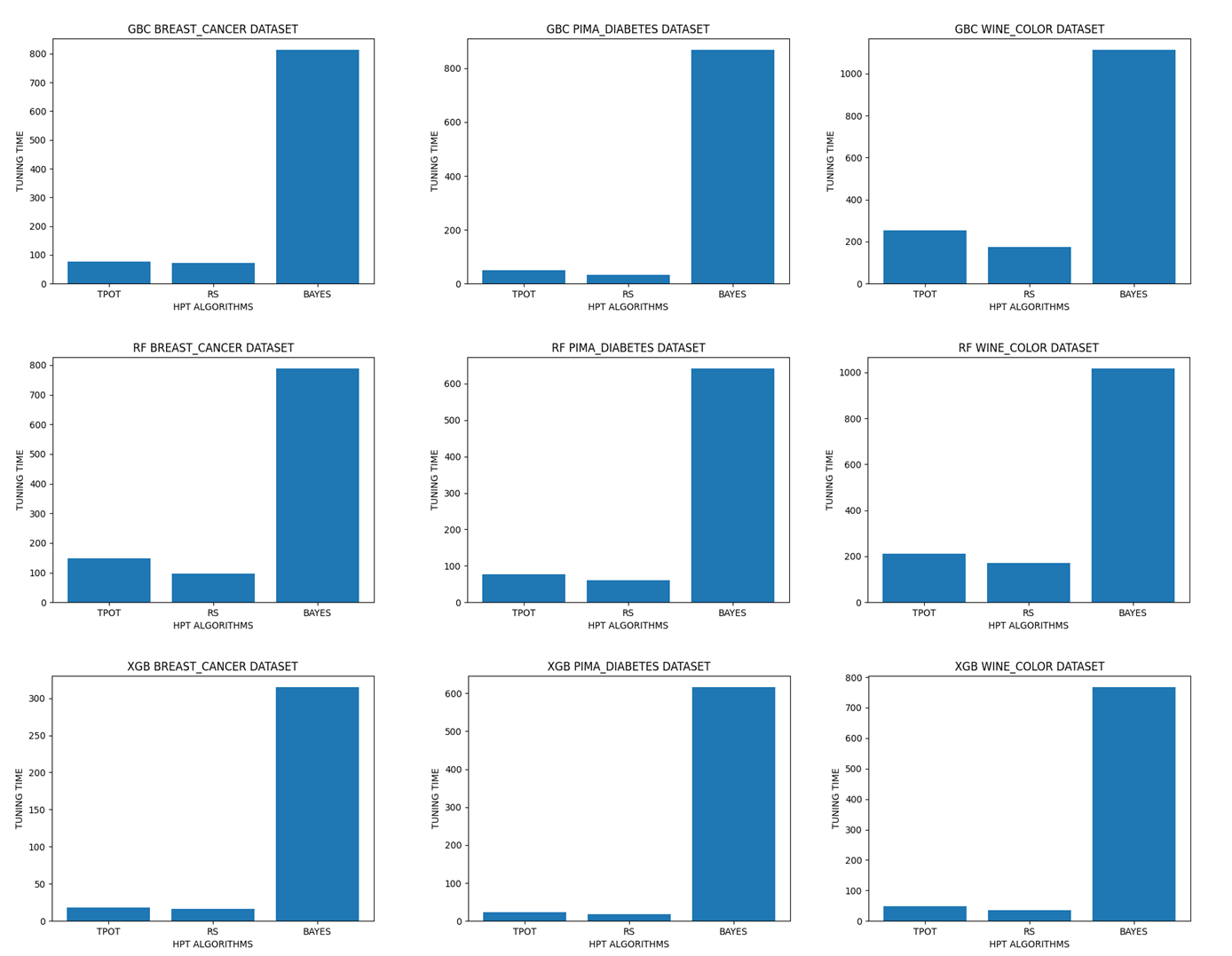
# Πειραματικά αποτελέσματα

Εδώ θα γίνει η ανάλυση των πειραματικών αποτελεσμάτων δηλαδή η σύγκριση των αλγορίθμων αναζήτησης βέλτιστων υπερπαραμέτρων που χρησιμοποιήθηκαν. Αυτό θα γίνει σε τρία στάδια. Πρώτα θα γίνει η ανάλυση της ταχύτητας των αλγορίθμων ερμηνεύοντας τα ραβδογράμματα. Έπειτα, θα γίνει η ανάλυση των γραμμικών διαγραμμάτων με σκοπό να διερευνηθεί η συνέπεια όσον αφορά τις επιδόσεις των αλγορίθμων δηλαδή την δυνατότητα τους να εντοπίζουν μοντέλα με καλές επιδόσεις σε όλη την διάρκεια αναζήτησης. Τέλος θα γίνει η σύγκριση των βέλτιστων μοντέλων που βρέθηκαν ξεχωριστά για κάθε σύνολο δεδομένων και κάθε μοντέλο ώστε να γίνουν φανερές τυχόν διαφορές μεταξύ των αλγορίθμων αναζήτησης ως προς τις επιδόσεις των μοντέλων που παράγουν.

Πριν όμως γίνει η ανάλυση όλων αυτών πρέπει να γίνει ένας έλεγχος για τις υπερπαραμέτρους που βρέθηκαν για τα βέλτιστα μοντέλα που εκπαιδεύτηκαν. Αυτό είναι σημαντικό, καθώς στην περίπτωση που για ένα μοντέλο μηχανικής μάθησης έχουν βρεθεί από δύο ή και τους τρείς αλγορίθμους αναζήτησης ίδιες τιμές για τις υπερπαραμέτρους και οι μετρικές απόδοσης των μοντέλων αυτών διαφέρουν τότε προφανώς αυτό οφείλεται σε κάποια τυχαιότητα κατά την εκπαίδευση των μοντέλων και έτσι δεν έχει νόημα η ερμηνεία των μετρικών για την σύγκριση τους μιας και οι αλγόριθμοι αναζήτησης συμφωνούν στο βέλτιστο μοντέλο. Σε αντίθετη περίπτωση, όταν δηλαδή οι υπερπαράμετροι που βρέθηκαν διαφέρουν για τα μοντέλα τότε υπάρχει λόγος σύγκρισης των μοντέλων με σκοπό την σύγκριση της απόδοσης των αλγορίθμων αναζήτησης. Έτσι, στην **Εικόνα 16** παρουσιάζονται οι βέλτιστοι υπερπαράμετροι που επιλέχτηκαν από τους τρείς αλγορίθμους εύρεσης βέλτιστων υπερπαραμέτρων κατά την διαδικασία της αναζήτησης. Παρατηρείται πως οι τριάδες μοντέλων που εκπαιδεύτηκαν για κάθε επιλογή αλγορίθμου μηχανικής μάθησης και σύνολο δεδομένων παρουσιάζουν όλες τους διαφορετικούς συνδυασμούς υπερπαραμέτρων δηλαδή οι αλγόριθμοι αναζήτησης έδωσαν όλοι διαφορετικά αποτελέσματα και έτσι υπάρχει σε κάθε περίπτωση λόγος εξέτασης των σχετικών μετρικών για την εξαγωγή επιπλέον συμπερασμάτων.

**Εικόνα 16**. Όλοι οι υπερπαράμετροι των 27 βέλτιστων μοντέλων που εκπαιδεύτηκαν.

## Ανάλυση ραβδογραμμάτων.

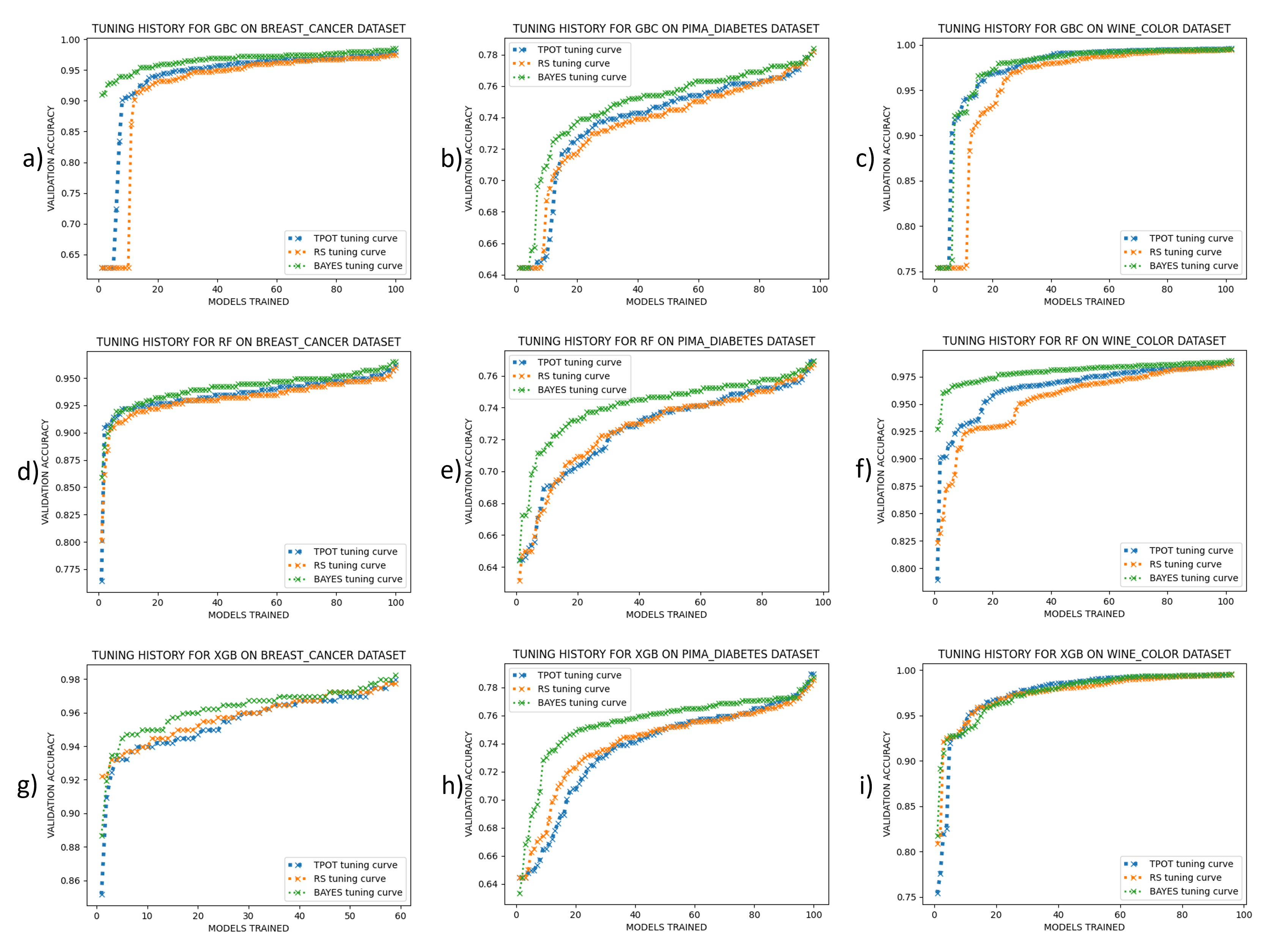
Από τους σημαντικότερους παράγοντες κατά την σύγκριση δύο αλγορίθμων είναι ο υπολογιστικός τους χρόνος. Στην **Εικόνα 17** παρουσιάζονται τα 9 ραβδογράμματα που κάθε ένα περιγράφει τους υπολογιστικούς χρόνους των αλγορίθμων αναζήτησης κατά την ρύθμιση των υπερπαραμέτρων πάνω σε ένα μοντέλο μηχανικής μάθησης το οποίο εκπαιδεύτηκε σε ένα σύνολο δεδομένων. Είναι εύκολο να παρατηρήσει κανείς πως επιλέγοντας ένα μοντέλο μηχανικής μάθησης και ένα σύνολο δεδομένων προκύπτουν τρία βέλτιστα μοντέλα, ένα από κάθε αλγόριθμο αναζήτησης βέλτιστων υπερπαραμέτρων. Είναι προφανές πως σε κάθε ένα από τα ραβδογράμματα τηρείται μία ιεραρχία όσον αφορά τον χρόνο εκτέλεσης της αναζήτησης με τον πιο γρήγορο αλγόριθμο να είναι ο RandomizedSearchCV στην συνέχεια ακολουθεί πάντα ο TPOTClassifier που παρουσιάζει χρόνο εκτέλεσης συνήθως λίγο μεγαλύτερο της τυχαίας αναζήτησης με εξαίρεση την περίπτωση του μοντέλου τυχαίων δασών για το σύνολο δεδομένων του καρκίνου του μαστού όπου παρουσιάζει σχεδόν διπλάσιο χρόνο εκτέλεσης σε σχέση με τον RandomizedSearchCV και τέλος ακολουθεί ο BayesSearchCV ο οποίος είναι σαφώς πολύ πιο αργός στην εκτέλεσή του αφού φαίνεται να παρουσιάζει έως και σχεδόν 8 φορές μεγαλύτερο χρόνο εκτέλεσης σε κάποιες περιπτώσεις. Αυτή η διαφορά του BayesSearchCV έναντι των άλλων αλγορίθμων οφείλεται στην πολυπλοκότητα με την οποία εκτελεί την διαδικασία αναζήτησης του βέλτιστων υπερπαραμέτρων.

**Εικόνα 17**. Ραβδογράμματα υπολογιστικού χρόνου.

Η αναζήτηση βέλτιστων υπερπαραμέτρων με τον BayesSearchCV χρησιμοποιεί ένα πιθανοθεωρητικό μοντέλο για την διαδικασία αναζήτησης του βέλτιστου μοντέλου. Σε κάθε βήμα του αλγορίθμου αυτού βρίσκεται ένα μοντέλο μηχανικής μάθησης το οποίο στο επόμενο βήμα του αντικαθίσταται από ένα καλύτερο καθώς αξιολογούνται νέοι συνδυασμοί υπερπαραμέτρων. Αυτό δίνει την δυνατότητα λήψης πιο ενημερωμένων αποφάσεων σχετικά με το ποιοι συνδυασμοί θα δημιουργηθούν και δοκιμαστούν σε κάθε βήμα βάσει των πληροφοριών που αντλήθηκαν από προηγούμενες αξιολογήσεις. Έτσι αυτός ο αλγόριθμος μπορεί να απαιτεί πρόσθετους υπολογιστικούς πόρους για την προσαρμογή του μοντέλου και τις ενημερώσεις σε σχέση με τον RandomizedSearchCV που επιλέγει τυχαίους συνδυασμούς υπερπαραμέτρων. Βέβαια, ο αλγόριθμος TPOTClassifier παρουσιάζεται ως μεγαλύτερης πολυπλοκότητας αλγόριθμος στην βιβλιογραφία. Αυτό συμβαίνει καθώς ο TPOTClassifier εκτός της δυνατότητάς του να εκτελεί εύρεση βέλτιστων υπερπαραμέτρων με χρήση γενετικών αλγορίθμων προσφέρει επίσης την δυνατότητα αναζήτησης του καλύτερου αγωγού (pipeline) που ουσιαστικά είναι μία σειρά από βήματα για την διαχείριση των δεδομένων και την κατασκευή του βέλτιστου μοντέλου πρόβλεψης για αυτά. Μία τέτοιου είδους αναζήτηση θα απαιτούσε σαφώς μεγαλύτερο χρόνο εκτέλεσης από αυτόν που βλέπουμε στα ραβδογράμματα όμως δίνοντας τις κατάλληλες τιμές στις παραμέτρους ‘template’ και ‘config\_dict’ του TPOTClassifier ορίσαμε πως θα εκτελεί αναζήτηση βέλτιστων υπερπαραμέτρων μόνο για τα ζητούμενα μοντέλα μειώνοντας έτσι τον αναμενόμενο χρόνο εκτέλεσης σημαντικά. Έτσι στα πλαίσια της εργασίας ο RandomizedSearchCV είναι ο ταχύτερος, ακολουθεί ο TPOTClassifier και ο BayesSearchCV είναι ο πιο αργός.

Υπάρχουν βέβαια και μερικές άλλες σημαντικές παρατηρήσεις σχετικά με την **Εικόνα 17**. Αρχικά, εξετάζοντας τα ραβδογράμματα κατά τα μοντέλα μηχανικής μάθησης, παρατηρούμε πως για τα μοντέλα XGBoost η εύρεση των υπερπαραμέτρων γίνεται εμφανώς πιο γρήγορα από τα άλλα μοντέλα για κάθε αλγόριθμο αναζήτησης σε κάθε σύνολο δεδομένων. Μάλιστα για το σύνολο που αφορά την διάγνωση του καρκίνου του μαστού γίνεται έως και τρεις φορές πιο γρήγορα. Τα μοντέλα gradient boosting είναι γενικά τα πιο αργά στην διαδικασία εύρεσης των βέλτιστων υπερπαραμέτρων τους ενώ για τα μοντέλα τυχαίων δασών η διαδικασία αργεί για το σύνολο δεδομένων που αφορά την διάγνωση του καρκίνου του μαστού σχετικά με τα άλλα μοντέλα ενώ στα άλλα δύο σύνολα δεδομένων η διαδικασία εκτελείται πιο γρήγορα από ότι για τα αντίστοιχα μοντέλα gradient boosting. Επίσης, εξετάζοντας ως προς τα σύνολα δεδομένων, μπορούμε να πούμε πως η διαδικασία εύρεσης των βέλτιστων υπερπαραμέτρων ήταν γενικά πιο αργή για το σύνολο δεδομένων του χρώματος του κρασιού κάτι που πιθανότατα να οφείλεται στο πλήθος των δειγμάτων αυτού του συνόλου που ήταν σαφώς μεγαλύτερο, σχετικά μικροί χρόνοι εκτέλεσης φαίνονται και για το σύνολο δεδομένων που αφορά την διάγνωση του διαβήτη όπου είχαμε περίπου ίδιο πλήθος δειγμάτων με το σύνολο δεδομένων πρόγνωσης του καρκίνου αλλά σαφώς λιγότερα χαρακτηριστικά για κάθε δείγμα. Τα συμπεράσματα αυτά μπορούν να εξαχθούν και μελετώντας τα τμήματα στα **Πίνακας 1**, **Πίνακας 2** **Πίνακας 3** που αφορούν τον χρόνο (GBC\_TIME,XGB\_TIME,RF\_TIME) εκτέλεσης των αλγορίθμων αναζήτησης.

## Ανάλυση γραμμικών γραφημάτων.

Σημαντική είναι επίσης και η εξέλιξη της σύγκλισης των αλγορίθμων αναζήτησης στο βέλτιστο μοντέλο. Όπως αναφέρθηκε, κατά την διαδικασία αναζήτησης βέλτιστων υπερπαραμέτρων δημιουργείται ένα πλήθος μοντέλων μηχανικής μάθησης τα οποία αξιολογούνται και από αυτά επιλέγεται το καλύτερο. Είναι σημαντικό να έχουμε εικόνα των επιδόσεων αυτών των μοντέλων για μία πιο αναλυτική σύγκριση των αλγορίθμων αναζήτησης βέλτιστων υπερπαραμέτρων. Έτσι, στην **Εικόνα 18** παρουσιάζονται γραφήματα με τις καμπύλες ρύθμισης των υπερπαραμέτρων από τους αλγορίθμους αναζήτησης βέλτιστων υπερπαραμέτρων για κάθε μοντέλο και σύνολο δεδομένων που χρησιμοποιήθηκε.

**Εικόνα 18.** Γραμμικά γραφήματα.

Παρατηρούμε πως ο BayesSearchCV στα a,b,d,e,f,g και h βρίσκει ένα σύνολο μοντέλων που στο μεγαλύτερο ποσοστό του έχει καλύτερες τιμές ακρίβειας από τα σύνολα μοντέλων που εντοπίζουν οι άλλοι αλγόριθμοι καθώς η καμπύλη του τοποθετείται σχεδόν σε όλο το μήκος της και σχεδόν σε όλα τα γραφήματα πάνω από τις καμπύλες των άλλων αλγορίθμων αναζήτησης με εξαίρεση το διάγραμμα i όπου φαίνεται και οι τρεις καμπύλες να συμπίπτουν άρα οι αλγόριθμοι έχουν παρόμοιες συμπεριφορές και στο c όπου οι καμπύλες του TPOTClassifier και του BayesSearchCV συμπίπτουν άρα επίσης παρουσιάζουν όμοιες συμπεριφορές. Επίσης, με εξαίρεση αυτές τις δύο περιπτώσεις, ο BayesSearchCV φαίνεται να εντοπίζει μοντέλα με καλές τιμές ακρίβειας εκπαιδεύοντας μικρότερο πλήθος μοντέλων σε σχέση με τους άλλους δύο, κάτι που γίνεται φανερό από την μονοτονία της καμπύλης του που παρουσιάζει σε σχέση με τις άλλες απότομη αύξηση στην αρχή της και στην συνέχεια σταθεροποιείται ενώ πάντα βρίσκεται πάνω από τις υπόλοιπες καμπύλες αυτό μάλιστα παρατηρείται έντονα στα b,c,e,h και f ενώ στα άλλα οι διαφορές είναι ελάχιστες ή μηδενικές όπως στο i και d. Έτσι, ο BayesSearchCV φαίνεται να παρουσιάζει γενικά μεγαλύτερη συνέπεια στην αναζήτηση βέλτιστων υπερπαραμέτρων για τα δοσμένα σύνολα δεδομένων και μοντέλα μηχανικής μάθησης.

Έχοντας τα ίδια κριτήρια κατά νου, μπορούμε να εξάγουμε το συμπέρασμα πως και ο TPOTClassifier σε γενικές γραμμές παρουσιάζει συνέπεια καλύτερη του RandomizedSearchCV. Αυτό συμβαίνει γιατί με εξαίρεση το διάγραμμα i όπου υπάρχει ταύτιση, για τα e,g και h όπου οι καμπύλες του TPOTClassifier και του RandomizedSearchCV είναι όμοιες, οι καμπύλες του TPOTClassifier είναι κατά το μεγαλύτερο ποσοστό τους πάνω από αυτές του RandomizedSearchCV. Μπορούμε συνεπώς να συμπεράνουμε πως ο RandomizedSearchCV στα παρόντα σύνολα δεδομένων και μοντέλα μηχανικής μάθησης όσον αφορά την συνέπεια στην αναζήτηση των βέλτιστων υπερπαραμέτρων μπορεί να αποδώσει συμπεριφορά το πολύ παρόμοια με αυτή του TPOTClassifier. Συνεπώς η σύγκριση αυτή έδειξε πως η μέθοδος με την πιο συνεπή διαδικασία αναζήτησης είναι ο BayesSearchCV ακολουθεί ο TPOTClassifier και τέλος ο RandomizedSearchCV.

Μία επίσης σημαντική σύγκριση που θα μπορούσε να εξαχθεί από τέτοιου είδους γραφήματα είναι και ποια μέθοδος επέφερε την εύρεση του καλύτερου μοντέλου. Όμως στην συγκεκριμένη περίπτωση οι καμπύλες σε όλες τις περιπτώσεις συγκλίνουν σε πολύ κοντινά σημεία με αποτέλεσμα να μην είναι εύκολα ορατή η διαφορά τους. Παρ ’όλα αυτά ένα τέτοιο συμπέρασμα μπορεί να εξαχθεί μελετώντας τα αριθμητικά αποτελέσματα των μετρικών που αποθηκεύτηκαν για κάθε βέλτιστο μοντέλο που βρέθηκε. Αυτά παρουσιάζονται στην παράγραφο IV.C.

## Ανάλυση μετρικών βέλτιστων μοντέλων.

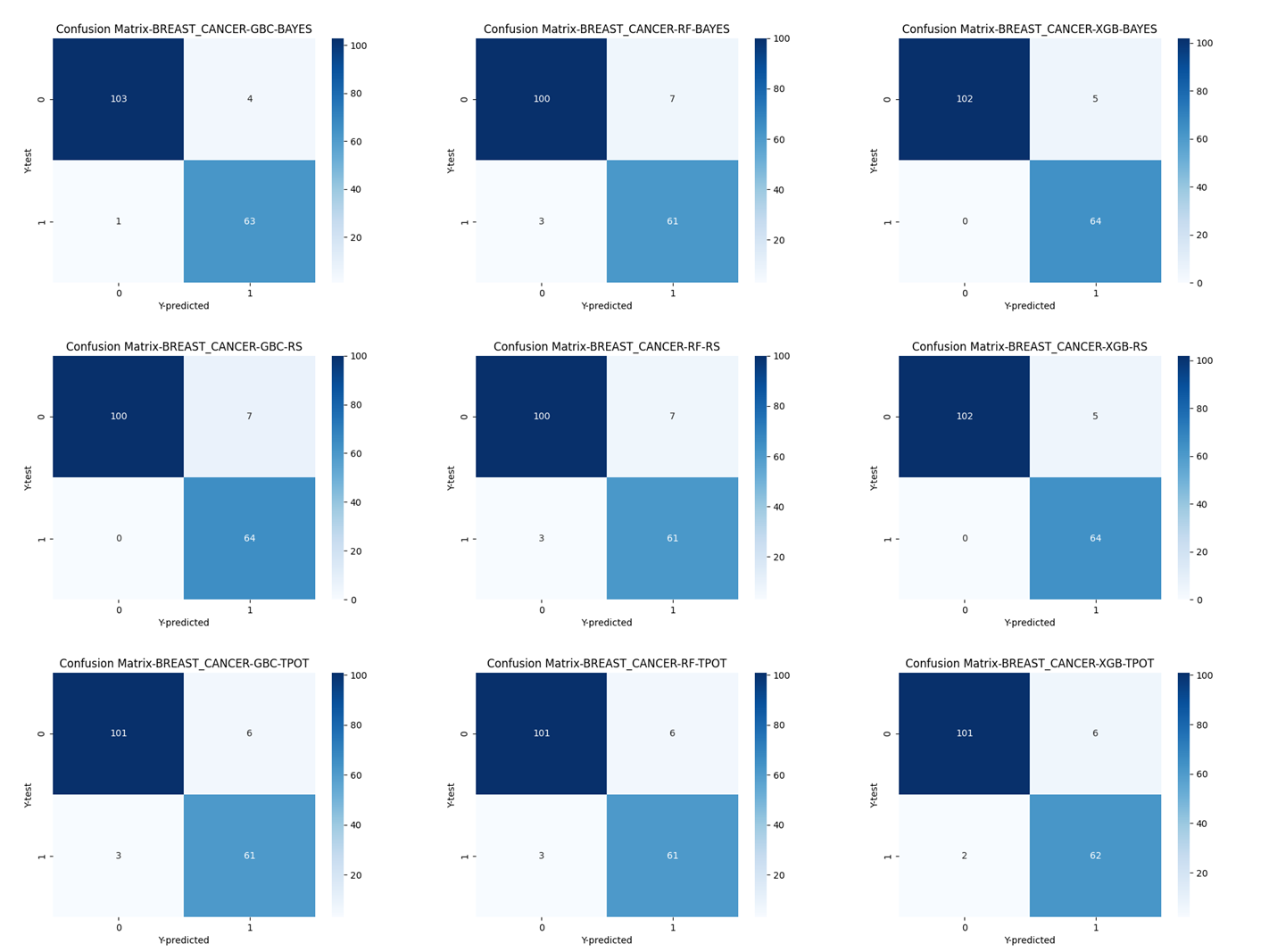
Για την σύγκριση των βέλτιστων μοντέλων και τελικά την σύγκριση των αλγορίθμων αναζήτησης δημιουργήθηκαν τρεις πίνακες σχετικών μετρικών που αφορούν τις επιδόσεις των μοντέλων για τα τρία σύνολα δεδομένων. Αυτοί παρουσιάζονται στο παράρτημα VIII. Επίσης δημιουργήθηκαν πίνακες σύγχυσης για κάθε ένα από τα μοντέλα για μία πιο ολοκληρωμένη εικόνα της σύγκρισης και ερμηνείας των μετρικών. Η ανάλυση των πινάκων του παραρτήματος θα γίνει κατά σύνολο δεδομένων. Έτσι θα αφιερωθεί μία παράγραφος για την ανάλυση τον μοντέλων κάθε συνόλου δεδομένων.

### Ανάλυση για το σύνολο δεδομένων ‘breast cancer’.

Παρατηρώντας τους πίνακες μετρικών **Πίνακας 1, Πίνακας 2, Πίνακας 3** από πάνω προς τα κάτω για τις στήλες που αφορούν το σύνολο δεδομένων πρόγνωσης του καρκίνου του μαστού πρώτα εμφανίζονται τα πεδία που αφορούν την ακρίβεια που υπολογίστηκε από το σύνολο εκπαίδευσης (GBC\_TRAIN,XGB\_TRAIN,RF\_TRAIN). Σε αυτή την περίπτωση οι τιμές βρίσκονται κοντά στο 1 και υποδεικνύουν πως τα μοντέλα για αυτό το σύνολο δεδομένων εκπαιδεύτηκαν κατά τρόπο επαρκή και δεν υπήρξαν σοβαρά προβλήματα με τα δεδομένα που χρησιμοποιήθηκαν ή την πολυπλοκότητα των μοντέλων. Μάλιστα, αν συγκρίνουμε τις ακρίβειες του συνόλου εκπαίδευσης με αυτές του συνόλου ελέγχου (GBC\_TEST,XGB\_TEST,RF\_TEST) βλέπουμε πως επειδή οι διαφορές τους είναι μικρές και βρίσκονται στο δεύτερο δεκαδικό ψηφίο για κάθε μοντέλο, μπορούμε να υποθέσουμε πως δεν υπήρξαν φαινόμενα underfitting ή overfitting.

Επίσης, χρήσιμες πληροφορίες μπορούν να εξαχθούν και από τα πεδία των πινάκων που αφορούν τις τιμές του R2, του συντελεστή Pearson και των p-values. Είναι φανερό από τις τιμές του R2 πως τα μοντέλα XGBoost κατά μέσο όρο προβλέπουν καλύτερα την διακύμανση τον τιμών της κλάσης των δεδομένων ελέγχου που στην περίπτωση μίας κατηγοριοποίησης σημαίνει πως υπάρχουν λιγότερα σφάλματα στις προβλέψεις τους. Οι αμέσως μικρότερες τιμές του R2 εμφανίζονται για τα μοντέλα gradient boosting και τελευταία είναι τα μοντέλα τυχαίων δασών. Επίσης, τα μοντέλα που βρέθηκαν από τον BayesSearchCV έχουν κατά μέσο όρο μεγαλύτερες τιμές από την τυχαία αναζήτηση ενώ οι μικρότερες τιμές παρουσιάζονται στον TPOTClassifier. Οι ίδιες ακριβώς ιεραρχίες παρουσιάζονται και για τον συντελεστή Pearson όπου μεγαλύτερες τιμές είναι ενδεικτικές μίας καλύτερης γραμμικής συσχέτισης των δεδομένων κάτι που καταδεικνύει με ισχυρότερο τρόπο πως υπάρχει μικρό ποσοστό σφαλμάτων. Τέλος, όλες οι τιμές των p-values είναι σαφώς κοντά στο μηδέν που υποδεικνύει πως οι προβλέψεις που δόθηκαν από τα μοντέλα είναι σχετικές με τις πραγματικές τιμές δηλαδή η πρόβλεψη σαφώς δεν γίνεται τυχαία. Τα παραπάνω συμπεράσματα συμφωνούν και με τους πίνακες σύγχυσης **Εικόνα 19** όπου τα τεταρτημόρια σφαλμάτων ψευδών θετικών και ψευδών αρνητικών στο πρώτο και τρίτο τεταρτημόριο παρουσιάζουν σχετικά μικρό πλήθος σφαλμάτων.

Στην συνέχεια, συγκρίνοντας τις τιμές των μετρικών ακρίβειας του συνόλου ελέγχου, τις τιμές ακρίβειας διασταυρούμενης επικύρωσης και μέσου απόλυτου σφάλματος έτσι ώστε για κάθε είδος μοντέλου να βρίσκεται αυτό με τις καλύτερες τιμές μπορεί να εξαχθεί μία εικόνα για την απόδοση των μοντέλων. Βάσει αυτών, μπορεί να παρατηρηθεί πως τα μοντέλα που η εύρεση βέλτιστων υπερπαραμέτρων τους έγινε από τον BayesSearchCV παρουσιάζουν κατά μέσο όρο καλύτερες επιδόσεις από τον αλγόριθμο τυχαίας αναζήτησης του οποίου τα μοντέλα παρουσιάζουν με την σειρά τους καλύτερες επιδόσεις από αυτές των μοντέλων που βρέθηκαν από τον TPOTClassifier όχι όμως για την ακρίβεια διασταυρούμενης επικύρωσης όπου το TPOTClassifier έχει καλύτερες τιμές από την τυχαία αναζήτηση.

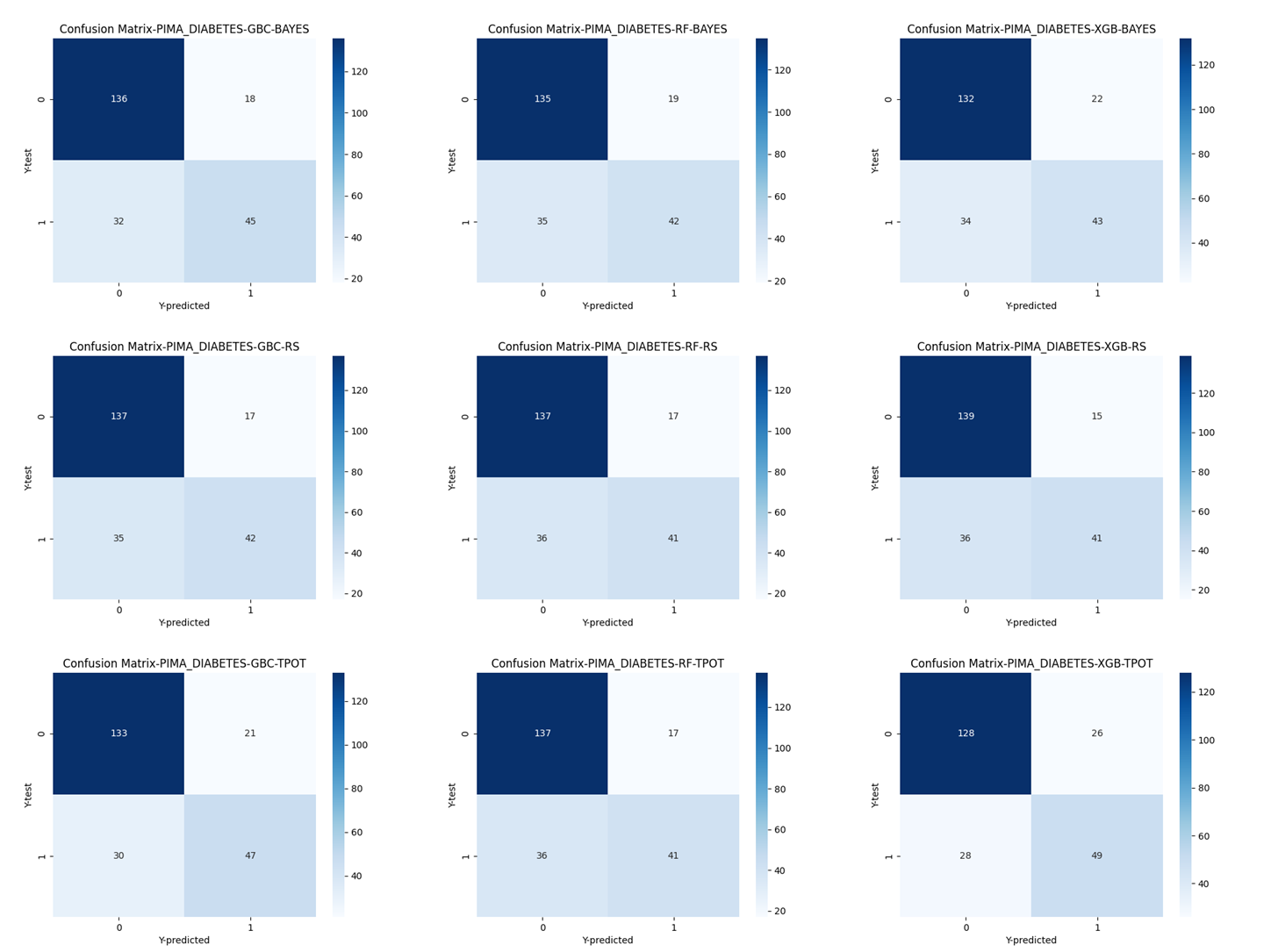
Όμως αυτές οι ενδείξεις στηρίζονται σε αριθμητικά αποτελέσματα που διαφέρουν σε επίπεδο δεύτερου και τρίτου δεκαδικού ψηφίου και για αυτό θα γίνει και ανάλυση των μετρικών ορθότητας (precision) και ανάκλησης (recall) καθώς και μελέτη των πινάκων σύγχυσης για μία πιο αναλυτική σύγκριση. Για το σύνολο δεδομένων πρόγνωσης του καρκίνου του μαστού θα δοθεί ιδιαίτερη βαρύτητα στην μετρική της ορθότητας καθώς η λάθος διάγνωση στην περίπτωση θετικού σε καρκίνο περιστατικού κρίνεται σημαντικότερη από την λάθος διάγνωση σε περίπτωση αρνητικού σε καρκίνο περιστατικού. Έτσι από τις τιμές της ορθότητας φαίνεται πως τα μοντέλα που προέκυψαν από τον BayesSearchCV έχουν μικρότερο πλήθος τέτοιου είδους σφαλμάτων σε σχέση με τις σωστές θετικές διαγνώσεις από τα μοντέλα του TPOTClassifier ενώ τα περισσότερα εμφανίστηκαν στα μοντέλα της τυχαίας αναζήτησης. Σύμφωνα με την μετρική της ανάκλησης τα λιγότερα σφάλματα τύπου δύο σχετικά με τις σωστές θετικές διαγνώσεις εντοπίστηκαν στα μοντέλα τυχαίας αναζήτησης μετά στον BayesSearchCV και τέλος στον TPOTClassifier με τους δύο τελευταίους να παρουσιάζουν μικρές διαφορές. Τα παραπάνω επιβεβαιώνουν και οι πίνακες σύγχυσης **Εικόνα 19**.

**Εικόνα 19**.Πίνακες σύγχυσης για το σύνολο δεδομένων ‘breast cancer’.

Συμπερασματικά, για το σύνολο δεδομένων ‘breast cancer’ βάσει των παραπάνω κριτηρίων μπορεί να ειπωθεί πως ο αλγόριθμος BayesSearchCV παρουσιάζει μοντέλα τα οποία είναι πιο αποτελεσματικά και συνεπή στις προβλέψεις τους από τα μοντέλα του TPOTClassifier ο οποίος παράγει μοντέλα γενικά καλύτερα της τυχαίας αναζήτησης όμως όχι με μεγάλες διαφορές στην απόδοση.

### Ανάλυση για το σύνολο δεδομένων ‘Pima diabetes’.

Παρόμοια με πριν, από τους πίνακες μετρικών για τις στήλες που αφορούν το σύνολο δεδομένων για την διάγνωση του διαβήτη παρατηρούμε πως οι τιμές για τις ακρίβειες στο σύνολο εκπαίδευσης είναι απομακρυσμένες από το 1, ιδιαίτερα για τα μοντέλα τυχαίων δασών. Αυτό υποδεικνύει πως υπήρξε πρόβλημα κατά την διάρκεια εκπαίδευσης των μοντέλων και κατόπιν μελέτης του συνόλου δεδομένων εντοπίστηκαν ακραίες τιμές, επίσης υπάρχουν ιδιαιτερότητες στο σύνολο δεδομένων που είναι δύσκολο να εντοπιστούν από τα μοντέλα μηχανικής μάθησης έτσι είναι φυσιολογικό να περιμένουμε μοντέλα με χαμηλότερες αποδόσεις από αυτές που παρατηρήθηκαν στο προηγούμενο σύνολο δεδομένων. Σε κάθε περίπτωση αυτό δεν μας εμποδίζει από το να συγκρίνουμε τα μοντέλα που εκπαιδεύτηκαν για τους σκοπούς της εργασίας. Μάλιστα, αν παρατηρήσουμε τις ακρίβειες του συνόλου εκπαίδευσης σε σχέση με αυτές του συνόλου ελέγχου βλέπουμε πως οι τιμές για το σύνολο ελέγχου δεν είναι ιδιαίτερα μικρές έτσι μπορούμε να υποθέσουμε πως δεν υπήρξαν έντονα φαινόμενα underfitting ή overfitting.

Από τις τιμές του R2, του συντελεστή Pearson και των p-values θα εξάγουμε όπως πριν τα συμπεράσματα. Πιο συγκεκριμένα, από τις τιμές του R2 που είτε βρίσκονται κοντά στο μηδέν ή είναι αρνητικές μπορούμε να πούμε πως η διακύμανση τον τιμών της κλάσης των δεδομένων ελέγχου δεν εξηγείται καλά από τα μοντέλα πράγμα που σημαίνει πως παρουσιάζονται αρκετές λανθασμένες προβλέψεις σε σχέση με τις σωστές. Επίσης οι χαμηλές τιμές του συντελεστή Pearson υποδεικνύουν μία όχι ξεκάθαρη γραμμική συσχέτιση που επίσης υποδεικνύει συσσώρευση σφαλμάτων είτε τύπου ένα είτε τύπου δύο, όμως η θετική του τιμή υποδεικνύει πως τα μοντέλα έχουν μεγαλύτερο πλήθος σωστών προβλέψεων από ότι λανθασμένων. Οι τιμές των p-values είναι σαφώς μεγαλύτερες από το προηγούμενο σύνολο δεδομένων όμως παραμένουν μικρές δηλαδή και πάλι η πρόβλεψη δεν γίνεται τυχαία ή τα μοντέλα συμπεριφέρονται καλύτερα από την τυχαία πρόβλεψη. Με τα συμπεράσματα αυτά συμφωνούν και οι πίνακες σύγχυσης **Εικόνα 20**.

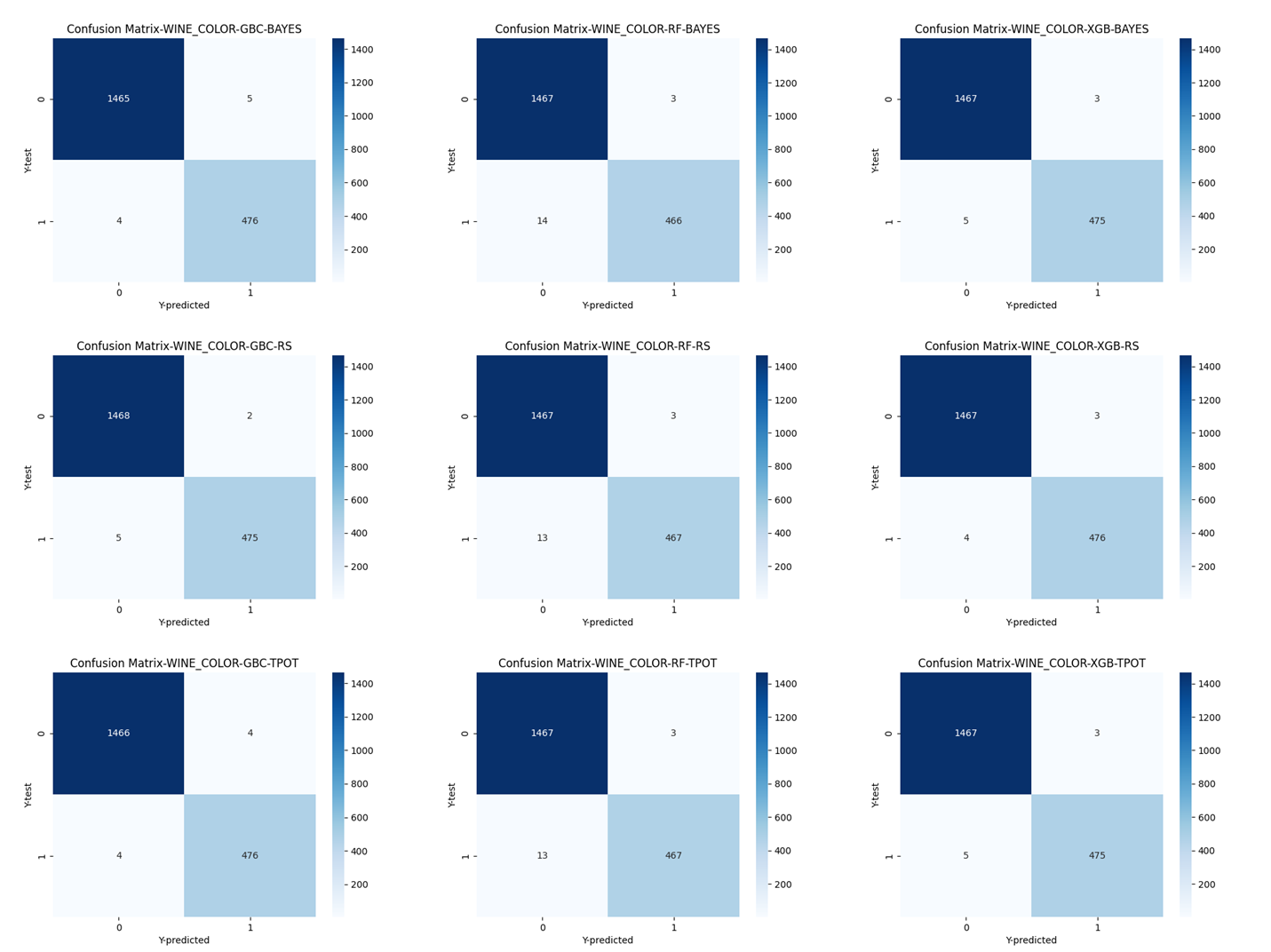
**Εικόνα 20.** Πίνακες σύγχυσης για το σύνολο δεδομένων ‘Pima diabetes’.

Συγκρίνοντας τις τιμές των μετρικών ακρίβειας του συνόλου ελέγχου, τις τιμές ακρίβειας διασταυρούμενης επικύρωσης και μέσου απόλυτου σφάλματος και δίνοντας βαρύτητα στις τιμές της ακρίβειας διασταυρούμενης επικύρωσης που είναι ένας γενικότερος δείκτης για την συμπεριφορά των μοντέλων συμπεραίνουμε πως αν και στα δεδομένα ελέγχου τα μοντέλα τυχαίας αναζήτησης έτυχε να δώσουν καλύτερες τιμές μετρικών από τα μοντέλα των TPOTClassifier και τον BayesSearchCV τα μοντέλα αυτών είναι πιο πιθανό να δώσουν καλύτερη ακρίβεια πρόβλεψης για νέα δεδομένα ελέγχου από τον αλγόριθμο τυχαίας αναζήτησης καθώς είχαν κατά μέσο όρο μεγαλύτερη τιμή ακρίβειας διασταυρούμενης επικύρωσης. Επίσης τα μοντέλα του TPOTClassifier παρουσίασαν γενικά καλύτερες τιμές μετρικών από αυτά του BayesSearchCV.

Μπαίνοντας στην διαδικασία ανάλυσης των σφαλμάτων στο σύνολο ελέγχου παρατηρούμε πως τα σφάλματα τύπου 2 είναι σημαντικά περισσότερα από τα σφάλματα τύπου ένα. Την μεγαλύτερη ακρίβεια και ορθότητα παρουσιάζουν τα μοντέλα της τυχαίας αναζήτησης και με σειρά ακολουθούν τα μοντέλα του BayesSearchCV και TPOTClassifier. Όσον αφορά τις τιμές ανάκλησης που περιγράφουν τα σφάλματα τύπου 2 που είναι πολλά **Εικόνα 20** και συνεπώς εξίσου σημαντικά με τα σφάλματα τύπου 1 σε αυτή την περίπτωση, ισχύει πως ο αλγόριθμος TPOTClassifier κρίθηκε αποδοτικότερος του BayesSearchCV και αυτός του αλγόριθμου τυχαίας αναζήτησης.

Επομένως, για το σύνολο δεδομένων ‘Pima diabetes’ μπορεί να ειπωθεί πως ο αλγόριθμος TPOTClassifier παρουσιάζει μοντέλα τα οποία είναι γενικά πιο συνεπή σε μεγαλύτερης ακρίβειας προβλέψεις από τα μοντέλα του BayesSearchCV και παρότι στο σύνολο ελέγχου τα μοντέλα που προέκυψαν από την τυχαία αναζήτηση είχαν γενικά καλύτερες μετρικές από τα άλλα μοντέλα αυτά δεν παρουσίασαν καλές τιμές ακρίβειας διασταυρούμενης επικύρωσης ούτε καλές τιμές ανάκλησης οπότε αναμένεται να έχουν επιδόσεις γενικότερα κατώτερες*.*

### Ανάλυση για το σύνολο δεδομένων ‘wine color’.

Τέλος, από τους πίνακες μετρικών για τις στήλες που αφορούν το σύνολο δεδομένων πρόβλεψης του χρώματος κρασιού οι τιμές της ακρίβειας στο σύνολο εκπαίδευσης βρίσκονται πολύ κοντά στο 1 και υποδεικνύουν πως τα μοντέλα για αυτό το σύνολο δεδομένων εκπαιδεύτηκαν κατά τρόπο επαρκή ενώ δεν υπήρξαν σημαντικά προβλήματα με το σύνολο δεδομένων. Επίσης, όπως στα προηγούμενα σύνολα δεδομένων μπορούμε να συμπεράνουμε πως δεν υπάρχουν προβλήματα underfitting ή overfitting.

**Εικόνα 21.** Πίνακες σύγχυσης για το σύνολο δεδομένων ‘wine color’.

Τα παραπάνω συμπεράσματα ενισχύονται αν παρατηρήσουμε και τις τιμές των R2, του συντελεστή Pearson και των p-values. Σε κάθε μία από αυτές τις μετρικές παρατηρούμε ιδιαίτερα μεγάλες τιμές σε σχέση με τα μοντέλα των προηγούμενων συνόλων και όλες οι τιμές των p-values είναι μηδενικές. Αυτά τα ευρήματα υποδηλώνουν μία πολύ καλή προσαρμογή των μοντέλων στο σύνολο δεδομένων που συνεπάγεται πολύ μικρό ποσοστό σφαλμάτων κάτι που μπορεί εύκολα να επιβεβαιωθεί κοιτώντας τους αντίστοιχους πίνακες σύγχυσης **Εικόνα 21**.

Αν επίσης εξετάσουμε τις τιμές των μετρικών ακρίβειας του συνόλου ελέγχου, τις τιμές ακρίβειας διασταυρούμενης επικύρωσης και μέσου απόλυτου σφάλματος θα διαπιστώσουμε πως για το μέσο απόλυτο σφάλμα παρουσιάζονται πολύ μικρές τιμές ενώ στις άλλες μετρικές παρουσιάζονται πολύ μεγάλες και σε κάθε περίπτωση οι διαφορές των μοντέλων είναι πολύ μικρές, της τάξης του τρίτου και τέταρτου δεκαδικού ψηφίου. Το γεγονός ύπαρξης τόσο μικρών διαφορών στις επιδόσεις των μοντέλων καθιστά δύσκολο τον διαχωρισμό και αξιολόγησή τους πρακτικά όμως αξιολογώντας τα στο επίπεδο που παρουσιάζουν διαφορές μπορεί να διαπιστωθεί πως τα μοντέλα της τυχαίας αναζήτησης παρουσίασαν γενικά καλύτερες τιμές ακρίβειας στα δεδομένα ελέγχου από αυτά του TPOTClassifier και έπειτα του BayesSearchCV. Όμως ο BayesSearchCV παρουσίασε καλύτερες τιμές ακρίβειας διασταυρούμενης επικύρωσης άρα κατασκεύασε μοντέλα με καλύτερη συνέπεια στις προβλέψεις τους σε νέα δεδομένα από τα υπόλοιπα ενώ δεν παρουσιάζονται πολλές διαφορές στην συνέπεια προβλέψεων μεταξύ τυχαίας αναζήτησης και TPOTClassifier.

Όσον αφορά τις τιμές της ανάκλησης και της ορθότητας επίσης δεν παρουσιάζονται μεγάλες διαφορές. Σε αυτή την περίπτωση κρίνονται σημαντικοί και οι δύο τύποι σφαλμάτων καθώς τα σφάλματα είναι αναλογικά με τις σωστές προβλέψεις ελάχιστα **Εικόνα 21**. Παρατηρούμε πως τις καλύτερες τιμές ορθότητας παρουσιάζουν τα μοντέλα τυχαίας αναζήτησης ενώ για την ανάκληση τα μοντέλα του TPOTClassifier. Ο BayesSearchCV με μικρή διαφορά παρουσίασε τις μικρότερες τιμές στο σύνολο ελέγχου.

Συνεπώς, σε αυτό το σύνολο δεδομένων ο αλγόριθμος BayesSearchCV φαίνεται γενικά να παράγει μοντέλα που δίνουν γενικά καλύτερες προβλέψεις σε νέα δεδομένα ενώ μεταξύ των άλλων δυο παρουσιάζονται ελάχιστες διαφορές. Βάσει του συνόλου ελέγχου καλύτερες επιδόσεις είχε η τυχαία αναζήτηση μαζί με τον TPOTClassifier και έπειτα ο BayesSearchCV. Σε κάθε περίπτωση οι διαφορές μεταξύ των μοντέλων ήταν ελάχιστες που υποδεικνύει σχεδόν παρόμοιες συμπεριφορές των μοντέλων.

### Σύνοψη.

Από τις παραπάνω συγκρίσεις μεταξύ των μοντέλων που παράχθηκαν για κάθε σύνολο δεδομένων μπορούμε να συμπεράνουμε πως στα δύο πρώτα σύνολα δεδομένων οι αλγόριθμοι αναζήτησης που φαίνεται να έχουν καλύτερες επιδόσεις είναι οι BayesSearchCV και TPOTClassifier ενώ στο τρίτο σύνολο δεδομένων αν και δεν υπάρχουν σημαντικές διαφορές αν δώσουμε βαρύτητα στις τιμές της ακρίβειας διασταυρούμενης επικύρωσης φαίνεται να υπάρχει καλύτερη επίδοση από τα μοντέλα του BayesSearchCV ενώ δεν υπάρχει διαφορά για τους άλλους δύο. Επομένως, θα μπορούσε να ειπωθεί πως γενικά τις καλύτερες επιδόσεις παρουσίασαν οι BayesSearchCV και TPOTClassifier και έπειτα ο αλγόριθμος RandomizedSearchCV.

# Συμπεράσματα

Μελετώντας τα πειραματικά αποτελέσματα που παράχθηκαν από την εκτέλεση του κώδικα που δημιουργήθηκε στα πλαίσια αυτής της εργασίας για την σύγκριση των μεθόδων εύρεσης βέλτιστων υπερπαραμέτρων με την χρήση των BayesSearchCV, TPOTClassifier και RandomizedSearchCV και δίνοντας ιδιαίτερη βαρύτητα στους χρόνους εκτέλεσης, στην συνέπεια της διαδικασίας αναζήτησης αλλά και στις επιδόσεις των τελικών βέλτιστων μοντέλων που προέκυψαν καταφέραμε να σχηματίσουμε άποψη για τις επιδόσεις των αλγορίθμων. Από την μελέτη των χρόνων υπολογισμού διαπιστώθηκε πως από τον πιο γρήγορο στην εκτέλεσή του ως προς τον πιο αργό αλγόριθμο έχουμε τον RandomizedSearchCV, TPOTClassifier και BayesSearchCV, όσον αφορά την μεθοδικότητα αναζήτησης των βέλτιστων υπερπαραμέτρων πιο μεθοδικός κρίθηκε ο BayesSearchCV έπειτα ο TPOTClassifier και στην συνέχεια ο RandomizedSearchCV ενώ από την αξιολόγηση των σχετικών μετρικών καταλήξαμε πως οι TPOTClassifier και BayesSearchCV είχαν παρόμοιες επιδόσεις με αυτές του BayesSearchCV να υπερέχουν ενώ ο RandomizedSearchCV παρουσίασε γενικά χαμηλές επιδόσεις.

Από τα παραπάνω, γίνεται φανερό πως ο πιο ισορροπημένος αλγόριθμος εύρεσης βέλτιστων υπερπαραμέτρων είναι ο TPOTClassifier ο οποίος παρουσίασε πολύ καλούς χρόνους εκτέλεσης, διαθέτει μία συνεπή μέθοδο αναζήτησης που σημαίνει πως είναι πιο πιθανό η διαδικασία αναζήτησης να συγκλίνει σε ένα καλό μοντέλο και επίσης είναι πιο πιθανό να βρει μοντέλα που αποδίδουν καλύτερα από αυτά της τυχαίας αναζήτησης. Από την άλλη πλευρά βρίσκεται ο BayesSearchCV ο οποίος είναι σαφώς πιο αργός όμως σε περιπτώσεις όπου υπάρχουν υπολογιστικοί πόροι και η καλή επίδοση των μοντέλων είναι επιτακτική, όπως σε περιπτώσεις ιατρικών διαγνώσεων, μπορεί να δώσει με μεγαλύτερη βεβαιότητα μοντέλα με καλύτερες επιδόσεις. Τέλος, ο αλγόριθμος RandomizedSearchCV παρουσιάζεται ως ο πιο γρήγορος στην εκτέλεσή του αλγόριθμος κάτι που για παράδειγμα μπορεί να φανεί ιδιαίτερα χρήσιμο στην εξερεύνηση του χώρου αναζήτησης υπερπαραμέτρων καθώς δεν είναι πάντα σαφές ποιες υπερπαράμετροι και τιμές είναι πιο σημαντικές για ένα δεδομένο πρόβλημα έτσι ο RandomizedSearchCV μπορεί να φανεί χρήσιμος στην εύρεση περιοχών στο σύνολο αναζήτησης που παρουσιάζουν καλύτερες επιδόσεις για τα μοντέλα.

Επίσης σημαντικά συμπεράσματα προκύπτουν και για τα μοντέλα μηχανικής μάθησης. Όσον αφορά τον χρόνο εκτέλεσης των αναζητήσεων για τα μοντέλα XGBoostClassifier φαίνεται πως η διαδικασία αναζήτησης βέλτιστων υπερπαραμέτρων γίνεται γενικά πιο γρήγορα από τα άλλα δύο είδη μοντέλων. Επίσης φαίνεται από τα **Πίνακας 1**, **Πίνακας 2**, **Πίνακας 3** πως τα βέλτιστα μοντέλα XGBoostClassifier δίνουν γενικά προβλέψεις λίγο καλύτερες από αυτές των μοντέλων GradientBoostingClassifier που με την σειρά τους έχουν αποδόσεις γενικά καλύτερες από τα μοντέλα RandomForestClassifier. Έτσι θα μπορούσε να πει κανείς πως τα πιο ισχυρά μοντέλα λαμβάνοντας υπόψιν την συνέπεια στις προβλέψεις και όσον αφορά την ταχύτητα εκπαίδευσης τους και εύρεσης των υπερπαραμέτρων τους είναι τα μοντέλα XGBoostClassifier έπειτα σε επιδόσεις ακολουθούν τα μοντέλα GradientBoostingClassifier και τέλος τα μοντέλα RandomForestClassifier.

Συμπερασματικά, μπορούμε να καταλήξουμε πως οι πιο αποδοτικοί αλγόριθμοι στα πλαίσια της εργασίας φάνηκαν να είναι οι TPOTClassifier και BayesSearchCV, με τον BayesSearchCV να είναι αργός στην εκτέλεσή του μα πιο ακριβής στα αποτελέσματα και ο TPOTClassifier που φάνηκε να είναι και αποδοτικός και σαφώς πιο γρήγορος. Τέλος ο RandomizedSearchCV βρέθηκε να μην είναι τόσο συνεπής στην εύρεση των βέλτιστων υπερπαραμέτρων αν και γρήγορος στην εκτέλεσή του. Ενώ όσον αφορά τα μοντέλα μηχανικής μάθησης που επιλέχτηκαν καλύτερα λειτούργησαν τα μοντέλα XGBoostClassifier έπειτα τα GradientBoostingClassifier και τέλος τα RandomForestClassifier. Συνεπώς θα μπορούσε να ειπωθεί πως για την εκπαίδευση ενός μοντέλου μηχανικής μάθησης με καλές επιδόσεις σε παρόμοια datasets η σωστή επιλογή αλγορίθμου εύρεσης βέλτιστων παραμέτρων βρίσκεται μεταξύ TPOTClassifier και BayesSearchCV ενώ για την σωστή επιλογή μοντέλου πρέπει να γίνει χρήση του XGBoostClassifier.

# Προτάσεις για μελλοντική έρευνα

Όπως αναφέρθηκε, το πρόβλημα σύγκρισης αλγορίθμων αναζήτησης εξαρτάται από πολλούς παράγοντες με αποτέλεσμα να είναι δύσκολο να χαρτογραφηθεί και να γίνει μία ολοκληρωμένη μελέτη του. Παρά τις προσπάθειες που έγιναν για την εξαγωγή μίας συμπερασματολογίας σχετικά με τις επιδόσεις των συγκρινόμενων αλγορίθμων τα αποτελέσματα που δόθηκαν είναι σαφώς περιορισμένα από τους υπολογιστικούς πόρους που ήταν διαθέσιμοι. Έτσι κρίθηκε σκόπιμο να παρουσιαστούν μερικοί τρόποι με τους οποίους η παρούσα εργασία θα μπορούσε να επεκταθεί και να παρέχει πιο ακριβή συμπεράσματα για ένα σύνολο αλγορίθμων αναζήτησης.

Σημαντικό κομμάτι της παρούσας εργασίας είναι οι χώροι αναζήτησης βέλτιστων υπερπαραμέτρων. Θα μπορούσε να γίνει μία μελέτη γύρω από τους χώρους αναζήτησης υπερπαραμέτρων με σκοπό να επεκταθούν. Κάτι τέτοιο θα μπορούσε να επηρεάσει θετικά τις διαδικασίες αναζήτησης η οποίες για διαφορετικά σύνολα δεδομένων θα ήταν πιθανό να εντοπίσουν πιο αποδοτικά μοντέλα μηχανικής μάθησης. Κάτι τέτοιο βέβαια θα είχε νόημα αν υπήρχε η δυνατότητα να εκπαιδευτούν περισσότερα μοντέλα για την εξαγωγή του βέλτιστου κάτι που απαιτεί σημαντικούς υπολογιστικούς πόρους. Για παράδειγμα, για τον γενετικό αλγόριθμο που χρησιμοποιήθηκε η αύξηση των τιμών των παραμέτρων του πολύ πιθανό να βελτίωνε σημαντικά την διαδικασία αναζήτησης ως προς την αποδοτικότητα του βέλτιστου μοντέλου που εντοπίζει, όμως κάτι τέτοιο θα κόστιζε χρονικά πολύ περισσότερο. Το ίδιο θα μπορούσε να γίνει και για τους άλλους αλγορίθμους αναζήτησης. Ουσιαστικά, ένα μεγαλύτερο σύνολο εκπαιδευμένων μοντέλων κατά την διαδικασία αναζήτησης διερευνά καλύτερα τους πιθανούς συνδυασμούς των υπερπαραμέτρων κάτι που μπορεί να επιφέρει καλύτερα αποτελέσματα.

Μία άλλη πρόταση που θα επέκτεινε την παρούσα εργασία είναι η χρήση περισσότερων μοντέλων μηχανικής μάθησης. Τα μοντέλα μηχανικής μάθησης που χρησιμοποιήθηκαν για την εξαγωγή των πειραματικών αποτελεσμάτων θα μπορούσε να ειπωθεί πως ανήκουν σε μία κατηγορία μοντέλων στηριζόμενη σε δένδρα αποφάσεων. Αυτό το σύνολο μοντέλων αν και αποτελεσματικό στην επίλυση προβλημάτων κατηγοριοποίησης είναι μόνο ένα μικρό υποσύνολο των διαθέσιμων μοντέλων στον τομέα της μηχανικής μάθησης. Έτσι με σκοπό να γίνει μία καλύτερη ανάλυση των αλγορίθμων αναζήτησης θα ήταν καλό αυτοί να εξεταστούν σε μεγαλύτερο πλήθος μοντέλων. Αυτό θα μπορούσε να οδηγήσει σε μία καλύτερη εικόνα της συμβατότητας των αλγορίθμων αναζήτησης με τα αντίστοιχα μοντέλα καθώς ένας αλγόριθμος αναζήτησης θα μπορούσε να παρουσιάζει καλύτερα βέλτιστα μοντέλα έναντι άλλων αλγορίθμων αναζήτησης για ένα μοντέλο μηχανικής μάθησης. Κάτι τέτοιο είναι σαφές πως θα μπορούσε να οδηγήσει σε τεράστιο πλήθος πληροφοριών για ανάλυση και σύγκριση ενώ το πλήθος των μοντέλων που θα έπρεπε να εκπαιδευτεί θα αύξανε σημαντικά και το εγχείρημα αυτό θα απαιτούσε περισσότερο χρόνο και υπολογιστική ισχύ.

Επίσης σημαντικό κομμάτι αυτής της εργασίας είναι και τα σύνολα δεδομένων που χρησιμοποιήθηκαν για την εκπαίδευση των μοντέλων. Στην συγκεκριμένη περίπτωση χρησιμοποιήθηκαν σύνολα δεδομένων τα οποία είχαν ως στόχο την κατηγοριοποίηση δύο κλάσεων τα οποία μάλιστα δεν παρουσίαζαν ανισορροπίες στο πλήθος των κλάσεων. Μόνο έχοντας αυτά τα δύο χαρακτηριστικά των συνόλων δεδομένων κατά νου θα μπορούσαν να επιλεχτούν επιπλέον σύνολα δεδομένων για την επίλυση προβλημάτων πολλαπλής κατηγοριοποίησης ή και παλινδρόμησης ή να επιλεχτούν δεδομένα με ανισορροπίες στις κλάσεις. Ακόμα, θα μπορούσε να γίνει διαχωρισμός των δεδομένων ως προς το πλήθος των χαρακτηριστικών που διαθέτουν για την διαδικασία της εκπαίδευσης και να διερευνηθεί πως αυτό επηρεάζει τις προβλέψεις των μοντέλων και ακόμη την αναζήτηση των βέλτιστων μοντέλων. Με άλλα λόγια, μία επέκταση της εργασίας θα μπορούσε να είναι η διεκπεραίωση διαφορετικών μελετών για σύνολα δεδομένων με διαφορές στις ιδιότητές τους με σκοπό να εντοπιστούν πιθανές επαναλαμβανόμενες συμπεριφορές των αλγορίθμων αναζήτησης και των μοντέλων μηχανικής μάθησης.

Για όλα τα παραπάνω σημαντική είναι η αποτελεσματική και ταχεία εκτέλεση ενός κώδικα που απαιτεί μεγάλη υπολογιστική ισχύ, κάτι που χρειάζεται τη στρατηγική εφαρμογή διαφόρων εργαλείων και τεχνικών για τη βελτιστοποίηση της απόδοσης και την καλύτερη χρήση των υπολογιστικών πόρων. Στην περίπτωση εργασίας πάνω σε μεγάλα σύνολα δεδομένων και εκτέλεσης πολύπλοκων υπολογισμών, η παραδοσιακή σειριακή επεξεργασία μπορεί να οδηγήσει σε μεγάλους χρόνους εκτέλεσης. Συνεπώς στην περίπτωση επέκτασης της εργασίας περαιτέρω θα μπορούσε να γίνει χρήση επιπλέον μέσων όπως αυτά που προτείνονται πιο κάτω.

Οι παράλληλοι υπολογισμοί αναδεικνύονται ως μια πολύ καλή λύση σε αυτήν την πρόκληση. Διαιρώντας τον φόρτο εργασίας σε μικρότερες εργασίες που μπορούν να υποβληθούν σε επεξεργασία ταυτόχρονα σε πολλούς πυρήνες ή μονάδες επεξεργασίας, ο παράλληλος υπολογισμός μπορεί να μειώσει σημαντικά τον χρόνο υπολογισμού. Επίσης, μια άλλη σημαντική μέθοδος σχετική με την διαχείριση του υπολογιστικού χρόνου πολύπλοκων υπολογιστικών διεργασιών είναι το cloud computing. Η αξιοποίηση των ηλεκτρονικών πλατφόρμων cloud επιτρέπει στους ερευνητές και τους προγραμματιστές να έχουν πρόσβαση σε σημαντικά μεγάλους υπολογιστικούς πόρους χωρίς το βάρος της παροχής και συντήρησης υλικού. Αυτή η προσέγγιση όχι μόνο επιταχύνει το έργο αλλά παρέχει επίσης οικονομική αποδοτικότητα και ευελιξία στην κατανομή των διαθέσιμων πόρων.

Μία ακόμη προσέγγιση θα ήταν η χρήση πιο αποδοτικών αλγόριθμων οι οποίοι διαδραματίζουν κεντρικό ρόλο στην επιτάχυνση των υπολογισμών. Η επιλογή αλγορίθμων που ταιριάζουν με τις συγκεκριμένες απαιτήσεις του έργου μπορεί να μειώσει σημαντικά την υπολογιστική πολυπλοκότητα και να επιταχύνει το χρόνο εκτέλεσης. Επιπλέον, η συνετή προεπεξεργασία δεδομένων, συμπεριλαμβανομένης της κανονικοποίησης, της επιλογής χαρακτηριστικών και της μείωσης διαστάσεων, μπορεί να βελτιώσει σημαντικά τους υπολογισμούς και να αυξήσει την απόδοση των μοντέλων μηχανικής μάθησης. Η βελτιστοποίηση του κώδικα είναι ζωτικής σημασίας για τη διασφάλιση γρήγορης εκτέλεσης. Η σύνταξη καλά δομημένου και βελτιστοποιημένου κώδικα μπορεί να εξαλείψει περιττούς βρόχους, να οργανώσει διάφορες λειτουργίες και να εκμεταλλευτεί βιβλιοθήκες με βελτιστοποιημένες συναρτήσεις. Αυτά τα μέτρα οδηγούν σε ταχύτερους χρόνους εκτέλεσης και βελτιωμένη συνολική απόδοση.

Επιπλέον, η χρήση τεχνικών προσωρινής αποθήκευσης μπορεί να αποδειχτεί πλεονεκτική σε ορισμένες περιπτώσεις. Η αποθήκευση ενδιάμεσων αποτελεσμάτων μπορεί να αποτρέψει περιττούς υπολογισμούς, με αποτέλεσμα ταχύτερους χρόνους εκτέλεσης όταν απαιτούνται εκ νέου υπολογισμοί. Αυτή η προσέγγιση είναι ιδιαίτερα ωφέλιμη όταν έχουμε να κάνουμε με επαναληπτικούς αλγόριθμους ή πολύπλοκες προσομοιώσεις.

Καταλήγοντας, μια επιτυχημένη στρατηγική για την ταχύτερη εκτέλεση μίας επέκτασης της συγκεκριμένης εργασίας θα απαιτούσε την έξυπνη χρήση διαφόρων μέσων και τεχνικών. Η χρήση παράλληλων υπολογισμών, η αξιοποίηση των δυνατοτήτων των πλατφόρμων cloud, η εφαρμογή αποτελεσματικών αλγορίθμων, η βελτιστοποίηση κώδικα και η χρήση προσωρινής αποθήκευσης είναι όλα τα βασικά στοιχεία που θα μπορούσαν να οδηγούσαν στην επίτευξη ταχύτερων χρόνων εκτέλεσης , διαχείρισης των υπολογιστικών πόρων και συνεπώς στην διεκπεραίωση της έρευνας.

# Βιβλιογραφία

Afifah, K., Yulita, I. N. & Sarathan, I., 2021. Sentiment Analysis on Telemedicine App Reviews using XGBoost Classifier. *International Conference on Artificial Intelligence and Big Data Analytics*, pp. 22-27.

Bacanin, N. και συν., 2023. On the Benefits of Using Metaheuristics in the Hyperparameter Tuning of Deep Learning Models for Energy Load Forecasting. *Energies*, 01 February.

Bergstra, J., Bardenet, R., Bengio, Y. & Kegl, B., 2011. Algorithms for Hyper-Parameter Optimization. *Advances in neural information processing systems 24*.

Brochu, E., Cora, V. M. & de Freitas, N., 2010. A Tutorial on Bayesian Optimization of Expensive Cost Functions, with Application to Active User Modeling and Hierarchical Reinforcement Learning. *arXiv preprint arXiv:1012.2599* , 14 December.

Chai, T. & Draxler, R. R., 2014. Root mean square error (RMSE) or mean absolute error (MAE)?. *Geoscientific Model Development*, 30 06, pp. 1247-1250.

Chakrabarty, N. και συν., 2018. Flight Arrival Delay Prediction Using Gradient Boosting Classifier. *Advances in Intelligent Systems and Computing*, 02 09.

Fränti, P. & Mariescu-Istodor, R., 2023. Soft precision and recall. *Pattern Recognition Letters*, 03.

Gevorkyan, M. N., Demidova, A. V. & Kulyabov, D. S., 2020. Comparative analysis of machine learning methods by the example of the problem of determining muon decay. *Discrete and Continuous Models and Applied Computational Science*, 15 12, pp. 105-119.

Kohavi, R., 1995. A Study of Cross-Validation and Bootstrap for Accuracy Estimation and Model Selection. *Appears in the International Joint Conference on Artificial Intelligence(IJCAI)*, August, pp. 1137-1145.

PLEVRIS, V., SOLORZANO, G., BAKAS, N. . P. & EL AMINE BEN SEGHIER, M., 2022. INVESTIGATION OF PERFORMANCE METRICS IN REGRESSION ANALYSIS AND MACHINE LEARNING-BASED PREDICTION MODELS. *The 8th European Congress on Computational Methods in Applied Sciences and Engineering* , 9 June.

Probst, P., Wright, M. & Boulesteix, A.-L., 2019. Hyperparameters and tuning strategies for random forest. *WIREs Data Mining and Knowledge Discovery*, 28 01.

Visa, S., Ramsay, B., Ralescu, A. & Knaap, E. v. d., 2011. Confusion Matrix-based Feature Selection. *Proceedings of the Twenty­second Midwest Artificial Intelligence and Cognitive Science Conference*, 17 April, pp. 120-127.

Yu, T. & Zhu, H., 2020. Hyper-Parameter Optimization: A Review of Algorithms. 12 March.

Zahedi, L. και συν., 2021. Search Algorithms for Automated Hyper-Parameter Tuning. *arXiv preprint arXiv:2104.14677* , 29 April.

@trivialfis, 2023. *GitHub - dmlc/xgboost: Scalable, Portable and Distributed Gradient Boosting (GBDT, GBRT or GBM) Library, for Python, R, Java, Scala, C++ and more. Runs on single machine, Hadoop, Spark, Dask, Flink and DataFlow.* [Ηλεκτρονικό]   
Available at: https://github.com/dmlc/xgboost  
[Πρόσβαση 18 August 2023].

Abu Syeed , S. A. και συν., 2021. An Approach to Detect Cyber Attack on Server-side Application by using Data Mining Techniques and Evolutionary Algorithms. *International Journal of Applied Information Systems (IJAIS) ISSN : 2249-0868*, June.

Alibrahim, H. & Ludwig, S. . A., 2021. Hyperparameter Optimization: Comparing Genetic Algorithm against Grid Search and Bayesian Optimization. *2021 IEEE Congress on Evolutionary Computation (CEC)*, 28 06.

ARAUNA, K. V., 2021. COMPARISON BETWEEN STOCHASTIC GRADIENT DESCENT AND SUPPORT VECTOR MACHINE ON ECOMMERCE DATA USING RANDOMIZEDSEARCHCV. 30 January, p. 32.

Bakas, . N. και συν., 2023. *nbml: Computer Software for Data Analysis and Predictive Modelling with Artificial Intelligence Algorithms.* [Ηλεκτρονικό]   
Available at: https://github.com/nbakas/nbml/blob/main/docs/\_\_nbml\_\_.pdf  
[Πρόσβαση 04 July 2023].

Bergstra, J., Yamins , D. & Cox , . D. D., 2013. *Making a Science of Model Search: Hyperparameter Optimization in Hundreds of Dimensions for Vision Architectures.* Atlanta, s.n.

Chen, T. & Guestrin, C., 2016. XGBoost: A Scalable Tree Boosting System. *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 13 08.

Datta, P. & Faroughi, . S. . A., 2022. A Multihead Lstm Technique for Prognostic Prediction of Soil Moisture. *SSRN Electronic Journal*.

Franceschi, L., Donini , M., Frasconi, P. & Pontil, M., 2017. Forward and Reverse Gradient-Based Hyperparameter Optimization. *International Conference on Machine Learning. PMLR*, July, pp. 1165-1173.

Hutter, F., Lucke, J. & Schmidt-Thieme, L., 2015. Beyond Manual Tuning of Hyperparameters. *KI - Künstliche Intelligenz*, 11 July.

jay-m-dev, 2023. *GitHub - EpistasisLab/tpot: A Python Automated Machine Learning tool that optimizes machine learning pipelines using genetic programming..* [Ηλεκτρονικό]   
Available at: https://github.com/EpistasisLab/tpot  
[Πρόσβαση 15 August 2023].

LEARNING, U. M., 2016. *Kaggle.* [Ηλεκτρονικό]   
Available at: https://www.kaggle.com/datasets/uciml/pima-indians-diabetes-database  
[Πρόσβαση 14 June 2023].

LEARNING, U. M., 2016. *Kaggle.* [Ηλεκτρονικό]   
Available at: https://www.kaggle.com/datasets/uciml/breast-cancer-wisconsin-data?select=data.csv  
[Πρόσβαση 14 June 2023].

Mantovani, R. . G. και συν., 2016. Hyper-Parameter Tuning of a Decision Tree Induction Algorithm. *2016 5th Brazilian Conference on Intelligent Systems (BRACIS)*, 10.

Mantovani, R. G. και συν., 2019. An empirical study on hyperparameter tuning of decision trees. 12 February.

Özdemir, M. A. και συν., 2023. Machine learning to predict the antimicrobial activity of cold atmospheric plasma-activated liquids. *Machine Learning: Science and Technology*, 01 03.

Pal, M., 2005. Random forest classifier for remote sensing classification. *International Journal of Remote Sensing*, 01.

Panichella, A., 2021. A Systematic Comparison of search-Based approaches for LDA hyperparameter tuning. *Information and Software Technology*, 02.

Rahmadayana, F. & Sibaroni, Y., 2021. Sentiment Analysis of Work from Home Activity using SVM with Randomized Search Optimization. *Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi)*, 25 October.

RUTHGN, 2021. *Kaggle.* [Ηλεκτρονικό]   
Available at: https://www.kaggle.com/datasets/ruthgn/wine-quality-data-set-red-white-wine  
[Πρόσβαση 14 June 2023].

Sinha, N. K., Khulal, M., Gurung, M. & Lal, . A., 2020. Developing A Web based System for Breast Cancer Prediction using XGboost Classifier. *International Journal of Engineering Research & Technology (IJERT)*, June.

Snoek, J., Larochelle, H. & Adams, R. . P., 2012. Practical Bayesian Optimization of Machine Learning Algorithms. *Advances in neural information processing systems 25* .

Thisted, R. . A., 1998. What is a P-value?. *Departments of Statistics and Health Studies*.

Turner, R. και συν., 2021. Bayesian Optimization is Superior to Random Search for Machine Learning Hyperparameter Tuning: Analysis of the Black-Box Optimization Challenge 2020. *NeurIPS 2020 Competition and Demonstration Track-Proceedings of Machine Learning Research*, 26 03.

Wu, J. και συν., 2019. Hyperparameter Optimization for Machine Learning. *JOURNAL OF ELECTRONIC SCIENCE AND TECHNOLOGY*, March, pp. 26-40.

Xie, Y. και συν., 2018. Evaluation of machine learning methods for formation lithology identification: A comparison of tuning processes and model performances. *Journal of Petroleum Science and Engineering*, January.

Yang, L. & Shami, A., 2020. On Hyperparameter Optimization of Machine Learning. *Neurocomputing*, November.

# Παράρτημα



**Πίνακας 1.** Πίνακας μετρικών για τα μοντέλα gradient boosting.



**Πίνακας 2.** Πίνακας μετρικών για τα μοντέλα τυχαίων δασών.



**Πίνακας 3.** Πίνακας μετρικών για τα μοντέλα XGBoost.