ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ
ΔΙΑΤΜΗΜΑΤΙΚΟ ΠΡΟΓΡΑΜΜΑ ΜΕΤΑΠΤΥΧΙΑΚΩΝ ΣΠΟΥΔΩΝ
«ΣΥΣΤΗΜΑΤΑ ΑΥΤΟΜΑΤΙΣΜΟΥ»

ΜΕΤΑΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

Συγκριτική αξιολόγηση μοντέλων μηχανικής μάθησης για την ταξινόμηση ελαττωμάτων σε χαλύβδινες πλάκες

Μεταπτυχιακός Φοιτητής : Κορδάτος Ιωάννης

Επιβλέπων Καθηγητής : Μπενάρδος Πανώριος

ΑΘΗΝΑ , ΙΟΥΝΙΟΣ 2021

NATIONAL TECHNICAL UNIVERSITY OF ATHENS
MASTER PROGRAM OF NATIONAL TECHNICAL UNIVERSITY OF ATHENS «AUTOMATION SYSTEMS»

Comparative analysis of machine learning algorithms for steel plates defects classification

BY

Ioannis D. Kordatos

SUPERVISOR

Assistant Prof. Panorios Benardos

ATHENS, JUNE 2021

Στους γονείς μου…

Ευχαριστίες

Εξαρχής, αισθάνομαι την ανάγκη να ευχαριστήσω και να εκφράσω την βαθύτατη εκτίμηση μου στον επιβλέποντα καθηγητή μου κ. Πανώριο Μπενάρδο για την στάση που έδειξε απέναντι μου καθώς και για την συνεχή καθοδήγηση και την πολύτιμη βοήθεια που μου προσέφερε στο πλαίσιο εκπόνησης της συγκεκριμένης μεταπτυχιακής εργασίας.

Επιπλέον, πολλές ευχαριστίες στους φίλους μου και σε αυτούς που είναι δίπλα μου. Η παρουσία τους είναι απαραίτητη.

Τέλος, θα ήθελα να εκφράσω την ευγνωμοσύνη μου στους γονείς μου Έλση και Δημήτρη, στους οποίους αφιερώνω την παρούσα εργασία για την συμπαράσταση και την συνεχή τους στήριξη στην εκπλήρωση των στόχων μου.

Περίληψη

Αντικείμενο της παρούσας μεταπτυχιακής εργασίας είναι η συγκριτική αξιολόγηση διάφορων αλγορίθμων μηχανικής μάθησης για την ταξινόμηση ελαττωμάτων σε πλάκες χάλυβα. Η σωστή διάγνωση και κατηγοριοποίηση (fault diagnosis) των ελαττωμάτων σε πλάκες χάλυβα αποτελεί μία σημαντική πρόκληση για τον τομέα της Παραγωγής. Μοντέλα μηχανικής μάθησης μπορούν να χρησιμοποιηθούν επικουρικά για την ανίχνευση ελαττωμάτων με σκοπό την καλύτερη ποιότητα του τελικού προϊόντος. Λάθος κατηγοριοποιήσεις και καταγραφές ελαττωμάτων που γίνονται κατά τον Ποιοτικό Έλεγχο των εργοστασίων μπορεί να έχουν σοβαρές επιπτώσεις που φτάνουν μέχρι και σε αστοχίες κατασκευών. Το σύνολο δεδομένων που χρησιμοποιήθηκε προέρχεται από το ερευνητικό κέντρο Semeion στην Ρώμη της Ιταλίας και αποτελείται από 1941 παρατηρήσεις, 27 χαρακτηριστικά και 7 τύπους ελαττωμάτων. Τα χαρακτηριστικά έχουν εξαχθεί, μέσω εργαλείων επεξεργασίας εικόνας, από φωτογραφίες των πλακών χάλυβα και αποτυπώνουν την γεωμετρική φύση των ελαττωμάτων. Οι τύποι των ελαττωμάτων για τις πλάκες χάλυβα είναι οι Pastry, Z_Scratch, K_Scatch, Stains, Dirtiness, Bumps και Other_Faults. Το πρόβλημα ανήκει στην κατηγορία πολλαπλής ταξινόμησης ελαττωμάτων με ανομοιογενή δεδομένα (imbalanced multiclass classification). Έγινε εφαρμογή πέντε μοντέλων μηχανικής μάθησης, τα οποία είναι τα Logistic Regression , Support Vector Machines, k-Nearest Neighbor, Random Forest και Gradient Boosting Trees, για την δημιουργία συγκρίσεων και επιπλέον υλοποιήθηκαν τεχνικές ανάλυσης, προ-επεξεργασίας, μετασχηματισμοί και οπτικοποιήσεις των δεδομένων. Πραγματοποιήθηκε ρύθμιση των παραμέτρων (hyperparameter tuning) και βελτιστοποίηση των μοντέλων με την δημιουργία πλέγματος για διαφορετικά εύρη των τιμών τους. Ακόμα, χρησιμοποιήθηκαν τεχνικές εύρεσης της σημαντικότητας των χαρακτηριστικών μέσω μοντέλων (model based feature importances) και αμοιβαίας πληροφορίας (mutual information). Για όλες τις διαφορετικές προ-επεξεργασίες των δεδομένων που πραγματοποιήθηκαν, το μοντέλο Random Forest έδωσε τα υψηλότερα αποτελέσματα. Η μεγαλύτερη ορθότητα ταξινόμησης για το RF ήταν 0.825 με πολυωνυμική προ-επεξεργασία (n=2) των δεδομένων και δημιουργία ενός καινούριου dataset με 54 χαρακτηριστικά. Επίσης, χρησιμοποιήθηκε η τεχνική παραγωγής συνθετικών δεδομένων (Adasyn) με το μοντέλο RF να δίνει αποτελέσματα ορθότητας 0.910. Τέλος, αναπτύχθηκε ένας προσαρμοσμένος ταξινομητής τριών σταδίων ταξινόμησης (three stages – custom classifier) με ορθότητα μεγαλύτερη του 0.90 σε κάθε στάδιο. Συνολικά, έγιναν 22 πειράματα εφαρμογής μοντέλων μηχανικής μάθησης και όλοι οι πίνακες σύγχυσης αντιστοιχούν στα τεστ ελέγχου. Η ανάπτυξη λογισμικού έγινε στην Python 3.8.8 με χρήση Jupyter για notebook που τρέχει μέσα σε περιβάλλον Anaconda (Windows). Τα δεδομένα προμηθεύτηκαν από το UCI Machine Learning Repository του University College of Irving (Provider).

α

Abstract

The evaluation of steel plates in the manufacturing sector is vital for the quality of the products and Fault Diagnosis (FD) has to be taken very seriously. The classification of different defects of steel plates is essential for the prevention of life threatening risks. Manual defect inspection by the Quality Control department is an exhausting assignment with many difficulties involved. The surety of the classification manually it is not guaranteed because of the big number of steel plates investigations that have to take place. To meet user requirements in this master thesis five machine learning algorithm have been tested for comparative analysis for the Faulty Steel Plates dataset provided by UCI machine learning repository. This study deals with the classification and diagnosis of seven commonly occurring faults of the steel plate that are named Pastry, Z_Scratch, K_Scatch, Stains, Dirtiness, Bumps and Other_Faults. The features of the dataset have been extracted by image processing of 1941 photographs of steel plates. The ML models Logistic Regression, Support Vector Machines, k-Nearest Neighbor, Random Forest and Gradient Boosting Trees have been trained and tuned using GridSearch CV with different data preparation techniques. Model based feature importances and mutual information results are generated. The synthetic data production technique Adasyn (Adaptive Synthetic Sampling) has been used to tackle the problem of class imbalances. A three – stage custom classifier has been implemented for reasons of high diagnostic accuracy. Random forest with polynomial feature engineering (n=2) and a 54 features dataset gave 0.825 accuracy. Synthetic data generation with ADASYN as a preprocessing technique in combination with RF reached 0.910 accuracy. The custom classifier has over 0.900 accuracy in every of the three classifying stages.

Keywords: Automated Manufacturing, Fault diagnosis, Steel plate faults, multiclass classification, imbalanced learning, Random Forests

ΠΙΝΑΚΑΣ ΠΕΡΙΕΧΟΜΕΝΩΝ

Περίληψη **5**

Abstract **6**

ΚΑΤΑΛΟΓΟΣ ΕΙΚΟΝΩΝ ΚΑΙ ΣΧΗΜΑΤΩΝ **9**

ΚΑΤΑΛΟΓΟΣ ΠΙΝΑΚΩΝ **10**

Κεφάλαιο 1: ΕΙΣΑΓΩΓΗ **11**

1.1 Βιομηχανική Επανάσταση 4.0 11

1.2 Προσφορά της Μηχανικής Μάθησης στην Παραγωγή Πλακών Χάλυβα 12

1.3 Στόχος και περιγραφή της εργασίας 14

1.4 Περιγραφή Δεδομένων και Προβλήματος 15

1.5 Βιβλιογραφική Ανασκόπηση 19

1.5.1 Προσέγγιση με απλή εφαρμογή μοντέλων μηχανικής μάθησης 19

1.5.2 Προσέγγιση με ρύθμιση των παραμέτρων 19

1.5.3 Προσέγγιση με επιλογή των καλύτερων χαρακτηριστικών βάσει μοντέλων μηχανικής μάθησης 20

1.5.4 Προσέγγιση με αφαίρεση ακραίων τιμών και εφαρμογή μοντέλων μηχανικής μάθησης 21

1.5.5 Σύνθετες προσεγγίσεις **21**

Κεφάλαιο 2: ΜΗΧΑΝΙΚΗ ΜΑΘΗΣΗ **23**

2.1 Εργαλειοθήκη 23

2.1.1 Γλώσσα Προγραμματισμού και Εργαλεία 23

2.1.2 Κύριες Βιβλιοθήκες 23

2.1.3 Numpy 24

2.1.4 Pandas 24

2.1.5 Scikit- learn 25

2.2 Μοντέλα Μηχανικής Μάθησης 25

2.2.1 Logistic Regression 25

2.2.2 K-Nearest Neighbor (k-NN) 26

2.2.3 Support Vector Machines 26

2.2.4 Random Forest 27

2.2.5 Gradient Boosting Trees 27

2.3 Ρύθμιση Παραμέτρων Μοντέλων Μηχανικής Μάθησης 28

2.4 Αξιολόγηση Μοντέλων 29

Κεφάλαιο 3: ΑΝΑΛΥΣΗ ΠΡΟΒΛΗΜΑΤΟΣ 32
3.1 Ανάλυση Δεδομένων 32
3.2 Εύρεση των χαρακτηριστικών με την μεγαλύτερη σημαντικότητα 41
3.3 Οπτικοποίηση των δεδομένων και μείωση διάστασης 44
3.4 Μετασχηματισμοί Δεδομένων 49
3.4.1 Κατασκευή πολυωνυμικών χαρακτηριστικών (Polynomial Feature Engineering) 49
3.4.2 RBF Kernel και PCA 51

Κεφάλαιο 4: ΑΝΑΠΤΥΞΗ ΜΟΝΤΕΛΩΝ 52
4.1 Εισαγωγή 52
4.2 Μέθοδοι προ-επεξεργασίας των δεδομένων 52
4.2.1 Απλή ρύθμιση υπέρ – παραμέτρων 52
4.2.2 Μείωση Διαστατικότητας και εφαρμογή μοντέλων 54
4.2.3 Πολυωνυμική προ-επεξεργασία των δεδομένων 56
4.2.4 Επιλογή Χαρακτηριστικών και Ρύθμιση Παραμέτρων 58
4.3 Αποτελέσματα μοντέλων 60

Κεφάλαιο 5: ΣΥΝΘΕΤΙΚΑ ΔΕΔΟΜΕΝΑ 63
5.1 Εισαγωγή 63
5.2 Δημιουργία συνθετικών δεδομένων 63
5.3 Προετοιμασία δεδομένων 64
5.4 Αποτελέσματα Μοντέλου 66

Κεφάλαιο 6: ΠΡΟΣΑΡΜΟΣΜΕΝΟΣ ΤΑΞΙΝΟΜΗΤΗΣ 68
6.1 Εισαγωγή 68
6.2 Διαδικασία επίλυσης σε τρία επίπεδα ταξινόμησης 68
6.2.1 Πρώτο Στάδιο Ταξινόμησης – Common Faults vs Other Faults 72
6.2.2 Δεύτερο Στάδιο Ταξινόμησης – Φανταστικές Κλάσεις 74
6.2.3 Τρίτο Στάδιο Ταξινόμησης – Τρεις Δυαδικές Ταξινομήσεις 76
6.3 Συγκεντρωτικά αποτελέσματα και Mutual Information 83

Κεφάλαιο 7: ΤΡΟΠΟΠΟΙΗΣΗ ΤΟΥ ΠΡΟΒΛΗΜΑΤΟΣ 86
7.1 Εισαγωγή 86
7.2 Διαδικασία Επίλυσης με Gradient Boosting Trees 86
7.3 Αποτελέσματα 88

Κεφάλαιο 8: ΣΥΜΠΕΡΑΣΜΑΤΑ 90

ΒΙΒΛΙΟΓΡΑΦΙΑ 92

ΠΑΡΑΡΤΗΜΑ 1. ΓΡΑΜΜΙΚΗ ΣΥΣΧΕΤΙΣΗ ΧΑΡΑΚΤΗΡΙΣΤΙΚΩΝ 97

ΠΑΡΑΡΤΗΜΑ 2. ΚΑΤΑΝΟΜΕΣ ΔΕΔΟΜΕΝΩΝ 98

ΠΑΡΑΡΤΗΜΑ 3. ΔΙΑΓΡΑΜΜΑΤΑ ΧΑΡΑΚΤΗΡΙΣΤΙΚΩΝ 111

ΠΑΡΑΡΤΗΜΑ 4. ΠΙΝΑΚΕΣ ΣΥΓΧΥΣΗΣ – ΚΕΦΑΛΑΙΟ 4 118

ΚΑΤΑΛΟΓΟΣ ΕΙΚΟΝΩΝ ΚΑΙ ΣΧΗΜΑΤΩΝ

Εικόνα 1. Εφαρμογές αυτόματης επιθεώρησης τεμαχίων στο ΕΤΚ-ΕΜΠ με μηχανική μάθηση. 14
Εικόνα 2. Ποσοστά τύπων ελαττωμάτων 18
Εικόνα 3. Logo Numpy 24
Εικόνα 4. Logo Pandas 25
Εικόνα 5. Logo Scikit 25
Εικόνα 6. Bagging vs Boosting 28
Εικόνα 7. Normalization vs Standardization 35
Εικόνα 8. Κατανομή χαρακτηριστικού Empty_Index 38
Εικόνα 9. Κατανομή χαρακτηριστικού Steel plate thickness 38
Εικόνα 10. Κατανομή χαρακτηριστικού LogOfAreas 39
Εικόνα 11. Τιμές χαρακτηριστικού X_Maximum και χαρακτηριστικού X_Minimum σε σχέσεις με τις κλάσεις τους 40
Εικόνα 12. Τιμές χαρακτηριστικού Edges_Y_Index και χαρακτηριστικού Edges_X_Index σε σχέση με τις κλάσεις τους 41
Εικόνα 13. Mutual Information για K_Scratch vs Stains 44
Εικόνα 14. Pareto diagram – PCA 45
Εικόνα 15. Διάγραμμα PCA - δύο συνιστώσες 46
Εικόνα 16. Διάγραμμα PCA - τρεις συνιστώσες 47
Εικόνα 17. Διάγραμμα T-SNE - δύο vectors 48
Εικόνα 18. Διάγραμμα t-SNE - τρία vectors 48
Εικόνα 19. Δίαγραμμα κώδικα με απλη ρύθμιση παραμέτρων 53
Εικόνα 20. Διάγραμμα με PCA 55
Εικόνα 21. Polynomial feature engineering για n=1, n=3, n=20 56
Εικόνα 22. Διάγραμμα με polynomial feature enginnering 57
Εικόνα 23. Κώδικας για feature importances - ml models 59
Εικόνα 24. Συγκεντρωτικά αποτελέσματα μοντέλων 60
Εικόνα 25. Random Forest - Polynomial Feature Engineering and Tuning – Best Results 62
Εικόνα 26. Κλάσεις αρχικού dataset - count observations per class 64
Εικόνα 27. Κλάσεις dataset με συνθετικά δεδομένα - count observations per class 64
Εικόνα 28. Λογικό διάγραμμα κώδικα ADASYN 65
Εικόνα 29. Αποτελέσματα - train set - test set – Adasyn 66
Εικόνα 30. Confusion Matrix - Adasyn 67
Εικόνα 31. Επιλογή ένωσης κλάσεων μέσω οπτικοποίησης με PCA 70
Εικόνα 32. Custom Classifier - 3 Stages of Classification 71
Εικόνα 33. Αποτελέσματα – Πρώτο Στάδιο – Custom Classifier 72

Εικόνα 34. Πίνακας Σύγχυσης – Πρώτο Στάδιο Ταξινόμησης 73
Εικόνα 35. Αποτελέσματα - Train Set - Test Set- Δεύτερο Στάδιο Ταξινόμησης..... 75
Εικόνα 36. Πίνακας Σύγχυσης - Δεύτερο Στάδιο Ταξινόμησης..... 75
Εικόνα 37. Τρίτο Στάδιο Ταξινόμησης - Pastry Vs Dirtiness 77
Εικόνα 38. Τρίτο Στάδιο Ταξινόμησης – Πίνακας Σύγχυσης - Pastry Vs Dirtiness 78
Εικόνα 39. Τρίτο Στάδιο Ταξινόμησης – Z_Scratch Vs Bumps 79
Εικόνα 40. Τρίτο Στάδιο Ταξινόμησης - Πίνακας Σύγχυσης - Z_Scratch Vs Bumps..... 80
Εικόνα 41. Τρίτο Στάδιο Ταξινόμησης - Πίνακας Σύγχυσης - Z_Scratch Vs Bumps..... 81
Εικόνα 42. Τρίτο Στάδιο Ταξινόμησης - Πίνακας Σύγχυσης – K_Scratch Vs Stains 82
Εικόνα 43. Mutual Information - Pastry Vs Dirtiness..... 84
Εικόνα 44. Mutual Information - Z_Scratch Vs Bumps 85
Εικόνα 45. Mutual Information - K_Scratch Vs Stain..... 85
Εικόνα 46. Λογικό διάγραμμα κώδικα - GBT..... 87
Εικόνα 47. Τύποι ελαττωμάτων χωρίς την Other Faults..... 88
Εικόνα 48. Confusion Matrix - GBT..... 89

ΚΑΤΑΛΟΓΟΣ ΠΙΝΑΚΩΝ

Πίνακας 1. Data Dictionary..... 16
Πίνακας 2. Κλάσεις Ελαττωμάτων..... 18
Πίνακας 3. Μετρικές αποτίμησης μοντέλων μηχανικής μάθησης 30
Πίνακας 4. Πίνακας σύγχυσης για binary classification..... 31
Πίνακας 5. Βασικά στατιστικά μεγέθη 33
Πίνακας 6. Διακύμανση Χαρακτηριστικών 34
Πίνακας 7. Πίνακας Γραμμικών Συσχετίσεων Δεδομένων για τα πρώτα 8 χαρακτηριστικά 37
Πίνακας 8. Feature Importance Score για RF και GBT..... 42
Πίνακας 9. Πρώτες ένδεκα τιμές των συνιστωσών για την κλάση 1..... 47
Πίνακας 10. Νέο σύνολο δεδομένων με 54 χαρακτηριστικά και 7 κλάσεις 50
Πίνακας 11. Πρώτες 10 τιμές των 17 συνιστωσών 54
Πίνακας 12. Χαρακτηριστικά χαμηλής σημαντικότητας σύμφωνα με το RF..... 58
Πίνακας 13. Συγκεντρικός πίνακας χρόνων εκπαίδευσης και βέλτιστων παραμέτρων 61
Πίνακας 14. Φανταστικές Κλάσεις - Δεύτερο Στάδιο Ταξινόμησης..... 74
Πίνακας 15. Τρίτο Στάδιο Ταξινόμησης - Παράμετροι και χρόνοι SVM 76
Πίνακας 16. Συνολικά Αποτελέσματα - Προσαρμοσμένος ταξινομητής 83
Πίνακας 17. GBT - parameters - combinatation and time elapsed..... 88

Κεφάλαιο 1: ΕΙΣΑΓΩΓΗ

1.1 Βιομηχανική Επανάσταση 4.0

Ο κόσμος μας την σημερινή περίοδο βιώνει μία τεχνολογική πρόοδο που αποτελεί ορόσημο της τεχνολογικής εξέλιξης και ανάπτυξης και έχει επικρατήσει να αποκαλείται ως 4η Βιομηχανική Επανάσταση (I.4) ή στα γερμανικά INDUSTRIE 4.0. Εμφανίστηκε για πρώτη φορά στην Γερμανία σαν όρος στην Έκθεση του Ανόβερου το 2011. Η Βιομηχανική Επανάσταση 4ης γενιάς περιλαμβάνει στον πυρήνα της την αυτοματοποίηση, καθώς και την ανταλλαγή πληροφορίας , δεδομένων, εντολών και συστάσεων στα μέσα μαζικής παραγωγής μέσω ασύρματης δικτύωσης, για αυτό και εισάγει μία νέα πραγματικότητα στον τομέα της Παραγωγής και μας οδηγεί σε αυτό που ονομάζεται «έξυπνο εργοστάσιο». Σε ένα «έξυπνο εργοστάσιο» κυβερνό-φυσικά συστήματα συνεργάζονται ώστε να παρακολουθούνται (monitoring) οι φυσικές διεργασίες του εργοστασίου, καθώς άλλα συστήματα τεχνητής νοημοσύνης δημιουργούν συστάσεις (recommendations) για την παραγωγή αποκεντρωμένων αποφάσεων και διαχείρισης. Τα φυσικά συστήματα του εργοστασίου επικοινωνούν και «συνεργάζονται» μεταξύ τους με χρήση Internet of Things συσκευών σε πραγματικό χρόνο. Ουσιαστικά η 4η Βιομηχανική Επανάσταση δημιουργεί και προσκομίζει στην καθημερινότητα και στην Βιομηχανία καινοτομίες που κάνουν χρήση τεχνολογιών κυβερνο-φυσικών συστημάτων (Cyber – Physical Systems), Ψηφιακών Διδύμων (Digital Twins), Βιομηχανικού Διαδικτύου των Πραγμάτων (IIoT) , Υπολογιστικού Νέφους (cloud computing), Τεχνητής Νοημοσύνης (Artificial Intelligence - AI) , Ρομποτικής καθώς και αρκετών άλλων.

Παρακάτω, ακολουθούν βασικά χαρακτηριστικά που θα πρέπει να υπάρχουν σαν προϋπόθεση ώστε ένα σύστημα ή εργοστάσιο να θεωρηθεί ότι ανήκει στην βιομηχανία 4ης γενιάς [1].

Διαλειτουργικότητα: Τα κυβερνο-φυσικά συστήματα συνδέονται και επικοινωνούν μεταξύ τους μέσω δικτύου.

Εικονικότητα: Δημιουργείται ένα εικονικό ψηφιακό αντίγραφο λειτουργίας του συστήματος και των δεδομένων που χρησιμοποιούνται.

Αποκέντρωση: Τα κυβερνο-φυσικά συστήματα μπορούν να λαμβάνουν συστάσεις λειτουργίας, σε πραγματικό χρόνο ή και όχι, μετά από ανάλυση των δεδομένων λειτουργίας καθώς και επεξεργασίας τους από απλά συστήματα συστάσεων ή από συστήματα τεχνητής νοημοσύνης.

Προσαρμοστικότητα: Τα συστήματα αυτά θα πρέπει να μπορούν να προσαρμόζονται σε μεγάλη γκάμα των συνθηκών λειτουργίας. Η χρήση της τεχνητής νοημοσύνης και αλγόριθμων μηχανικής μάθησης δίνει λύσεις, καθώς παρέχει μεγάλη ευελιξία στις λειτουργικότητες του εργοστασίου [1].

Σε κάθε περίπτωση η πρωτοφανής αυτή τεχνολογική εξέλιξη με την Βιομηχανική Επανάσταση 4.0 δημιουργεί έναν νέο χάρτη στην Βιομηχανία με πολλές επιπτώσεις τόσο στην κοινωνία αλλά και σε διεθνές πολιτικό-οικονομικό περιβάλλον. Η χάραξη εθνικών στρατηγικών και σχεδιασμού αρχίζει και γίνεται όλο και πιο αναγκαία για την εκμετάλλευση και την απορρόφηση αυτής της τεχνολογικής προόδου σε κρατικό επίπεδο.

1.2 Προσφορά της Μηχανικής Μάθησης στην Παραγωγή Πλακών Χάλυβα

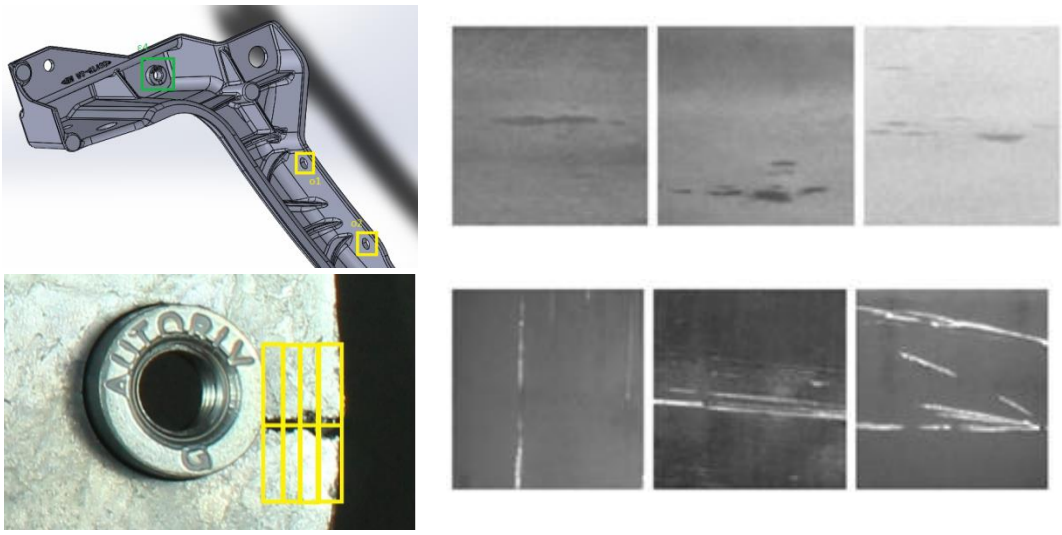
Με την εξέλιξη της Βιομηχανικής Επανάστασης 4.0 έχει δημιουργηθεί μεγάλο ενδιαφέρον στην χρήση μεθόδων τεχνητής νοημοσύνης καθώς και ανάλυσης των δεδομένων στην Παραγωγή και στα εργοστάσια χάλυβα. Το IIoT καθώς και τα Κυβερνο-Φυσικά Συστήματα (Cyber-Physical Systems) σε συνεργασία με συστήματα Τεχνητής Νοημοσύνης είναι μερικά από τα καίρια θέματα ενδιαφέροντος που απασχολούν γενικά τον τομέα της Παραγωγής Χάλυβα στα πλαίσια του κύματος της Βιομηχανικής Επανάστασης 4.0. Ο χάλυβας αποτελεί ένα αναπόσπαστο κομμάτι των κατασκευαστικών εφαρμογών και γενικά, μπορεί να ταξινομηθεί, σχετικά με την χημική του σύσταση και το ποσοστό του σε άνθρακα, σε χάλυβα χαμηλής περιεκτικότητας άνθρακα (low carbon steel με C % = 0.05 -0.25), σε χάλυβα μεσαίας περιεκτικότητας άνθρακα (mild carbon steel με C% = 0.29 to 0.54), σε χάλυβα υψηλής περιεκτικότητας άνθρακα (C% = 0.55 - 0.9) και σε χάλυβα πολύ υψηλής περιεκτικότητας σε άνθρακα (C% = 0.96 to 2.1) [2]. Όπως είναι γνωστό, η περιεκτικότητα σε άνθρακα επηρεάζει άμεσα την αντοχή της κατασκευής καθώς επιδρά στο όριο διαρροής και στο όριο θραύσης του υλικού. Οι βιομηχανικές κατασκευές χαλύβδινων πλακών μπορούν να υποφέρουν από πολλά είδη και κατηγορίες ελαττωμάτων δημιουργώντας στη συνέχεια αστοχίες στις κατασκευαστικές εφαρμογές τους κάτι που μπορεί να έχει πολύ αρνητικές επιπτώσεις. Οι πλάκες χάλυβα που παράγονται σε ένα εργοστάσιο μπορούν να εμφανίζουν αποκλίσεις σε ένα πλήθος παραγόντων όπως στο σχήμα, στην εμφάνιση, στον χρωματισμό, στη δομή, στις διαστάσεις, στην τραχύτητα της επιφάνειας [3], καθώς και άλλων παραγόντων δημιουργώντας ένα προϊόν που δεν καλύπτει τα υπάρχοντα πρότυπα κατασκευής και χρήσης. Μέσω λογισμικών τεχνητής νοημοσύνης και πιο συγκεκριμένα, μηχανικής μάθησης, είναι δυνατή η προσφορά λύσεων στην παρακολούθηση βιομηχανικών εγκαταστάσεων παραγωγής χαλύβδινων πλακών, η βελτίωση και πρόβλεψη της παραγόμενης ποιότητας καθώς και ο έγκαιρος εντοπισμός και αναγνώριση της κατηγορίας ελαττώματος (anomaly detection – type of fault detection) της κάθε χαλύβδινης πλάκας κατά τη διαδικασία παραγωγής, αλλά και της επεξεργασίας [4]. Η μηχανική μάθηση ως ένας κλάδος της τεχνητής νοημοσύνης χρησιμοποιείται από πλήθος βιομηχανιών σε πολλούς διαφορετικούς τομείς με σκοπό την δημιουργία συστημάτων που έχουν την ικανότητα με χρήση μαθηματικών μοντέλων να

μαθαίνουν και να εκπαιδεύονται από τις πληροφορίες (δεδομένα) που επικρατούν στο περιβάλλον του συστήματος, με σκοπό την δημιουργία προβλέψεων καθώς και βέλτιστων συστάσεων λειτουργίας σε μελλοντικά γεγονότα [5]. Για αυτό τον λόγο, οι τεχνικές μηχανικής μάθησης χρησιμοποιούνται ευρέως για τον εντοπισμό μοτίβων, ανωμαλιών, σφαλμάτων και ελαττωμάτων κατά την διάρκεια της παραγωγής διευκολύνοντας έτσι την εργασία του τμήματος Ελέγχου Ποιότητας (Quality Control) και μειώνοντας την πιθανότητα αποστολής προϊόντος με ενδεχόμενες αστοχίες.

Είναι ευκόλως κατανοητό ότι η οπτική επιθεώρηση από εργαζόμενους κατά την παραγωγή των χαλύβδινων πλακών απαιτεί πολύ παραπάνω χρόνο καθώς και ενέχει την πιθανότητα του ανθρωπίνου λάθους που θα μεγαλώνει για μεγαλύτερους αριθμούς πλακών. Επιπροσθέτως, τα κριτήρια επιλογής και της κατηγοριοποίησης μιας χαλύβδινης πλάκας από εργαζόμενους ως ελαττωματικής ή της επιλογής του τύπου του ελαττώματος της χαλύβδινης πλάκας σε συγκεκριμένες περιπτώσεις είναι ασφαλές να υποτεθεί ότι μπορεί να μην είναι καθόλου απλά. Η διαρκής επικινδυνότητα στους χώρους της παραγωγής ενός εργοστασίου δεν βοηθάει καθόλου την παρακολούθηση (monitoring) και την καταγραφή ελαττωμάτων, καθώς η κάθε πλάκα με αμφιλεγόμενο βάρος κάθε φορά θα πρέπει αφού σηκωθεί να τοποθετείται σε κατάλληλο σημείο ώστε να ελέγχεται από όλες της τις πλευρές. Η δημιουργία ατυχημάτων σε επαναλαμβανόμενους ελέγχους είναι κάτι που θα πρέπει και οφείλει το εργοστάσιο και η διοίκηση να λαμβάνει πολύ σοβαρά υπόψιν.

Η παρακολούθηση χαλύβδινων πλακών με τεχνικές μηχανικής μάθησης, έχοντας ως σκοπό τον εντοπισμό των ελαττωμάτων της κάθε πλάκας κατά την παραγωγή, από την εκάστοτε βιομηχανία μπορεί να πραγματοποιηθεί με αρκετούς διαφορετικούς τρόπους. Μία γενική κατηγοριοποίηση που θα μπορούσε να γίνει είναι η παρακολούθηση μέσω φωτογραφιών ή βίντεο των ελασμάτων όπου τα δεδομένα είναι εικόνες [6] και η παρακολούθηση των χαλύβδινων πλακών μέσω δεδομένων που έχουν εξαχθεί μέσω επεξεργασία εικόνας από τις φωτογραφίες των χαλύβδινων πλακών. Στην πρώτη περίπτωση, τα δεδομένα είναι εικόνες, ενώ στην δεύτερη, τα δεδομένα είναι αριθμητικές τιμές. Αυτό, όπως είναι κατανοητό, έχει πολύ μεγάλη επίδραση ως προς την επιλογή της κατηγορίας των κατάλληλων μοντέλων μηχανικής μάθησης που διαλέγει ο μηχανικός ώστε να αξιολογήσει την απόδοση τους και τελικά να ανακαλύψει αυτό που είναι το καλύτερο για το δεδομένο πρόβλημα. Είναι λογικό να υποτεθεί, ως προς την διαδικασία εντοπισμού των ελαττωμάτων με μεθόδους μηχανικής μάθησης σε χαλύβδινες πλάκες, ότι αυτό που είναι το ζητούμενο είναι η γρηγορότερη, πιο άμεση αναγνώριση των σφαλμάτων με την μεγαλύτερη ακρίβεια νωρίς κατά την διαδικασία της παραγωγής. Η επιπλέον εξαγωγή χαρακτηριστικών από φωτογραφίες σίγουρα θα μπορούσε να είναι χρονοβόρα αλλά αυτό θα έπρεπε σε αρχικό επίπεδο να αντιπαρατεθεί με την πολυπλοκότητα των μοντέλων μηχανικής μάθησης που χρησιμοποιούνται και στις δύο περιπτώσεις σε κάθε συγκεκριμένο πρόβλημα. Επιπλέον, η αυτοματοποίηση στα πλαίσια συνεργασίας μεταξύ λογισμικών είναι τόσο αναπτυγμένη που μειώνει τους χρόνους, καθώς θα μπορούσε ένα λογισμικό να κάνει εξαγωγή χαρακτηριστικών από τις φωτογραφίες (image processing) και στην συνέχεια να αποστέλλει τα δεδομένα στο μοντέλο μηχανικής μάθησης που είναι υπεύθυνο για τις συστάσεις. Τα δεδομένα που χρησιμοποιήθηκαν στην συγκεκριμένη μεταπτυχιακή εργασία δίνονται σαν αριθμητικές τιμές μετά από επεξεργασία

εικόνας. Αντικείμενο έρευνας αποτελεί το αν η μετατροπή αριθμητικών δεδομένων σε εικόνες και η χρήση των αντίστοιχων μοντέλων μηχανικής μάθησης (CNNs) μπορεί να δώσει καλύτερα αποτελέσματα σε συγκεκριμένα προβλήματα [7].



Εικόνα 1. Εφαρμογές αυτόματης επιθεώρησης τεμαχίων στο ΕΤΚ-ΕΜΠ με μηχανική μάθηση.

1.3 Στόχος και περιγραφή της εργασίας

Ο στόχος της παρούσας μεταπτυχιακής εργασίας είναι η δημιουργία και η αξιολόγηση διαφορετικών μοντέλων μηχανικής μάθησης για την κατηγοριοποίηση βλαβών και την αναγνώριση διαφορετικών τύπων ελαττωμάτων σε χαλύβδινες πλάκες με την χρήση δεδομένων που έχουν εξαχθεί από φωτογραφίες τους, σε συνεργασία με την υπάρχουσα Βιβλιογραφία. Επιπλέον, χρησιμοποιούνται διάφορες μαθηματικές και στατιστικές τεχνικές για την προ-επεξεργασία των δεδομένων του προβλήματος καθώς και για την αποσαφήνιση και την καλύτερη κατανόησή τους. Επίσης, προσδιορίζονται οι βέλτιστες τιμές των παραμέτρων των μοντέλων που χρησιμοποιήθηκαν. Τα δεδομένα του προβλήματος στην συγκεκριμένη μεταπτυχιακή διατριβή προέρχονται από το ερευνητικό κέντρο Semeion που βρίσκεται στην Ρώμη, Ιταλία. Τα δεδομένα αυτά δημοσιοποιήθηκαν πρώτη φορά το 2010 από το Πανεπιστημιακό Κολλέγιο του Irving (UCI) στην ανοικτή βάση δεδομένων που διαθέτει για προβλήματα μηχανικής μάθησης (machine learning repository) [8]. Στη συνέχεια, ο διαδικτυακός ιστότοπος Kaggle δημοσίευσε το 2017 τα δεδομένα στα πλαίσια ανοικτού διαγωνισμού πάνω σε δεδομένα που αφορούν την Μηχανολογία και την Παραγωγή, όπου νικήτρια ομάδα θα ήταν αυτή που ο ταξινομητής της θα παρουσίαζε την καλύτερη απόδοση [9]. Πέρα από την δημιουργία μοντέλων μηχανικής μάθησης για το συγκεκριμένο πρόβλημα, παράλληλος σκοπός της εργασίας ήταν η γενίκευση των αλγόριθμων που παράχθηκαν με σκοπό την χρήση τους σε προβλήματα αναγνώρισης ελαττωμάτων που τα δεδομένα τους

έχουν εξαχθεί από εικόνες στους τομείς της Βιομηχανικής Παραγωγής και της Μηχανολογίας. Τέλος, για την καλύτερη προσέγγιση της επίλυσης του προβλήματος δημιουργήθηκε η ανάγκη της κατασκευής ενός μοντέλου που θα λειτουργεί σε πολλά επίπεδα κατηγοριοποίησης των βλαβών και αποτελεί και τη σημαντικότερη καινοτομία της συγκεκριμένης μεταπτυχιακής εργασίας.

1.4 Περιγραφή Δεδομένων και Προβλήματος

Ο αριθμός των δεδομένων του συγκεκριμένου προβλήματος το εντάσσουν σε μία κατηγορία μεσαίου μεγέθους προβλήματος μηχανικής μάθησης. Αυτό είχε ως αποτέλεσμα την άσκηση περιορισμού για χρήση κάποιων μοντέλων βαθιάς μηχανικής μάθησης (deep learning) και την προτίμηση για χρήση μοντέλων χαμηλότερης πολυπλοκότητας. Συγκεκριμένα, το σύνολο όλων το δεδομένων είναι 66,994 κελιά (datapoints) – 1941 γραμμές x 34 στήλες. Τα δεδομένα του προβλήματος είναι στο σύνολο τους αριθμητικές τιμές (πέρα των κλάσεων) και αποτελούν χαρακτηριστικά που έχουν εξαχθεί μετά από επεξεργασία εικόνας φωτογραφιών χαλύβδινων πλακών με ελαττώματα. Από τις 34 στήλες οι 27 πρώτες είναι χαρακτηριστικά (features) και οι επόμενες 7 είναι οι τύποι ελαττωμάτων (multi class problem) για κάθε χαλύβδινη πλάκα ή ισοδύναμα, οι κλάσεις του προβλήματος. Η καταγραφή των συγκεκριμένων δεδομένων και η τοποθέτηση τους στις παρακάτω κλάσεις έχει γίνει, όπως έχει αναφερθεί, από το ερευνητικό κέντρο Semeion που βρίσκεται στην Ρώμη, Ιταλία. Δεν υπάρχει συγκεκριμένη αναφορά για τον τρόπο με τον οποίο έχει γίνει η καταγραφή των δεδομένων και η τοποθέτηση τους σε συγκεκριμένες κλάσεις, κάτι που δεν μπορεί με ασφάλεια να αποκλείσει την πιθανότητα ύπαρξης λαθών, αλλά αυτό έτσι και αλλιώς αποτελεί κομμάτι αξιολόγησης από έναν μηχανικό κατά την διάρκεια δημιουργίας μοντέλων μηχανικής μάθησης. Όπως έχει αναφερθεί ο σκοπός είναι η δημιουργία διαφορετικών μοντέλων – ταξινομητών (classifiers) για την σωστή κατηγοριοποίηση των σφαλμάτων βάσει των χαρακτηριστικών τους. Στον Πίνακα 1 παρουσιάζεται το λεξικό των δεδομένων (data dictionary) [10].

Συγκεκριμένα, τα 7 είδη ελαττωμάτων για τις χαλύβδινες πλάκες, έτσι όπως κατηγοριοποιήθηκαν από το ερευνητικό κέντρο Semeion είναι: Pastry, Z scratch, K scratch, Stains , Dirtiness , Bumps and Other Faults. Η τελευταία κλάση (Other Faults) δημιουργήθηκε για την κατηγοριοποίηση χαλύβδινων πλακών με ελάττωμα που δεν μπορούσε να συμπεριληφθεί στις προηγούμενες 6 και από εκεί πήρε και το όνομά της. Υπάρχουν αρκετές αναφορές στο διαδίκτυο για την επίλυση του συγκεκριμένου προβλήματος με 2 κλάσεις (binary classification), Common Faults και Other Faults, όπου η πρώτη κλάση περιλαμβάνει όλες τι προηγούμενες 6 [11]. Επίσης, τα δεδομένα της κλάσης Other faults, όπως καταλαβαίνει κανείς από το όνομα της, μπορεί να περιλαμβάνουν χαρακτηριστικά που δεν είναι καθόλου κοινά μεταξύ τους αφού περιλαμβάνει όλες τις πλάκες με ελάττωμα που δεν μπορούσαν να ενσωματωθούν στις υπόλοιπες 6 κλάσεις. Είναι σαφές πως κάτι τέτοιο αυξάνει το επίπεδο δυσκολίας του προβλήματος.

Πίνακας 1. Data Dictionary

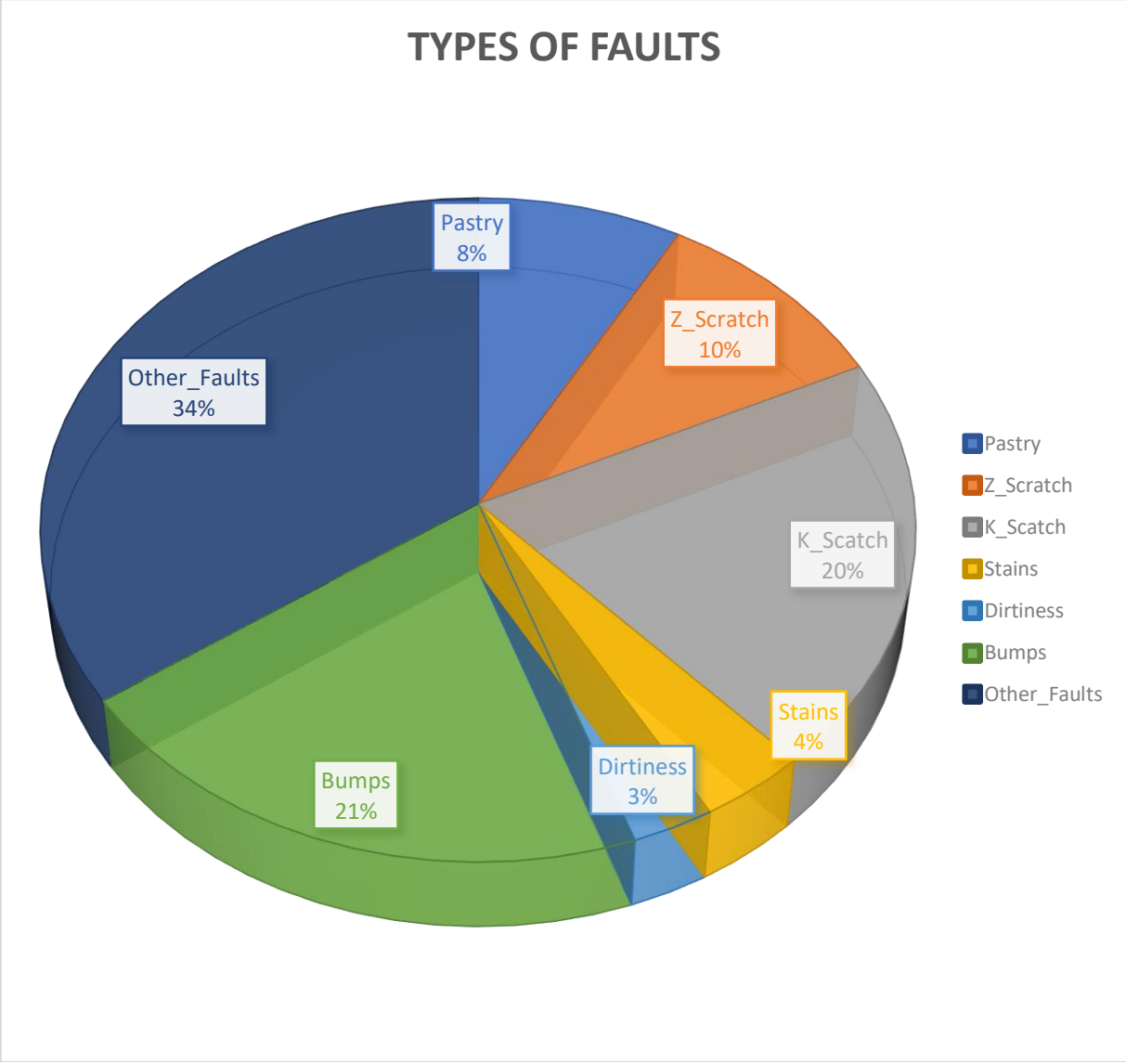
A/A	Attributes	Type	Description
1	X_Minimum	Integer	Minimum horizontal length of the plate from centre
2	X_Maximum	Integer	Maximum horizontal length of the plate from centre
3	Y_Minimum	Integer	Minimum vertical length of the plate from centre
4	Y_Maximum	Integer	Maximum vertical length of the plate from center
5	Pixels_Areas	Integer	Overall Pixel Area
6	X_Perimeter	Integer	Horizontal Perimeter length of the plate
7	Y_Perimeter	Integer	Vertical Perimeter length of the plate
8	Sum_of_Luminosity	Integer	Sum of all rate at which the energy is emitted
9	Minimum_of_Luminosity	Integer	Minimum rate at which the energy is emitted
10	Maximum_of_Luminosity	Integer	Maximum rate at which the energy is emitted
11	Length_of_Conveyor	Integer	The length of the conveyer belt
12	TypeOfSteel_A30	Categorical	‘1’ if the type of steel is A300 else ‘0’
13	TypeOfSteel_A40	Categorical	‘1’ if the type of steel is A400 else ‘0’
14	Steel_Plate_Thick ness	Integer	Thickness of the steel plate
15	Edges_Index	Float	Measure of the index edge of the plate
16	Empty_Index	Float	Measure of the hollowness or emptiness of the plate
17	Square_Index	Float	Measure of Squareness of the plate
18	Outside_X_Index	Float	Measure of Outside horizontal Index
19	Edges_X_Index	Float	Measure of horizonal edges of the plate

20	Edges_Y_Index	Float	Measure of vertical edges of the plate
21	Outside_Global_I ndex	Float	Measure of how much is the global index number
22	LogOfAreas	Float	Measure of the logarithmic value of Area of the plate
23	Log_X_Index	Float	Logarithmic values of horizontal index of the plate
24	Log_Y_Index	Float	Logarithmic values of vertical index of the plate
25	Orientation_Index	Float	Overall orientation measure of the plate
26	Luminosity_Index	Float	Measure of rate at which the energy is emitted
27	SigmoidOfAreas	Float	The measure of Sigmoid function of the area of the plate
28	Pastry	Categorical	‘1’ if the defect is Pastry else
29	Z_Scratch	Categorical	‘1’ if the defect is Z_Scratch else ‘0’
30	K_Scatch	Categorical	‘1’ if the defect is K_Scatch else ‘0’
31	Stains	Categorical	‘1’ if the defect is Stains else
32	Dirtiness	Categorical	‘1’ if the defect is Dirtiness else ‘0’
33	Bumps	Categorical	‘1’ if the defect is Bumps else
34	Other_Faults	Categorical	‘1’ if the defect is Other_Faults else ‘0’

Μία σημαντική ιδιαιτερότητα του προβλήματος, η οποία σε συνδυασμό με τον όχι αρκετά μεγάλο αριθμό δεδομένων που δίνονται και δημιουργεί δυσκολίες, είναι η διαφοροποίηση ως προς τον αριθμό των δεδομένων για κάθε μία από τις 7 κλάσεις (class imbalances). Στο συγκεκριμένο πρόβλημα πολλαπλής κατηγοριοποίησης (multiclass), ο αριθμός των δεδομένων για κάθε κλάση είναι αρκετά διαφορετικός κάτι που δεν λειτουργεί καθόλου θετικά ως προς την εκμάθηση και την εκπαίδευση των μοντέλων μηχανικής μάθησης. Βέβαια, στον πραγματικό κόσμο έχουμε πραγματικά δεδομένα τα οποία δεν μπορούν ποτέ να είναι ιδανικά. Αυτό είναι ένα επιπλέον πρόβλημα των συγκεκριμένων δεδομένων που δυσχεραίνει την απόδοση των μοντέλων μηχανικής μάθησης. Παρακάτω ακολουθεί ο Πίνακας 2 και η Εικόνα 2 με τον αριθμό των δεδομένων για εκμάθηση (training) και για πρόβλεψη και διασταύρωση αποτελεσμάτων (testing) ανά κλάση για το συγκεκριμένο πρόβλημα.

Πίνακας 2. Κλάσεις Ελαττωμάτων

CLASSES	OBSERVATIONS
PASTRY	158
Z SCRATCH	190
K SCRATCH	391
STAINS	72
DIRTINESS	55
BUMPS	402
OTHER FAULTS	673
SUM of observations	1941



Εικόνα 2. Ποσοστά τύπων ελαττωμάτων

1.5 Βιβλιογραφική Ανασκόπηση

Στην παρούσα ενότητα θα αναφερθούν μοντέλα, διαδικασίες – τεχνικές και αποτελέσματα που έχουν δημοσιευτεί από διάφορους ερευνητές που έχουν καταπιαστεί με το συγκεκριμένο πρόβλημα από την στιγμή της δημοσιοποίησης του. Διαφορετικές προσεγγίσεις έγιναν με σκοπό την παραγωγή καλύτερων αποτελεσμάτων αλλά και κατανόησης των δεδομένων που εξήχθησαν από φωτογραφίες χαλύβδινων πλακών. Η παρούσα βιβλιογραφία έχει ως πεδίο αναζήτησης την δημιουργία μοντέλων μηχανικής μάθησης για τον τομέα της Παραγωγής και του συγκεκριμένου προβλήματος που επιλύεται σε αυτή την μεταπτυχιακή εργασία.

1.5.1 Προσέγγιση με απλή εφαρμογή μοντέλων μηχανικής μάθησης

Στα [12] [13] που δημοσιεύτηκαν το 2014 και το 2019 αντίστοιχα δοκιμάστηκε ένα πλήθος μοντέλων μηχανικής μάθησης για το συγκεκριμένο πρόβλημα ώστε να αξιολογηθούν ως προς τα ποσοστά επιτυχίας τους. Δεν πραγματοποιήθηκε κάποια ρύθμιση των παραμέτρων των μοντέλων μηχανικής μάθησης. Επιπλέον, δεν έγινε κάποια προσπάθεια προ-επεξεργασίας των δεδομένων πριν την είσοδο τους στα μοντέλα μηχανικής μάθησης, καθώς και ούτε έγινε κάποια προσπάθεια για επιλογή των κατάλληλων χαρακτηριστικών με σκοπό την επιλογή τους. Επίσης, δεν ερευνήθηκε κάποια μέθοδος για να αντιμετωπιστεί το πρόβλημα των μη σταθμισμένων κλάσεων (class imbalances). Τα αποτελέσματα των μοντέλων επιδεικνύουν το πώς αντιδρά το κάθε μοντέλο μηχανικής μάθησης (classifier) βάσει των μαθηματικών συσχετισμών που το διέπουν καθώς και τον τρόπο λειτουργίας του στο συγκεκριμένο πρόβλημα, κάτι που παρέχει σημαντική πληροφορία (insights) για την καλύτερη επιλογή μοντέλων. Στο [12], σε αντίθεση με το [13], τα δεδομένα που χρησιμοποιήθηκαν για να υπολογιστεί η απόδοση των μοντέλων μέσω πινάκων σύγχυσης (confusion matrix) αναγράφεται ότι είναι αληθινά δεδομένα (actual data) από μετρήσεις. Αυτό μας οδηγεί στο συμπέρασμα ότι τα συγκεκριμένα αποτελέσματα δεν μπορούν να αποτελέσουν μέσο σύγκρισης για την συγκεκριμένη μεταπτυχιακή εργασία όπως και για κάθε άλλη που τα αποτελέσματά της προκύπτουν μόνο από το συγκεκριμένο σύνολο δεδομένων (dataset) που δωρίστηκε από το ερευνητικό κέντρο Semeion της Ρώμης στο U.C.L (2010).

1.5.2 Προσέγγιση με ρύθμιση των παραμέτρων μοντέλων μηχανικής μάθησης

Στο [14] που δημοσιεύτηκε το 2019 χρησιμοποιήθηκαν 3 μοντέλα μηχανικής μάθησης ως προς σύγκριση των αποτελεσμάτων τους στην αναγνώριση των τύπων ελαττώματος σε συνδυασμό με ρύθμιση των παραμέτρων τους (hyperparameter tuning). Για την διαδικασία της ρύθμισης δημιουργήθηκε ένα πλέγμα (grid) των τιμών των παραμέτρων με συγκεκριμένα

όρια για τις παραμέτρους και για το κάθε μοντέλο μηχανικής μάθησης και με σταθερό βήμα εκτελέστηκαν όλες οι επαναλήψεις για όλους τους δυνατούς συνδυασμούς, παράγοντας τα σχετικά αποτελέσματα. Επιλέχθηκαν οι παράμετροι των μοντέλων που έδιναν το μικρότερο σφάλμα και την καλύτερη απόδοση (accuracy) ως προς χρήση. Η βιβλιοθήκη που χρησιμοποιήθηκε για την ρύθμιση των αποτελεσμάτων ονομάζεται GridSearchCV και είναι μία πολύ διαδεδομένη βιβλιοθήκη που έχει χρησιμοποιηθεί σε πλήθος προβλήματων [15] του πακέτου για μηχανική μάθηση scikit της Python.

1.5.3 Προσέγγιση με επιλογή των καλύτερων χαρακτηριστικών βάσει μοντέλων μηχανικής μάθησης

Στο [16] που η δημοσίευση του πραγματοποιήθηκε το 2012 χρησιμοποιήθηκαν τεχνικές επιλογής των καλύτερων χαρακτηριστικών (best features) για το πρόβλημα. Η λογική πίσω από αυτό είναι να γίνεται επιλογή κάποιων χαρακτηριστικών για χρήση στα μοντέλα μηχανικής μάθησης και όχι η χρήση όλων (dimensionality reduction). Τέτοιες τεχνικές είναι σύνηθες να εφαρμόζονται σε προβλήματα τα οποία έχουν μεγάλο αριθμό χαρακτηριστικών καθώς είναι λογικό να υπάρχει μειωμένη ή αυξημένη συσχέτιση μεταξύ χαρακτηριστικών και κλάσεων. Επίσης, μειώνοντας την διάσταση των δεδομένων ο χρόνος επεξεργασίας (computing time) μειώνεται και αυτός. Σε προβλήματα με πολλές κλάσεις, η χρήση τέτοιων τεχνικών μπορεί να δημιουργήσει προβλήματα με την λογική το ότι διαφορετικά χαρακτηριστικά μπορεί να επηρεάζουν ένα τύπο προβλήματος και διαφορετικά έναν άλλο. Για αυτό και η επιλογή των καλύτερων χαρακτηριστικών γίνεται σταθμισμένα, δηλαδή υπολογίζοντας ανάλογα με την τεχνική που χρησιμοποιείται, την συσχέτιση του κάθε χαρακτηριστικού με όλες τις κλάσεις. Έτσι, συγκρίνοντας βάσει συγκεκριμένων κριτηρίων τη συσχέτιση κάθε χαρακτηριστικού με όλες τις κλάσεις επιλέγονται τα χαρακτηριστικά με την μεγαλύτερη συσχέτιση και τα υπόλοιπα δεν λαμβάνονται υπόψιν για είσοδο στο μοντέλο μηχανικής μάθησης. Η επιλογή των καλύτερων χαρακτηριστικών για ένα πρόβλημα μηχανικής μάθησης αποτελεί μεγάλο αντικείμενο έρευνας και ένα πλήθος τεχνικών έχουν δημιουργηθεί. Στην εργασία του 2012 για να εξαχθεί η σημαντικότητα των χαρακτηριστικών χρησιμοποιήθηκαν τρία μοντέλα (logistic regression , c5.0 decision trees, MLPNN) τα οποία μπορούν και υπολογίζουν με διαφορετικούς τρόπους τη σημαντικότητα των χαρακτηριστικών βάσει δικών τους κριτηρίων [17] [18] [19]. Η συμφωνία ως προς την επιλογή των χαρακτηριστικών των μοντέλων (model based feature importances) δεν είναι δεδομένη σε κανένα βαθμό, καθώς τα χαρακτηριστικά που θεωρεί το κάθε μοντέλο ως σημαντικά έχει να κάνει με τον τρόπο λειτουργίας του. Για την συγκεκριμένη δημοσίευση χρησιμοποιήθηκαν 13 χαρακτηριστικά για το μοντέλο c5.0 και 7 χαρακτηριστικά για την λογιστική παλινδρόμηση και το MLPNN.

Το μοντέλο C5.0 DT παρουσίασε πολύ καλύτερα αποτελέσματα από τα υπόλοιπα. Το συγκεκριμένο μοντέλο [20] [21] δεν δίνεται σαν λογισμικό ανοικτού κώδικα (open source) από την Python και το scikit [22] καθώς για την χρήση του απαιτείται ιδιωτική άδεια (proprietary license). Αυτό δημιουργεί προβλήματα και δυσκολίες στην χρήση του, καθώς απαιτείται η μοντελοποίηση (custom model) εξαρχής του συγκεκριμένου [23].

1.5.4 Προσέγγιση με αφαίρεση ακραίων τιμών και εφαρμογή μοντέλων μηχανικής μάθησης

Στο [24] θεωρήθηκε η ύπαρξη ακραίων τιμών (outliers) στα δεδομένα του προβλήματος και πραγματοποιήθηκε η αφαίρεση τους. Μία καλή πρακτική για την αφαίρεση ακραίων τιμών, όταν πραγματοποιείται, είναι ο αριθμός τους να μην ξεπερνά το 5% των συνολικών δεδομένων, καθώς κανείς δεν μπορεί με ασφαλή τρόπο να υποθέσει ότι τα δεδομένα που αφαιρούνται δεν είναι αντιπροσωπευτικά. Η τεχνική για την ανίχνευση των ακραίων τιμών έγινε με μη επιβλεπόμενη μηχανική μάθηση για την ομαδοποίηση των δεδομένων (clustering) και την εμφάνιση στατιστικών ανωμαλιών (unsupervised learning for anomaly detection). Αναφορά σε διαφορετική εφαρμογή της συγκεκριμένης τεχνικής γίνεται στο [25]. Στη συνέχεια, αφού αφαιρέθηκαν κάποιες τιμές (records), δοκιμάστηκαν 4 μοντέλα μηχανικής μάθησης για την ταξινόμηση των ελαττωμάτων.

1.5.5 Σύνθετες προσεγγίσεις

Στο [26] που βγήκε το 2015 αν και δεν υπήρξε μεγάλος αριθμός μοντέλων μηχανικής μάθησης που αναπτύχθηκαν για την ταξινόμηση, καθώς χρησιμοποιήθηκαν μόνο Support Vector Machines (SVMs), εφαρμόστηκε ένα πλήθος τεχνικών και για την αντιμετώπιση του προβλήματος. Αρχικά, χρησιμοποιήθηκε και τεχνική μείωσης της διάστασης των χαρακτηριστικών και επιλογή αυτών με την μεγαλύτερη σημαντικότητα για την παραγωγή και την δημιουργία αποτελεσμάτων αλλά και βγήκαν αποτελέσματα με χρήση όλων των δεδομένων. Αυτό είναι σημαντικό, καθώς η έρευνα θα μπορούσε να δώσει πληροφορίες (insights) σε μελλοντικούς ερευνητές για το αν είναι χρήσιμη η συγκεκριμένη επιλογή καλύτερων χαρακτηριστικών (feature importances) στο παρόν πρόβλημα. Πιο συγκεκριμένα, χρησιμοποιήθηκε για την επιλογή των καλύτερων χαρακτηριστικών η τεχνική αναδρομικής αφαίρεσης χαρακτηριστικών (Recursive Feature Elimination) [27]. Κατά αυτή την τεχνική, το μοντέλο SVM υπολογίζει ένα δείκτη σημαντικότητας και κατατάσσει τα χαρακτηριστικά κατά φθίνουσα σειρά. Αυτό επαναλαμβάνεται για μια σειρά επαναλήψεων, όπου κάθε φορά αφαιρείται το χαρακτηριστικό με το μικρότερο δείκτη σημαντικότητας και έως ότου μείνει ένας προεπιλεγμένος αριθμός χαρακτηριστικών. Τα αποτελέσματα της RFE με σκοπό τον υψηλότερο δείκτη απόδοσης (accuracy) οδήγησαν στην χρήση των 13 πρώτων χαρακτηριστικών. Επίσης, δημιουργήθηκε κώδικας για την αντιμετώπιση του προβλήματος της διαφοροποίησης των αριθμών των παρατηρήσεων της κάθε κλάσης (class imbalances). Αυτό που έγινε ήταν η χρησιμοποίηση επαναλαμβανόμενων παρατηρήσεων (observations) μέχρι να φτάσει ο αριθμός των δεδομένων να είναι ίδιος για κάθε κλάση (sample size balancing). Δηλαδή, χρησιμοποιήθηκαν επαναλαμβανόμενα δεδομένα για κάθε κλάση μέχρι οι παρατηρήσεις σε κάθε κλάση να είναι ίδιες με τις παρατηρήσεις της μεγαλύτερης (Other Faults). Τέλος, εφαρμόστηκαν τεχνικές βελτιστοποίησης (tuning) για το μοντέλο μηχανικής μάθησης με Γενετικούς Αλγόριθμους (Genetic Algorithms – GA) [28] και Βελτιστοποίησης Σμήνους Σωματιδίων (Particle Swarm Optimization – PSO) [29] αλλάζοντας το χώρο λύσεων

των παραμέτρων του μοντέλου. Η συγκεκριμένη έρευνα παρουσιάζει ενδιαφέρον καθώς είναι η πρώτη στην οποία έγινε προσπάθεια αντιμετώπισης των μη σταθμισμένων κλάσεων.
Στο [30] που δημοσιεύτηκε το 2020 παρουσιάζεται ένα μεγάλο εύρος τεχνικών και μοντέλων μηχανικής μάθησης. Αρχικά υλοποιούνται μέθοδοι για την επιλογή των πιο σημαντικών χαρακτηριστικών. Επίσης, παρουσιάζονται τρεις μέθοδοι για την αντιμετώπιση των μη σταθμισμένων κλάσεων. Ύστερα γίνεται αντιγραφή με τυχαίο τρόπο δεδομένων των κλάσεων με μειωμένο αριθμό παρατηρήσεων μέχρι τα δείγματα να φτάσουν τον αριθμό της μέγιστης κλάσης (random oversampling). Στη συνέχεια πραγματοποιείται μείωση όλων των δεδομένων των κλάσεων μέχρι αυτά να φτάσουν τις παρατηρήσεις της κλάσης με τον χαμηλότερο αριθμό παρατηρήσεων (random undersampling). Ακολουθεί μία τεχνική που ονομάζεται SMOTE (Synthetic Minority Oversampling Technique) [31] η οποία παράγει συνθετικά δεδομένα προσπαθώντας να μην αλλοιώσει τις κατανομές των δεδομένων και όλες οι κλάσεις να έχουν τον ίδιο αριθμό παρατηρήσεων. Το πρόβλημα των μη σταθμισμένων κλάσεων στους τύπους των ελαττωμάτων των χαλύβδινων πλακών δημιουργεί δυσκολίες στα μοντέλα μηχανικής μάθησης να εκπαιδευτούν ως προς τις κλάσεις με τον μικρότερο αριθμό παρατηρήσεων (minority class) καθώς επηρεάζεται σε μεγάλο βαθμό από το μεγαλύτερο αριθμό παρατηρήσεων των άλλων κλάσεων και σαφώς και αυτής της κλάσης με τις περισσότερες παρατηρήσεις (majority class) [32]. Αποτελέσματα παράγονται για όλα τα μοντέλα μηχανικής μάθησης που χρησιμοποιήθηκαν μετά τις τρεις αυτές τεχνικές για αξιολόγηση και σύγκριση. Τέλος δημιουργείται ένας προσαρμοσμένος ταξινομητής με βάση τις τεχνικές XAI-Explainable Artificial Intelligence methods, Association Rules of RF και Association Rules of Mining, τα αποτελέσματα του οποίου συγκρίνονται με τα αποτελέσματα του μοντέλου μηχανικής μάθησης (Random Forest) με την καλύτερη απόδοση μετά από βελτιστοποίηση παραμέτρων (tuning) για το αρχικό σύνολο δεδομένων καθώς και για το σύνολο δεδομένων που προέκυψε μετά από παραγωγή συνθετικών δεδομένων (SMOTE).

Κεφάλαιο 2: ΜΗΧΑΝΙΚΗ ΜΑΘΗΣΗ

Ο σκοπός του παρόντος κεφαλαίου είναι η ανάλυση των κύριων εργαλείων που χρησιμοποιήθηκαν για το πρόβλημα της ταξινόμησης των τύπων των ελαττωμάτων, όπως η γλώσσα προγραμματισμού για την ανάπτυξη κώδικα, το notebook που γράφτηκε ο κώδικας , το σύστημα για την εισαγωγή πακέτων κώδικα, οι βιβλιοθήκες αλλά και τα μοντέλα μηχανικής μάθησης που επιλέχθηκαν για αξιολόγηση. Εδώ, θα πρέπει να αναφερθεί ότι η υλοποίηση των αλγορίθμων έγινε πάνω σε έναν Η/Υ με επεξεργαστή Core i7 και μέγεθος μνήμης 8G RAM.

2.1 Εργαλειοθήκη

2.1.1 Γλώσσα Προγραμματισμού και Εργαλεία

Η γλώσσα προγραμματισμού για την εκπόνηση της μεταπτυχιακής εργασίας ήταν η Python 3.8. Η Python είναι μία ανοικτού λογισμικού , γενικού σκοπού και υψηλού επιπέδου γλώσσα, η οποία δημιουργήθηκε από τον Ολλανδό Guido van Rossum το 1989 [33]. Ο κώδικας για όλα τα μοντέλα γράφτηκε στο ανοικτού λογισμικού notebook Jupyter που δημιουργήθηκε από τον Κολομβιανό-Αμερικανό Fernardo Perez το 2015. Το Jupyter αποτελεί ένα διαδικτυακό διαδραστικό υπολογιστικό περιβάλλον (web-based interactive computational environment) και θεωρείται ένα από τα πιο διαδεδομένα πλέον notebooks για την συγγραφή Python κώδικα για εφαρμογές μηχανικής μάθησης και επιστήμης των δεδομένων [34]. Επιπλέον, έγινε η χρήση του συστήματος ανοικτού κώδικα Anaconda, το οποίο αποτελεί μια διανομή για Python και R κώδικα, το οποίο βοηθάει στην αλλαγή περιβάλλοντος για την εισαγωγή κώδικα (code environment management) καθώς και στην εισαγωγή πακέτων και βιβλιοθηκών (package management and deployment). Το Anaconda δημιουργήθηκε από τους Peter Wang και Travis Oliphant το 2012 και αποτελεί ίσως την πιο διαδεδομένη πλατφόρμα παγκοσμίως για μηχανική μάθηση και υπολογιστική ανάλυση δεδομένων [35].

2.1.2 Κύριες Βιβλιοθήκες

Για την παραγωγή των μοντέλων μηχανικής μάθησης έγινε χρήση πολλών βιβλιοθηκών της Python καθώς οι απαιτήσεις για την προ-επεξεργασία των δεδομένων με διάφορες τεχνικές, την οπτικοποίηση τους καθώς και την παραγωγή των μοντέλων που να δίνουν ικανοποιητικά αποτελέσματα υπήρξαν αρκετά υψηλές. Παρακάτω γίνεται αναφορά μόνο σε βασικά-θεμελιώδη πακέτα της Python που χρησιμοποιήθηκαν για την συγκεκριμένη εργασία. Αυτά είναι τα Numpy [36], Pandas [37], Scikit-learn [38] . Βιβλιοθήκες όπως Tensor-Flow [39], Keras [40] και PyTorch [41] αν και αρχικά υπήρξε το σκεπτικό να χρησιμοποιηθούν τελικά

δεν χρησιμοποιήθηκαν, καθώς εφαρμόζονται κυρίως για βαθιά μηχανική μάθηση (deep learning) κάτι που έρχεται σε αντίθεση με το σχετικά μεσαίο προς μικρό αριθμό των δεδομένων του συγκεκριμένου προβλήματος. Αξίζει να αναφερθεί ότι πολλές φορές προέκυψε πρόβλημα συμβατότητας ανάμεσα στην έκδοση της Python 3.8 που χρησιμοποιήθηκε με διάφορες βιβλιοθήκες καθώς αυτές δεν είχαν ενημερωθεί και δεν μπορούσαν να λειτουργήσουν στην συγκεκριμένη έκδοση. Για αυτό πραγματοποιήθηκε η αλλαγή περιβάλλοντος της γλώσσας προγραμματισμού για την συγγραφή κάποιων scripts.

2.1.3 Numpy

Η Numpy είναι βασική βιβλιοθήκη και μία από τις πιο διαδεδομένες στην επιστημονική κοινότητα για την χειραγώγηση πολυδιάστατων δεδομένων, καθώς και για την εκτέλεση μαθηματικών πράξεων στα δεδομένα. Χρησιμοποιεί πίνακες και γραμμική άλγεβρα και είναι «ελαφριά» βιβλιοθήκη, καθώς ο τρόπος που είναι γραμμένη της επιτρέπει την γρήγορη εκτέλεση μαθηματικών πράξεων. Βασικές λειτουργίες της είναι η εκτέλεση μαθηματικών και λογικών πράξεων στα δεδομένα, ο μετασχηματισμός των δεδομένων, η επιλογή κελιών και παρατηρήσεων, η αφαίρεση και η πρόσθεση δεδομένων, ο μετασχηματισμός Fourier , η εκτέλεση γραμμικής άλγεβρας και στατιστικών μετασχηματισμών και γενικά η χειραγώγηση πολυδιάστατων πινάκων (n- dimensional arrays).

Εικόνα 3.Logo Numpy

2.1.4 Pandas

Η Pandas είναι μία βιβλιοθήκη που χρησιμοποιείται κυρίως για χειραγώγηση των δεδομένων και για ανάλυση (data manipulation and analysis). Η χρήση της γίνεται πριν το μοντέλο μηχανικής μάθησης εκπαιδευτεί. Ουσιαστικά παρέχει τεράστια ευκολία στον χειρισμό χρονοσειρών αλλά και πολυδιάστατων δεδομένων καθώς δουλεύει με πλαίσια δεδομένων (dataframes). Μερικές ενδεικτικές βασικές λειτουργίες της συγκεκριμένης βιβλιοθήκης είναι η αλλαγή διαστάσεων και η επεξεργασία dataframes, η ένωση (merge) και ο χωρισμός dataframes, η εύρεση χαμένων τιμών, το φιλτράρισμα δεδομένων (data filtration) καθώς και άλλες. Πρόκειται για μία από τις βασικές βιβλιοθήκες για την επεξεργασία των δεδομένων καθώς η μετατροπή τους σε dataframes δίνει την δυνατότητα για την χειραγώγησή τους με τον επιθυμητό τρόπο . Το διάγραμμα από την βάση δεδομένων της Google για την χρήση της άνωθεν βιβλιοθήκης μπορεί να βρεθεί εδώ [42].

Εικόνα 4. Logo Pandas

2.1.5 Scikit- learn

Η scikit – learn αποτελεί μία από τις πιο γνωστές και πολυχρησιμοποιημένες βιβλιοθήκες της Python για μηχανική μάθηση. Μερικές ενδεικτικές πληροφορίες είναι ότι περιλαμβάνει μεγάλο αριθμό μοντέλων επιβλεπόμενης (supervised) και μη επιβλεπόμενης (unsupervised) μηχανικής μάθησης για ταξινόμηση (classification), παλινδρόμηση (regression), ομαδοποίηση (clustering), μείωση διάστασης προβλήματος (dimensionality reduction), επιλογή μοντέλων και ένωσή τους (model selection and pipelines), προ-επεξεργασία δεδομένων (data preprocessing and modeling), καθώς και εργαλεία για την οπτικοποίησή τους και ανάλυσή τους (visualization and data analysis). Αποτελεί ουσιαστικά μία πολύ ευέλικτη και εύχρηστη βιβλιοθήκη για την ανάπτυξη και την επεξεργασία μοντέλων μηχανικής μάθησης. Περιλαμβάνει και μοντέλα για βαθιά μηχανική μάθηση (deep learning) αν και συνήθως γίνεται προτίμηση άλλων βιβλιοθηκών καθώς κυρίως το scikit χρησιμοποιείται για ρηχή μηχανική μάθηση (shallow machine learning).

Εικόνα 5. Logo Scikit

2.2 Μοντέλα Μηχανικής Μάθησης

Στην συγκεκριμένη ενότητα γίνεται περιγραφή καθώς και σχετικές αναφορές για τα μοντέλα επιβλεπόμενης μηχανικής μάθησης (supervised machine learning models) που χρησιμοποιήθηκαν στα πλαίσια της εκπόνησης της συγκεκριμένης μεταπτυχιακής εργασίας. Οι τύποι (labels) των ελαττωμάτων δίνονταν για όλες τις παρατηρήσεις (observations). Τα μοντέλα χρησιμοποιήθηκαν μέσω της βιβλιοθήκης scikit της Python και παρουσιάζονται στην συνέχεια.

2.2.1 Logistic Regression

Η λογιστική παλινδρόμηση είναι ένα πολύ γνωστό και απλό μοντέλο, το οποίο αποτελεί εξέλιξη και γενίκευση της γραμμικής παλινδρόμησης [43]. Είναι μία μη γραμμική τεχνική

παλινδρόμησης που χρησιμοποιείται και για προβλήματα ταξινόμησης. Γενικά η εφαρμογή της λογιστικής παλινδρόμησης χρησιμοποιείται για την ταξινόμηση δύο κλάσεων (binary classification). Ουσιαστικά, πρόκειται για μια στατιστική τεχνική που υπολογίζει μία πιθανότητα με βάση την οποία γίνεται η ταξινόμηση ανάμεσα σε δύο κλάσεις (y=0 ή y=1) που αποτελούν και την εξαρτημένη μεταβλητή του προβλήματος. Για τον υπολογισμό αυτής της πιθανότητας και κατ' επέκταση την ταξινόμησης της κλάσης, χρησιμοποιείται μια σιγμοειδής συνάρτηση, η οποία λαμβάνει στην είσοδο της οποιαδήποτε τιμή και βγάζει στην έξοδό της τιμές από 0 μέχρι 1 (sigmoid function S). Για την εφαρμογή της στο συγκεκριμένο πρόβλημα πολλαπλών κλάσεων χρησιμοποιήθηκε η τεχνική One vs All (OvR) που παρέχεται από το scikit [44]. Χρησιμοποιώντας αυτή την τεχνική, το πρόβλημα πολλαπλών κλάσεων χωρίζεται σε πολλά δυαδικά. Ο αλγόριθμος κάνει την ταξινόμηση κάθε κλάσης εναντίον όλων των άλλων. Στο παρόν πρόβλημα είναι σαφές ότι έχουμε 7 δυαδικές ταξινομήσεις (μία για κάθε κλάση) για την παραγωγή αποτελεσμάτων.

2.2.2 K-Nearest Neighbor (k-NN)

Ο αλγόριθμος μηχανικής μάθησης K-NN αποτελεί έναν απλό αλγόριθμο που ονομάζεται αλγόριθμος πλησιέστερου γείτονα. Χρησιμοποιείται τόσο για προβλήματα παλινδρόμησης όσο και ταξινόμησης. Σε μεγάλο αριθμό δεδομένων είναι αρκετά αργός για αυτό και ανήκει στους αλγόριθμους lazy learning [45]. Σε αυτόν τον αλγόριθμο η κλάση για τα νέα δεδομένα που προβλέπεται βγαίνει βάσει της κλάσης που είχαν οι πιο κοντινές παρατηρήσεις κατά την εκπαίδευση του αλγορίθμου. Το αποτέλεσμα δηλαδή και η κλάση της νέας παρατήρησης παράγεται από την πιο συχνά εμφανιζόμενη κλάση ανάμεσα στους γείτονες της. Υπάρχουν πολλοί διαφορετικοί τρόποι μέτρησης της απόστασης της παρατήρησης που γίνεται η πρόβλεψη με τους γείτονες της. Μερικές ενδεικτικές αποστάσεις είναι η Ευκλείδεια απόσταση, Manhattan, Chebyshec και η City block. Το k στο όνομα του αλγορίθμου, που αποτελεί και τη βασική παράμετρό του, αναπαριστά τον αριθμό των γειτόνων που λαμβάνονται υπόψιν κατά την ταξινόμηση μίας νέας παρατήρησης [46]. Η εφαρμογή του συγκεκριμένου μοντέλου που υλοποιήθηκε καθώς και περιγραφή των παραμέτρων του βρίσκεται στην αναφορά [47].

2.2.3 Support Vector Machines

Τα SVMs είναι από τα πιο γνωστά μοντέλα επιβλεπόμενης μάθησης, γνωστά ως Μηχανές Διανυσμάτων Υποστήριξης [48]. Το συγκεκριμένο μοντέλο έχει εφαρμογή κυρίως σε προβλήματα κατηγοριοποίησης (Support Vector Classification-SVC), αλλά μπορεί να γενικευτεί και για προβλήματα παλινδρόμησης (Support Vector Regression-SVR). Η κατηγοριοποίηση των δεδομένων πραγματοποιείται με την δημιουργία ενός κατάλληλου υπέρ-επιπέδου (hyperplane), το οποίο ορίζεται με την εύρεση των κατάλληλων διανυσμάτων (support vectors), με σκοπό τον διαχωρισμό τους βάσει της κλάσης τους (label). Το υπέρ–επίπεδο κάνει χρήση ενός μέγιστου περιθωρίου (maximum margin) για τον διαχωρισμό. Εφαρμόζονται διαφορετικές συναρτήσεις πυρήνα (kernel functions) για την απεικόνιση των

δεδομένων σε άλλους χώρους στην περίπτωση που αυτά δεν είναι γραμμικά διαχωρίσιμα [49]. Το συγκεκριμένο μοντέλο είναι ένας δυαδικός ταξινομητής (binary classification) και για την εφαρμογή του υλοποιήθηκε η γενίκευση OvR που παρέχεται από το scikit [50].

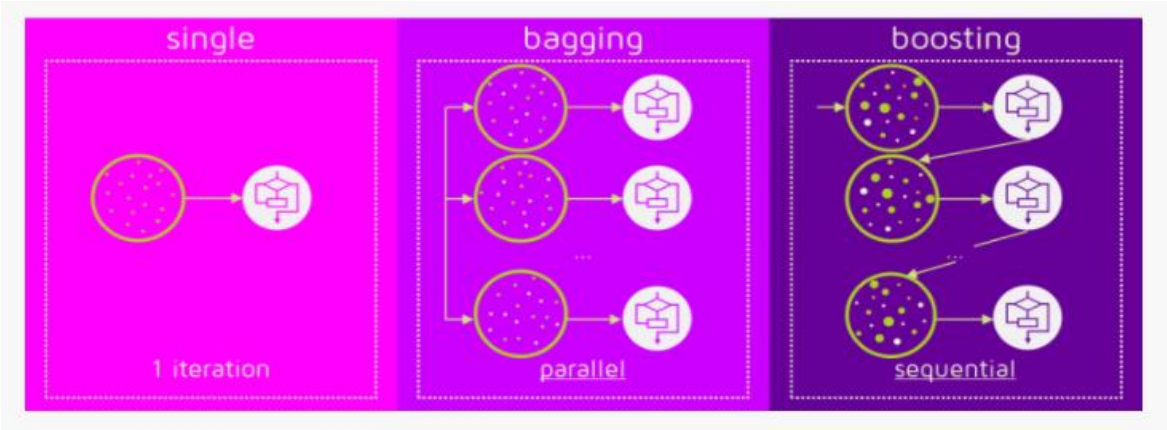
2.2.4 Random Forest

Το μοντέλο τυχαίου δάσους (Random Forest – RF) είναι ένα μοντέλο μηχανικής μάθησης που ανήκει στην κατηγορία της συλλογικής μάθησης (ensemble learning) και δημιουργήθηκε από τον Leo Breiman [51]. Μπορούμε να σκεφτούμε ότι τα μοντέλα συλλογικής μάθησης είναι μοντέλα που συνεργάζονται για την επίτευξη της καλύτερης δυνατής απόδοσης. Ο αλγόριθμος RF χρησιμοποιεί πολλούς αριθμούς δέντρων απόφασης (Decision Trees - DTs) κάτι που αποτελεί και μία από τις βασικές παραμέτρους του μοντέλου [52]. Για την χρήση πολλών δέντρων απόφασης απαιτείται και η χρήση πολλών συνόλων δεδομένων εκπαίδευσης (training sets) και κάθε δέντρο απόφασης εκπαιδεύεται στο αντίστοιχο σύνολο. Αυτό πραγματοποιείται χρησιμοποιώντας δειγματοληψία bootstrap (bootstrap sampling) για την τυχαία συλλογή δειγμάτων από το αρχικό σύνολο των δεδομένων εκπαίδευσης. Ουσιαστικά επιλέγονται από το αρχικό σύνολο εκπαίδευσης τυχαία δείγματα με εναπόθεση. Αυτά τα δείγματα αποτελούν τα νέα δεδομένα εκπαίδευσης για κάθε δέντρο απόφασης. Κάθε δέντρο απόφασης πραγματοποιεί και την δική του ταξινόμηση – ψηφίζει την συγκεκριμένη κλάση (vote). Στο τέλος επιλέγεται η κλάση με τις περισσότερες ψήφους. Γενικά το μοντέλο RF μπορεί να εφαρμοστεί και για προβλήματα ταξινόμησης και παλινδρόμησης και λόγω των πολλών δέντρων που χρησιμοποιούνται το σφάλμα γενίκευσης (generalization error) περιορίζεται, με αποτέλεσμα να μην εμφανίζεται εύκολα υπέρ-προσαρμογή (overfit) του μοντέλου. Γενικά τα μοντέλα συλλογικής μηχανικής μάθησης που εφαρμόζουν bootstrap ονομάζονται και bagging ensembles. Για την παρούσα εργασία το μοντέλο που εφαρμόστηκε, καθώς και οι παράμετροι του, μπορούν να βρεθούν στην αναφορά [53].

2.2.5 Gradient Boosting Trees

Ο αλγόριθμος GBT παρουσιάστηκε πρώτα από τον Friedman [54] και εφαρμόζεται ευρέως σε προβλήματα ανίχνευσης ανωμαλιών (anomaly detection). Αρχικά σχεδιάστηκε για προβλήματα παλινδρόμησης, αλλά στην συνέχεια γενικεύτηκε και για προβλήματα ταξινόμησης. Ανήκει στην κατηγορία των συλλογικών μοντέλων και χρησιμοποιεί ενίσχυση (boosting) με δέντρα απόφασης. Η ενίσχυση αποτελεί μία τεχνική όπου πολλά απλά μοντέλα (weak learners) συνδυάζονται μαζί εν σειρά για την κατασκευή ενός σύνθετου μοντέλου μηχανικής μάθησης. Κάθε νέο μοντέλο που δημιουργείται, εκπαιδεύεται πάνω στα σφάλματα πρόβλεψης των προηγούμενων με σκοπό την ελαχιστοποίηση της συνάρτησης σφάλματος (loss function). Βασική παράμετρος του μοντέλου είναι η επιλογή του αριθμού των σταδίων ενίσχυσης (n-estimators-boosting stages), δηλαδή πόσες φορές θα πραγματοποιηθεί το boosting που αποτελεί και αντίστοιχα τον αριθμό των μοντέλων που θα χρησιμοποιηθούν και στον συγκεκριμένο αλγόριθμο τον αριθμό των Δέντρων Απόφασης. Λόγω του ότι κάθε νέο μοντέλο πρέπει να εκπαιδευτεί σε ξεχωριστό χρόνο από το προηγούμενο, η εκπαίδευση για πολλά στάδια ενίσχυσης μπορεί να είναι μία χρονοβόρα διαδικασία. Η υλοποίηση που

έγινε για την ταξινόμηση των ελαττωμάτων των χαλύβδινων πλακών από το scikit μπορεί να βρεθεί στην αναφορά [55].



Εικόνα 6. Bagging vs Boosting

2.3 Ρύθμιση Παραμέτρων Μοντέλων Μηχανικής Μάθησης

Εκτός από την εφαρμογή των παραπάνω αλγόριθμων μηχανικής μάθησης χρησιμοποιήθηκε τεχνική ρύθμισης των παραμέτρων τους (hyperparameter tuning). Πιο συγκεκριμένα, έγινε η χρήση της ρουτίνας GridSearchCV (Grid Search Cross Validation) [56] που παρέχεται από το scikit και αποτελεί μία πολύ χρήσιμη τεχνική ρύθμισης, αλλά για περιορισμένο αριθμό δεδομένων. Σε μεγάλα σύνολα δεδομένων, η συγκεκριμένη μέθοδος είναι αρκετά αργή για αυτό και θεωρείται μη πρακτική η χρήση της. Ο λόγος που είναι αργή είναι επειδή δημιουργεί ένα πλέγμα (grid) με τις τιμές των παραμέτρων καθώς εκπαιδεύει επαναλαμβανόμενα το μοντέλο για όλους τους δυνατούς συνδυασμούς των τιμών αυτών. Οι παράμετροι που δίνουν το καλύτερο αποτέλεσμα είναι και αυτές που στο τέλος αποθηκεύονται. Έτσι, βελτιστοποιείται η απόδοση του μοντέλου μηχανικής μάθησης, καθώς καθορίζονται οι παράμετροι που δίνουν την υψηλότερη απόδοση. Σε ένα μοντέλο μηχανικής μάθησης με 4 παραμέτρους που κάθε παράμετρος μπορεί να πάρει 10 τιμές ο αριθμός των συνδυασμών είναι 10x10x10x10= 10,000. Αυτό σημαίνει ότι το μοντέλο που χρησιμοποιείται θα εκπαιδευτεί 10,000 φορές και στην συνέχεια θα βρεθεί αυτό με την καλύτερη απόδοση και θα επιλεχθούν οι παράμετροι που αντιστοιχούν σε αυτή. Επίσης, πέρα από αυτούς τους συνδυασμούς, αν εφαρμόζεται δισταυρούμενη επικύρωση (cross validation) κατά τη ρύθμιση, οι συνδυασμοί αυξάνονται παραπάνω. Η εφαρμογή της διασταυρούμενης επικύρωσης είναι συνήθης και απαραίτητη, καθώς οι παράμετροι του μοντέλου που προσδιορίζονται ως βέλτιστες δημιουργούν πιο γενικευμένα αποτελέσματα. Στο παραπάνω παράδειγμα για μια τυπική τιμή cv=10 (10 folds) θα έχουμε 10x10,000=100,000 συνδυασμούς. Αν ληφθεί υπόψιν ένα μεγάλο σύνολο δεδομένων, καθώς και η πολυπλοκότητα του κάθε μοντέλου που χρησιμοποιείται, τότε ο μεγάλος αριθμός συνδυασμών που δοκιμάζει η GridSearchCV ώστε να κάνει fit δημιουργεί δυσκολίες ως προς τον υπολογιστικό χρόνο. Επειδή το σύνολο των δεδομένων στην παρούσα εργασία δεν είναι τόσο μεγάλο (1941 παρατηρήσεις) προτιμήθηκε να χρησιμοποιηθεί η συγκεκριμένη τεχνική καθώς το μεγάλο της πλεονέκτημα της σε σχέση με άλλες τεχνικές ρύθμισης είναι ότι τρέχει όλες τις τιμές των

παραμέτρων στο εύρος που της δίνεται και δεν διαλέγει κάποιους συνδυασμούς επιλεκτικά, όπως για παράδειγμα γίνεται στην τεχνική RandomizedCV, παράγοντας έτσι το βέλτιστο αποτέλεσμα. Επίσης, σύγκριση της τεχνικής GridSearchCV με άλλες, όπως για παράδειγμα RandomizedCV, Γενετικοί Αλγόριθμοι, μπορεί να βρεθεί στην αναφορά [57] . Στην παρούσα εργασία για όλα τα μοντέλα μηχανικής μάθησης που δοκιμάστηκαν πραγματοποιήθηκε ρύθμιση των παραμέτρων τους. Οι βέλτιστες παράμετροι, ο αριθμός των συνδυασμών, καθώς και ο χρόνος που απαιτήθηκε αναφέρονται στα αποτελέσματα του κάθε Κεφάλαιου.

2.4 Αξιολόγηση Μοντέλων

Στην συγκεκριμένη μεταπτυχιακή εργασία υλοποιήθηκαν 22 πειράματα - σενάρια μοντέλων μηχανικής μάθησης. Το μεγάλο πλήθος μοντέλων και τεχνικών επεξεργασίας των δεδομένων που δημιουργήθηκε παρέχει σημαντική πληροφορία σχετικά με το πρόβλημα των ελαττωμάτων των χαλύβδινων πλακών, καθώς δημιουργεί μία αναλυτική εικόνα της απόδοσης τους. Για όλα τα μοντέλα που δοκιμάστηκαν ως προς σύγκριση μαζί με τις μεθόδους προ-επεξεργασίας των δεδομένων τα αποτελέσματα παρουσιάζονται αναλυτικά σε κάθε κεφάλαιο. Τα αποτελέσματα των μοντέλων συγκρίθηκαν παράγοντας για κάθε ένα από αυτά τους αντίστοιχους πίνακες σύγχυσης (confusion matrix). Παρακάτω γίνεται αναφορά στα χαρακτηριστικά ενός πίνακα σύγχυσης. Αρχικά πρέπει να ορίσουμε τα εξής:

- TP (True Positives): κατάσταση θετικών δειγμάτων που η κατηγοριοποίηση (πρόβλεψη) έχει γίνει σωστά από το μοντέλο ως θετικά

- FN (False Negatives): κατάσταση θετικών δειγμάτων που η κατηγοριοποίηση (πρόβλεψη) έχει γίνει εσφαλμένα από το μοντέλο ως αρνητικά,

- FP (False Positives): κατάσταση αρνητικών δειγμάτων που η κατηγοριοποίηση (πρόβλεψη) έχει γίνει εσφαλμένα από το μοντέλο ως θετικά,

- TN (True Negatives): κατάσταση αρνητικών δειγμάτων που η κατηγοριοποίηση (πρόβλεψη) από το μοντέλο έχει γίνει σωστά ως αρνητικά,

TPR (True Positive Rate): Ποσοστό ταξινόμησης (πρόβλεψης) θετικών δειγμάτων που έχουν προβλεφθεί σωστά όπου :

$$\mathrm{TPR} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}} \quad (1)$$

TNR (True Negative Rate): Ποσοστό ταξινόμησης (πρόβλεψης) αρνητικών δειγμάτων που έχει προβλεφθεί σωστά όπου :

$$\mathrm{TNR} = \frac{\mathrm{TN}}{\mathrm{TN} + \mathrm{FP}} \quad (2)$$

FPR (False Positive Rate): Ποσοστό ταξινόμησης (πρόβλεψη) αρνητικών δειγμάτων που έχουν προβλεφθεί ως θετικά όπου :

$$\text{FPR} = \frac{\text{FP}}{\text{TN} + \text{FP}} \quad (3)$$

FNR (False Negative Rate): Ποσοστό ταξινόμησης (πρόβλεψης) θετικών δειγμάτων που έχουν προβλεφθεί ως αρνητικά όπου :

$$\text{FNR} = \frac{\text{FN}}{\text{TP} + \text{FN}} \quad (4)$$

Βέβαια ισχύει και για τα N (negatives) και τα P (positives) οι εξής σχέσεις:

$$\text{N} = \text{FP} + \text{TN} \text{ και } \text{P} = \text{FN} + \text{TP} \quad (5)$$

Στον Πίνακα 3 γίνεται η περιγραφή των μετρικών (metrics) της απόδοσης ενός μοντέλου που υπολογίστηκαν μαζί τους πίνακες σύγχυσης στην παρούσα μεταπτυχιακή εργασία για όλα τα μοντέλα μηχανικής μάθησης με σκοπό την συγκριτική τους αξιολόγηση στα επόμενα κεφάλαια. Σε αυτό το σημείο επισημαίνεται ότι τα μοντέλα στην μεταπτυχιακή εργασία συγκρίθηκαν μεταξύ τους βάσει της ορθότητας (accuracy) τους στα δεδομένα δοκιμής (test set). Επίσης, πέρα από τα μετρικά του πίνακα βρέθηκε και η παράμετρος support που μετράει πόσα δείγματα από κάθε κλάση βρισκόντουσαν στα δεδομένα δοκιμής (test set).

Πίνακας 3. Μετρικές αποτίμησης μοντέλων μηχανικής μάθησης

Accuracy – Ορθότητα	Accuracy = (TP +TN) / (TP+FN+FP+TN)
Precision - Ακρίβεια	Precision=TP / (TP + FP)
Recall - Ανάκληση	Recall=TPR= TP / (TP+FN)
Specifity - Ειδικότητα	Specifity=1-FPR=TNR
F1 score – Αρμονικός μέσος	F1= 2 (Precision x Recall) / (Precision + Recall)

Ενδεικτικά στον Πίνακα 4 παρουσιάζεται και ένας πίνακας σύγχυσης για δυαδική ταξινόμηση.

Πίνακας 4. Πίνακας σύγχυσης για binary classification

		Προβλεπόμενες κλάσεις - Predicted Class	
		Class -1	Class - 2
Πραγματικές κλάσεις - Actual Class	Class - 1	TP (true positives)	FN (false negatives)
	Class - 2	FP (false positives)	TN (true negatives)

Είναι σημαντικό να αναφερθεί ότι για να πραγματοποιηθεί η συγκριτική αξιολόγηση μοντέλων μηχανικής μάθησης με σωστό τρόπο, πρέπει η προετοιμασία των δεδομένων πριν την εκπαίδευση να είναι ακριβώς η ίδια για όλα τα μοντέλα. Επιπροσθέτως, ο διαχωρισμός (split) των δεδομένων σε δεδομένα εκπαίδευσης (train set) και σε δεδομένα δοκιμής (test set) θα πρέπει να πραγματοποιείται με τον ίδιο τρόπο για όλα τα μοντέλα μηχανικής μάθησης. Στην μεταπτυχιακή εργασία τα δεδομένα χωρίστηκαν σε 80% δεδομένα εκπαίδευσης και 20% δεδομένα δοκιμής ανάλογα με πείραμα – σενάριο που αφορούσε την προετοιμασία των δεδομένων. Κάθε πείραμα - σενάριο βέβαια όπως είναι κατανοητό περιείχε την ίδια ακριβώς προετοιμασία των δεδομένων για όλα τα μοντέλα και περιγράφεται αναλυτικά στη συνέχεια της εργασίας.

Κεφάλαιο 3: ΑΝΑΛΥΣΗ ΠΡΟΒΛΗΜΑΤΟΣ

Στο παρόν κεφάλαιο γίνεται η ανάλυση του προβλήματος. Εφαρμόστηκε πλήθος τεχνικών ανάλυσης των δεδομένων, καθώς και διάφορες τεχνικές οπτικοποιήσεων. Ο σκοπός ήταν η ανάλυση, η αποσαφήνιση και η κατανόηση (analytics) των δεδομένων. Η χρήση μοντέλων μηχανικής μάθησης προϋποθέτει την ανάλυση δεδομένων και την εφαρμογή στατιστικών τεχνικών, καθώς μέσω αυτών παράγονται σημαντικές πληροφορίες (insights-kPIs) για το πρόβλημα. Η απλή εφαρμογή μοντέλων μηχανικής μάθησης για την δημιουργία προβλέψεων είναι κάτι που σε πραγματικά προβλήματα συνήθως δεν οδηγεί στα καλύτερα αποτελέσματα, καθώς στις περισσότερες περιπτώσεις τα δεδομένα δεν είναι ποτέ ιδανικά ώστε τα μοντέλα να ανακαλύψουν όλα τα patterns. Οπότε, πέρα από τον καθαρισμό των δεδομένων και τα imputations δεδομένων σε περιπτώσεις έλλειψης παρατηρήσεων , είναι απαραίτητη και η ανάλυση τους με σκοπό την καλύτερη κατανόηση του προβλήματος και τελικά την επίτευξη υψηλότερης ακρίβειας Στην συγκεκριμένη εργασία δόθηκε ιδιαίτερο βάρος στην ανάλυση δεδομένων, καθώς ζητούμενο δεν υπήρξε απλά η εκτέλεση μοντέλων μηχανικής μάθησης.

3.1 Ανάλυση Δεδομένων

Όπως έχει αναφερθεί το συγκεκριμένο σύνολο δεδομένων έχει 27 χαρακτηριστικά (features) που έχουν εξαχθεί με λογισμικό μέσω φωτογραφιών και 7 τύπους ελαττωμάτων για χαλύβδινες πλάκες. Το πρώτο πράγμα που έγινε ήταν ο έλεγχος των δεδομένων για χαμένες τιμές (data cleaning). Ο αριθμός των δεδομένων είναι 66,994 κελιά (datapoints – 1941 γραμμές x 34 στήλες) και δεν είναι καθόλου ασφαλές να υποθέσουμε ότι όλες οι τιμές είναι περασμένες. Αφού λοιπόν διαβάστηκαν τα δεδομένα και περάστηκαν σε dataframes (pandas) πραγματοποιήθηκε ο έλεγχος και δεν βρέθηκε να λείπουν δεδομένα. Επιπλέον, αντιγράφτηκε το σύνολο δεδομένων όπου αφαιρέθηκε το one-hot-encoding [58] και δημιουργήθηκε η τιμή στόχος (for i=1:7) για όλες τις κλάσεις του προβλήματος.

Έτσι αντί για 34 στήλες τα δεδομένα μας πλέον έχουν 28. Η αφαίρεση του one – hot – encoding έγινε για λόγους ευκολίας χρήσης κώδικα. Βέβαια, κρατήθηκε και το αρχικό σύνολο δεδομένων για διάφορες χρήσεις και επεξεργασίας των δεδομένων με την αρχική τους μορφή.

Στη συνέχεια, αυτό που έγινε ήταν ο υπολογισμός βασικών στατιστικών μεγεθών των δεδομένων, όπως μέση τιμή, τυπική αποκλιση, ελάχιστη και μέγιστη τιμή, 1ο τεταρτημόριο και 3ο τεταρτημόριο (mean value, standar deviation, mean and maximum values, 1st quartile , mean and 3rd quartiles), για να υπάρξει μία πρώτη εικόνα για τα δεδομένα. Στον Πίνακα 5 παρουσιάζονται οι τιμές βασικών στατιστικών μεγεθών για τα 27 χαρακτηριστικά του προβλήματος.

Πίνακας 5. Βασικά στατιστικά μεγέθη

	X_Minimum	X_Maximum	Y_Minimum	Y_Maximum	Pixels_Areas	X_Perimeter
count	1941	1941	1941	1941	1941	1941
mean	571.1360124	617.9644513	1650684.868	1650738.705	1893.878413	111.8552293
std	520.6906714	497.6274103	1774578.415	1774590.089	5168.45956	301.2091871
min	0	4	6712	6724	2	2
25%	51	192	471253	471281	84	15
50%	435	467	1204128	1204136	174	26
75%	1053	1072	2183073	2183084	822	84
max	1705	1713	12987661	12987692	152655	10449
	Y_Perimeter	Sum_of_Luminosity	Minimum_of_Luminosity	Maximum_of_Luminosity	Length_of_Conveyer	TypeOfSteel_A300
count	1941	1941	1941	1941	1941	1941
mean	82.96599691	206312.1479	84.54868624	130.1937146	1459.160227	0.400309119
std	426.4828792	512293.5876	32.1342757	18.69099187	144.5778233	0.490087208
min	1	250	0	37	1227	0
25%	13	9522	63	124	1358	0
50%	25	19202	90	127	1364	0
75%	83	83011	106	140	1650	1
max	18152	11591414	203	253	1794	1
	TypeOfSteel_A400	Steel_Plate_Thickness	Edges_Index	Empty_Index	Square_Index	Outside_X_Index
count	1941	1941	1941	1941	1941	1941
mean	0.599690881	78.73776404	0.331715198	0.414203349	0.57076713	0.033361103
std	0.490087208	55.08603169	0.299711749	0.137261489	0.271058385	0.058961169
min	0	40	0	0	0.0083	0.0015
25%	0	40	0.0604	0.3158	0.3613	0.0066
50%	1	70	0.2273	0.4121	0.5556	0.0101
75%	1	80	0.5738	0.5016	0.8182	0.0235
max	1	300	0.9952	0.9439	1	0.8759
	Edges_X_Index	Edges_Y_Index	Outside_Global_Index	LogOfAreas	Log_X_Index	Log_Y_Index
count	1941	1941	1941	1941	1941	1941
mean	0.610528645	0.813472231	0.575734158	2.492388357	1.335686141	1.403271303
std	0.243276919	0.234273623	0.48235199	0.788929853	0.481611609	0.454345162
min	0.0144	0.0484	0	0.301	0.301	0
25%	0.4118	0.5968	0	1.9243	1	1.0792
50%	0.6364	0.9474	1	2.2406	1.1761	1.3222
75%	0.8	1	1	2.9149	1.5185	1.7324
max	1	1	1	5.1837	3.0741	4.2587
	Orientation_Index	Luminosity_Index	SigmoidOfAreas			
count	1941	1941	1941			
mean	0.083287635	-0.131305049	0.585420453			
std	0.500868047	0.14876684	0.339451805			
min	-0.991	-0.9989	0.119			
25%	-0.3333	-0.195	0.2482			
50%	0.0952	-0.133	0.5063			
75%	0.5116	-0.0666	0.9998			
max	0.9917	0.6421	1			

Επιπροσθέτως, έγινε υπολογισμός της διακύμανσης των χαρακτηριστικών (σ^2 - variance). Η διακύμανση αποτελεί ένα σημαντικό στατιστικό μέγεθος, καθώς δείχνει την απόσταση των χαρακτηριστικών από την μέση τιμή τους. Δεδομένα τα οποία έχουν πολύ χαμηλή διακύμανση ή διακύμανση πολύ κοντά στο μηδέν είναι κατανοητό ότι οριακά μπορούν να θεωρηθούν σταθερές (constants) και δεν μπορούν να συνεισφέρουν στην απόδοση των μοντέλων μηχανικής μάθησης. Συνηθίζεται χαρακτηριστικά που έχουν πολύ χαμηλή διακύμανση να αφαιρούνται από το σύνολο των δεδομένων, καθώς δεδομένα με πολύ χαμηλή διακύμανση περιέχουν και λιγότερη πληροφορία, κάτι που δεν βοηθάει τα μοντέλα μηχανικής μάθησης. Στη συγκεκριμένη εργασία δεν αφαιρέθηκαν χαρακτηριστικά με χαμηλή διακύμανση καθώς χρησιμοποιήθηκαν πιο πολύπλοκες τεχνικές για την εκτίμηση της σημαντικότητας των χαρακτηριστικών (model based feature importances). Παρακάτω, στον Πίνακα 6 φαίνεται η διακύμανση για τα 28 χαρακτηριστικά των χαλύβδινων πλακών.

Πίνακας 6. Διακύμανση Χαρακτηριστικών

X_Minimum	271118.8
X_Maximum	247633
Y_Minimum	3.14913E+12
Y_Maximum	3.14917E+12
Pixels_Areas	26712970
X_Perimeter	90726.97
Y_Perimeter	181887.6
Sum_of_Luminosity	2.62445E+11
Minimum_of_Luminosity	1032.612
Maximum_of_Luminosity	349.3532
Length_of_Conveyer	20902.75
TypeOfSteel_A300	0.2401855
TypeOfSteel_A400	0.2401855
Steel_Plate_Thickness	3034.471
Edges_Index	0.08982713
Empty_Index	0.01884072
Square_Index	0.07347265
Outside_X_Index	0.003476419
Edges_X_Index	0.05918366
Edges_Y_Index	0.05488413
Outside_Global_Index	0.2326634
LogOfAreas	0.6224103
Log_X_Index	0.2319497
Log_Y_Index	0.2064295
Orientation_Index	0.2508688
Luminosity_Index	0.02213157
SigmoidOfAreas	0.1152275

Παρατηρούμε ότι δεν υπάρχουν χαρακτηριστικά με σχεδόν μηδενική διακύμανση, αν και υπάρχουν κάποια χαρακτηριστικά με αρκετά μικρή διακύμανση. Επίσης, παρατηρείται ότι τα χαρακτηριστικά TypeOfSteel_A300 και TypeOfSteel_A400 έχουν ακριβώς την ίδια διακύμανση. Αυτό συμβαίνει διότι τα χαρακτηριστικά αυτά είναι ουσιαστικά κατηγορικές μεταβλητές αλλά με το one hot encoding είναι απλά τιμές που λαμβάνουν είτε την τιμή μηδέν είτε την τιμή ένα. Επιπλέον, η κάθε χαλύβδινη πλάκα ανήκει ή στην μία ή στην άλλη σειρά χάλυβα με αποτέλεσμα η απόσταση τους από την μέση τιμή να είναι ίδια. Αυτό μας οδηγεί στο συμπέρασμα ότι η διακύμανση τους θα πρέπει να ταυτίζεται.

Επίσης, πραγματοποιήθηκε κανονικοποίηση (normalization) των δεδομένων με σκοπό την αναδιάταξη τους ώστε αυτά να λαμβάνουν τιμές στο εύρος (0,1). Στο πλαίσιο της απλής προ-επεξεργασίας έγινε και standardization των δεδομένων με μέσο όρο των στηλών στο μηδέν και τυπική απόκλιση στο ένα για απλή σύγκριση. Ο μετασχηματισμός των δεδομένων σε δεδομένα ίδιας κλίμακας είναι πολύ σημαντικός και απαραίτητος για την εφαρμογή μοντέλων μηχανικής μάθησης.

Η εξίσωση που χρησιμοποιείται για την κανονικοποίηση των δεδομένων βρίσκεται παρακάτω, όπου Xi είναι η τιμή του δείγματος του χαρακτηριστικού και Xmax, Xmin οι μέγιστες και ελάχιστες τιμές όλων των δειγμάτων για το X χαρακτηριστικό.

$$\text{Xi normalised} = \frac{\text{Xi} - \text{Xmin}}{\text{Xmax} - \text{Xmin}} \quad (6)$$

Η εξίσωση που χρησιμοποιείται για το standardization των δεδομένων βρίσκεται παρακάτω, όπου Xi είναι η τιμή του δείγματος του χαρακτηριστικού με σ την τυπική απόκλιση και μ την μέση τιμή.

$$\text{Xi standardizated} = \frac{\text{Xi} - \mu}{\sigma} \quad (7)$$

Όταν γίνεται χρήση του standardization τα δεδομένα δεν περιορίζονται σε ένα συγκεκριμένο εύρος τιμών σε αντίθεση με την κανονικοποίηση. Αυτό που συμβαίνει όμως είναι να ανασχηματίζονται με τέτοιο τρόπο ώστε να έχουν μέση τιμή μηδέν και τυπική απόκλιση ένα. Δεν υπάρχει γενικός κανόνας που να ορίζει το ποια μέθοδος είναι καλύτερη. Στη συγκεκριμένη εργασία δοκιμάστηκαν και οι δύο μέθοδοι αλλά προτιμήθηκε να γίνει χρήση του normalization. Βέβαια δεν υπήρχαν σημαντικές αποκλίσεις στις αποδόσεις και με την χρήση του standardization.

Εικόνα 7. Normalization vs Standardization

Στην συνέχεια υπολογίστηκε ο συντελεστής γραμμικής συσχέτισης (pearson correlation) [59] για όλα τα δεδομένα (all vs all). Ο συγκεκριμένος συντελεστής λαμβάνει τιμές στο εύρος (-1,1). Όσο μεγαλύτερη είναι η απόλυτη τιμή του τόσο μεγαλύτερη είναι η γραμμική συσχέτιση των δεδομένων. Στην περίπτωση που ο συντελεστής λαμβάνει την τιμή -1 σημαίνει ότι τα δύο χαρακτηριστικά συμπεριφέρονται αντιστρόφως ανάλογα. Στην μηχανική μάθηση η ταυτόχρονη χρήση δεδομένων που είναι γραμμικώς εξαρτημένα, ως παράμετροι εισόδου, πολλές φορές δεν επιφέρει αλλαγές στις αποδόσεις των μοντέλων. Επιπλέον, αφαιρώντας το ένα από τα χαρακτηριστικά, το μοντέλο μηχανικής μάθησης γίνεται πιο γρήγορο.

Η λογική που ακολουθείται είναι ότι σε περίπτωση που η απόδοση του μοντέλου μειώνεται ελάχιστα με την αφαίρεση των χαρακτηριστικών τότε καλώς αφαιρούνται, καθώς το μοντέλο μας γίνεται πιο απλό και γρήγορο. Βέβαια, εδώ πρέπει να αναφερθεί ότι ο συγκεκριμένος συντελεστής αυτό που δείχνει είναι συσχέτιση και όχι αιτιότητα (causality). Αυτό ουσιαστικά σημαίνει ότι σε περίπτωση που έχουμε δύο μεταβλητές που δεν έχουν καμία σχέση μεταξύ τους μέσα στο σύνολο δεδομένων μας και παρουσιάσουν υψηλή γραμμική συσχέτιση, δεν μπορεί να εξασφαλιστεί ότι αυτές οι μεταβλητές σχετίζονται αναγκαστικά με αιτιότητα. Ένα σχετικό άρθρο μπορεί να βρεθεί εδώ [60].

Παρακάτω, παρουσιάζεται ενδεικτικά ο τύπος για τον υπολογισμό της γραμμικής συσχέτισης μεταξύ δύο μεταβλητών. Όπου σ η τυπική απόκλιση και cov(x,y), η συνδιακύμανση (covariance) των δύο μεταβλητών.

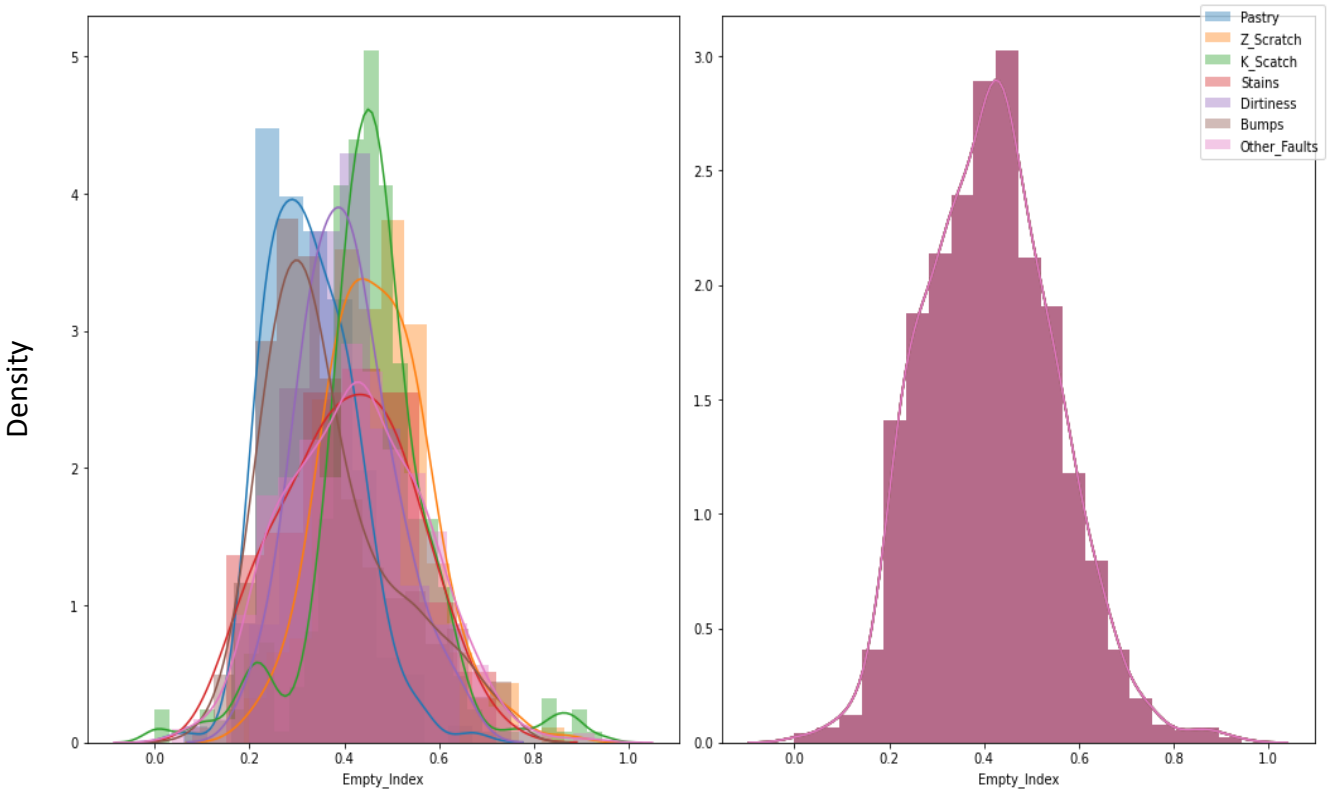
$$\rho_{xy} = \frac{\mathrm{cov}(x,y)}{\sigma_x \sigma_y} \quad (8)$$

Υπολογίστηκε η γραμμική συσχέτιση για όλα τα δεδομένα και παρουσιάζεται στον Πίνακα 7 για τα πρώτα 8 χαρακτηριστικά σε heat-map, όπου οι υψηλότερες γραμμικές συσχετίσεις έχουν πιο ερυθρό χρώμα. Στο Παράρτημα 1 - Γραμμικές Συσχετίσεις μπορεί να βρεθεί και ολόκληρο το διάγραμμα. Ενδεικτικά, παρατηρήθηκε η γραμμική συσχέτιση ανάμεσα στο X_minimum,Y_minimum και στα X_maximum, Y_maximum. Επιπλέον, το χαρακτηριστικό Pixels_Areas παρουσιάζει γραμμική συσχέτιση με τα X_Perimeter, Y_Perimeter, Sum_of_Luminosity. Κάποια άλλα χαρακτηριστικά που συσχετίζονται γραμμικά είναι το Maximum_of_Luminosity με το Luminosity_Index και το Log_X_Index. Επίσης, το Outside_X_Index με το Log_of_Areas. Τέλος, γραμμική συσχέτιση βλέπουμε στο SigmoidOfAreas με το LogOfAreas και το Log_Y_Index. Τα είδη της σειράς του χάλυβα δηλαδή το TypeOfSteel_A300 και το TypeOfSteel_A400 έχουν την μεγαλύτερη δυνατή αρνητική συσχέτιση δηλαδή -1, όπως ήταν αναμενόμενο και όπως έχει περιγραφεί προηγουμένως. Δεν επιλέχθηκε να αφαιρεθεί κάποιο από τα δύο καθώς οι δύο σειρές χάλυβα θα μπορούσαν να σχετίζονται με τον τύπο του ελαττώματος της χαλύβδινης πλάκας λόγω διαφορών στην χημική τους σύσταση. Γενικά, σε αυτό το σημείο είναι σημαντικό να αναφερθεί ότι δεν αφαιρέθηκε κανένα χαρακτηριστικό από τα δεδομένα καθώς δεν κρίθηκε αρκετή η ένδειξη από το στατιστικό μέγεθος της γραμμικής συσχέτισης και θεωρήθηκε ότι θα πρέπει να πραγματοποιηθούν και άλλοι έλεγχοι.

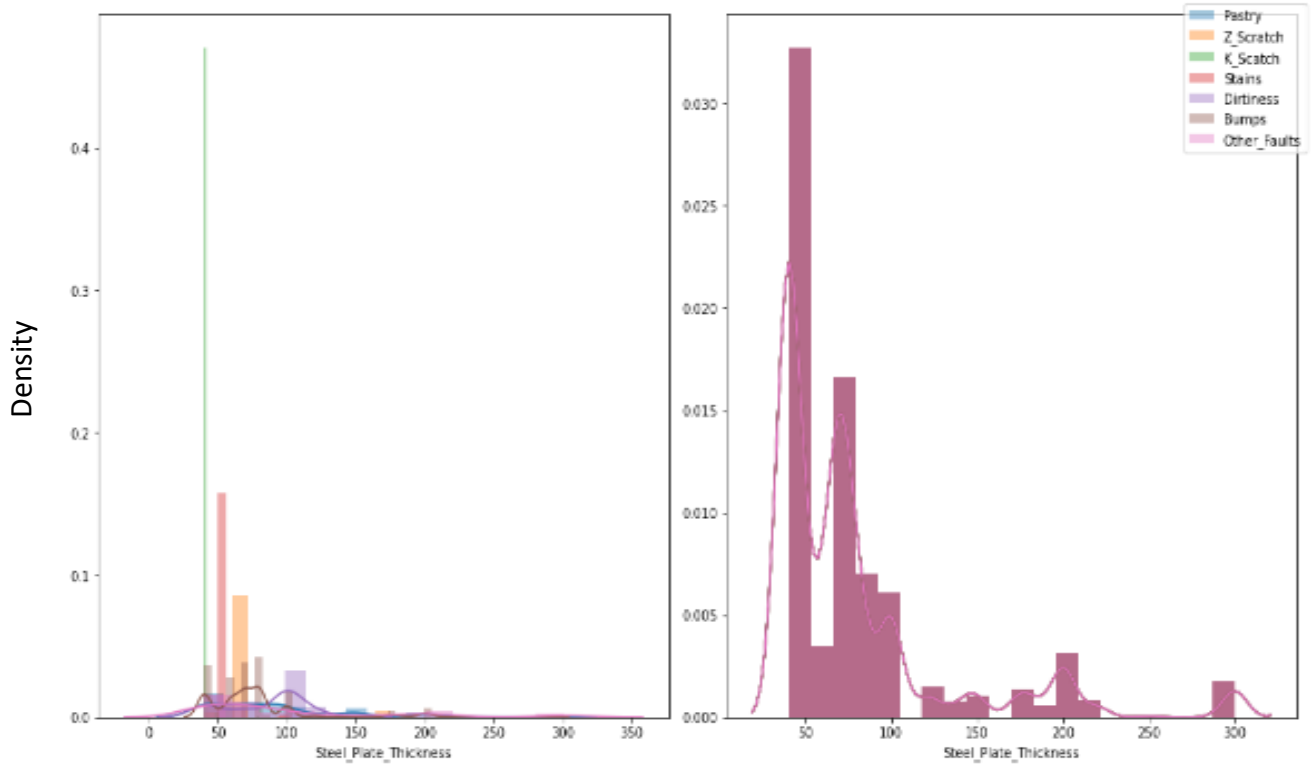
Πίνακας 7. Πίνακας Γραμμικών Συσχετίσεων Δεδομένων για τα πρώτα 8 χαρακτηριστικά

	X_Minimum	X_Maximum	Y_Minimum	Y_Maximum	Pixels_Areas	X_Perimeter	Y_Perimeter	Sum_of_Luminosity
X_Minimum	1	0.988313538	0.041821303	0.041806514	-0.307321756	-0.258937378	-0.118757368	-0.339045394
X_Maximum	0.988313538	1	0.052147349	0.052135154	-0.225399282	-0.186326058	-0.090137677	-0.247052068
Y_Minimum	0.041821303	0.052147349	1	0.999999973	0.017669766	0.023843168	0.024150084	0.007362208
Y_Maximum	0.041806514	0.052135154	0.999999973	1	0.017840349	0.024038111	0.0243804	0.00749885
Pixels_Areas	-0.307321756	-0.225399282	0.017669766	0.017840349	1	0.96664407	0.827199212	0.978951617
X_Perimeter	-0.258937378	-0.186326058	0.023843168	0.024038111	0.96664407	1	0.912436459	0.912955837
Y_Perimeter	-0.118757368	-0.090137677	0.024150084	0.0243804	0.827199212	0.912436459	1	0.704875735
Sum_of_Luminosity	-0.339045394	-0.247052068	0.007362208	0.00749885	0.978951617	0.912955837	0.704875735	1
Minimum_of_Luminosity	0.237637154	0.168649497	-0.065702585	-0.065733089	-0.497204417	-0.400427368	-0.213758225	-0.540565938
Maximum_of_Luminosity	-0.075554349	-0.062391556	-0.067785101	-0.067775823	0.110062541	0.111363038	0.061808674	0.136515251
Length_of_Conveyer	0.316662417	0.299390273	-0.049210774	-0.04921918	-0.155852584	-0.134239989	-0.063824645	-0.169330746
TypeOfSteel_A300	0.144319482	0.112008867	0.07516363	0.075150658	-0.235591307	-0.18925007	-0.09515373	-0.263631925
TypeOfSteel_A400	-0.144319482	-0.112008867	-0.07516363	-0.075150658	0.235591307	0.18925007	0.09515373	0.263631925
Steel_Plate_Thickness	0.136625071	0.106118853	-0.207640422	-0.207644246	-0.183735086	-0.147711687	-0.058888803	-0.204811585
Edges_Index	0.278074892	0.242846344	0.021314187	0.02129962	-0.275288501	-0.227589928	-0.111239929	-0.30145247
Empty_Index	-0.198460847	-0.152680237	-0.04311715	-0.043085317	0.272808218	0.306347742	0.188824662	0.293690984
Square_Index	0.063657853	0.048575007	-0.006134875	-0.006151853	0.017865472	0.004506931	-0.04751137	0.049606974
Outside_X_Index	-0.361159508	-0.214930402	0.054164575	0.054184891	0.588605799	0.51709842	0.209160155	0.658338913
Edges_X_Index	0.154778153	0.149258695	0.06608496	0.066051086	-0.294673005	-0.293038692	-0.195162304	-0.327728351
Edges_Y_Index	0.367907412	0.2719148	-0.036543194	-0.036549142	-0.463571447	-0.412099511	-0.136722659	-0.529745402
Outside_Global_Index	0.147281761	0.099252892	-0.062911339	-0.062901483	-0.109654571	-0.079105547	0.013438174	-0.121089808
LogOfAreas	-0.42855285	-0.332169137	0.044951797	0.044993592	0.650233889	0.5630359	0.294039794	0.712128115
Log_X_Index	-0.437943859	-0.324011794	0.070406288	0.070431806	0.603071866	0.524715503	0.228484845	0.667736457
Log_Y_Index	-0.326850545	-0.265990105	-0.008441982	-0.00838184	0.578342453	0.523471521	0.344377686	0.618795075
Orientation_Index	0.178585372	0.115019392	-0.086496905	-0.08647971	-0.137603658	-0.101731103	0.031381341	-0.158483002
Luminosity_Index	-0.031577766	-0.038996153	-0.090653533	-0.090665823	-0.043448768	-0.032616544	-0.047777525	-0.014066927
SigmoidOfAreas	-0.355251295	-0.286735946	0.025257091	0.025283891	0.422947493	0.380604947	0.191771984	0.464248328
Pastry	0.134955713	0.119813886	0.036488056	0.036488363	-0.076752414	-0.075418034	-0.01761618	-0.084306591
Z_Scratch	-0.228959501	-0.258178136	-0.063326779	-0.063328669	-0.088440214	-0.060581843	-0.025721009	-0.099592485
K_Scatch	-0.4192635	-0.336084078	-0.000420434	-0.000397046	0.556846091	0.455002584	0.203062982	0.616950055
Stains	0.073739654	0.061471499	-0.066601097	-0.066606081	-0.071182477	-0.067546583	-0.035743253	-0.078111362
Dirtiness	0.103923987	0.096523301	0.064262263	0.064262117	-0.050578121	-0.037820327	-0.01005756	-0.055271571
Bumps	0.221295537	0.201703517	0.12612119	0.126110378	-0.163738543	-0.140196899	-0.07098861	-0.179830793
Other_Faults	0.164803624	0.145783469	-0.084414679	-0.084422148	-0.184631794	-0.142903127	-0.066800915	-0.205889612

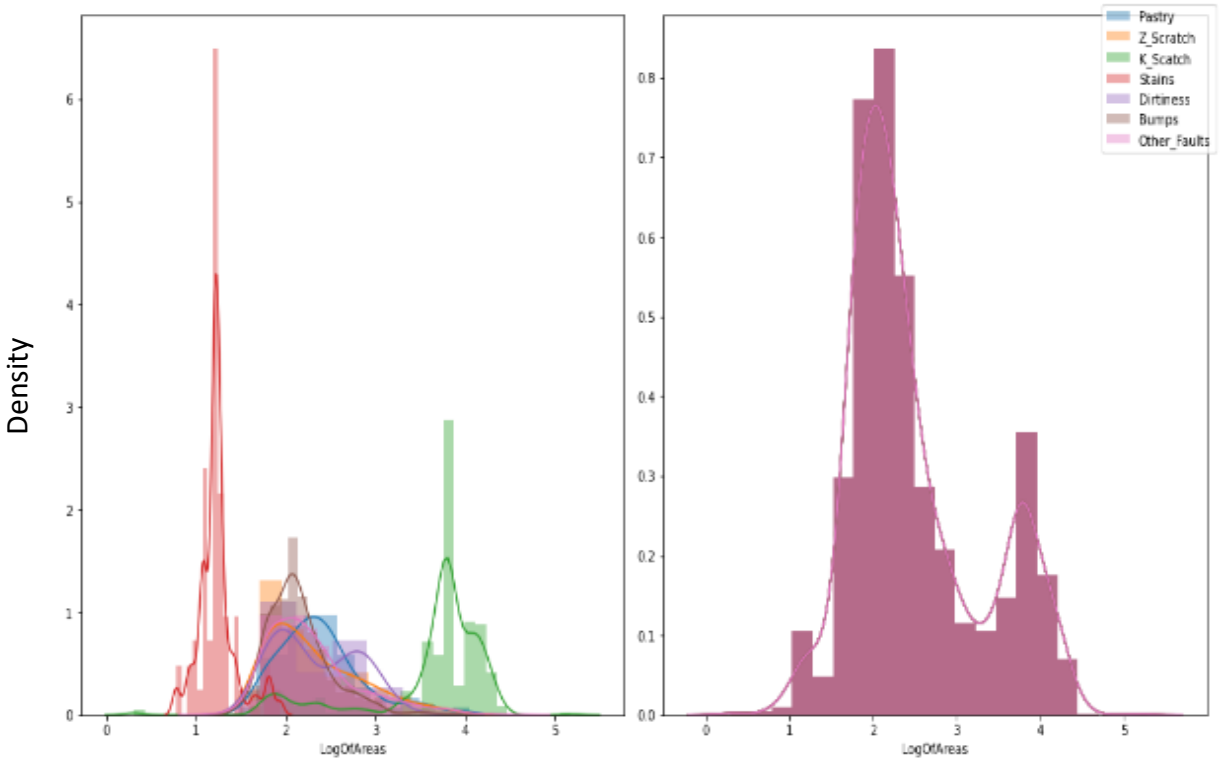
Για την καλύτερη κατανόηση των δεδομένων εκτυπώθηκαν οι 7 κατανομές των χαρακτηριστικών του συνόλου δεδομένων βάσει των τύπων ελαττωμάτων αλλά και 1 συνολική κατανομή για τις τιμές κάθε χαρακτηριστικού για όλα τα δεδομένα. Αυτό πραγματοποιήθηκε με σκοπό την παρατήρηση του εύρους τιμών των χαρακτηριστικών που σχετίζονται με συγκεκριμένους τύπους ελαττωμάτων (classes). Σε ένα dataset με 7 κλάσσεις (multiclass) δεν είναι καθόλου σίγουρο αν θα καταφέρουν τα μοντέλα μηχανικής μάθησης να διαχωρίζουν τα δεδομένα και τελικά να τα ταξινομούν σωστά. Αυτό εξαρτάται κυρίως από τις κατανομές των δεδομένων, καθώς αυτές επηρεάζουν σε πολύ μεγάλο βαθμό την απόδοση των μοντέλων. Με πιο απλά λόγια, αν δεν υπάρχουν καθόλου μοτίβα που ακολουθούν τα δεδομένα σε συμφωνία με τις αντίστοιχες κλάσεις, δημιουργείται δυσκολία που δυσχεραίνει την απόδοση των μοντέλων. Παρακάτω, στις Εικόνες 8 -10 παρουσιάζονται ενδεικτικά κατανομές κάποιων χαρακτηριστικών του συνόλου δεδομένων. Σε αυτό το σημείο είναι σημαντικό να σημειωθεί ότι τα γραφήματα για όλα τα χαρακτηριστικά μπορούν να βρεθούν στο Παράρτημα 2 - Κατανομές Δεδομένων.



Εικόνα 8. Κατανομή χαρακτηριστικού Empty_Index



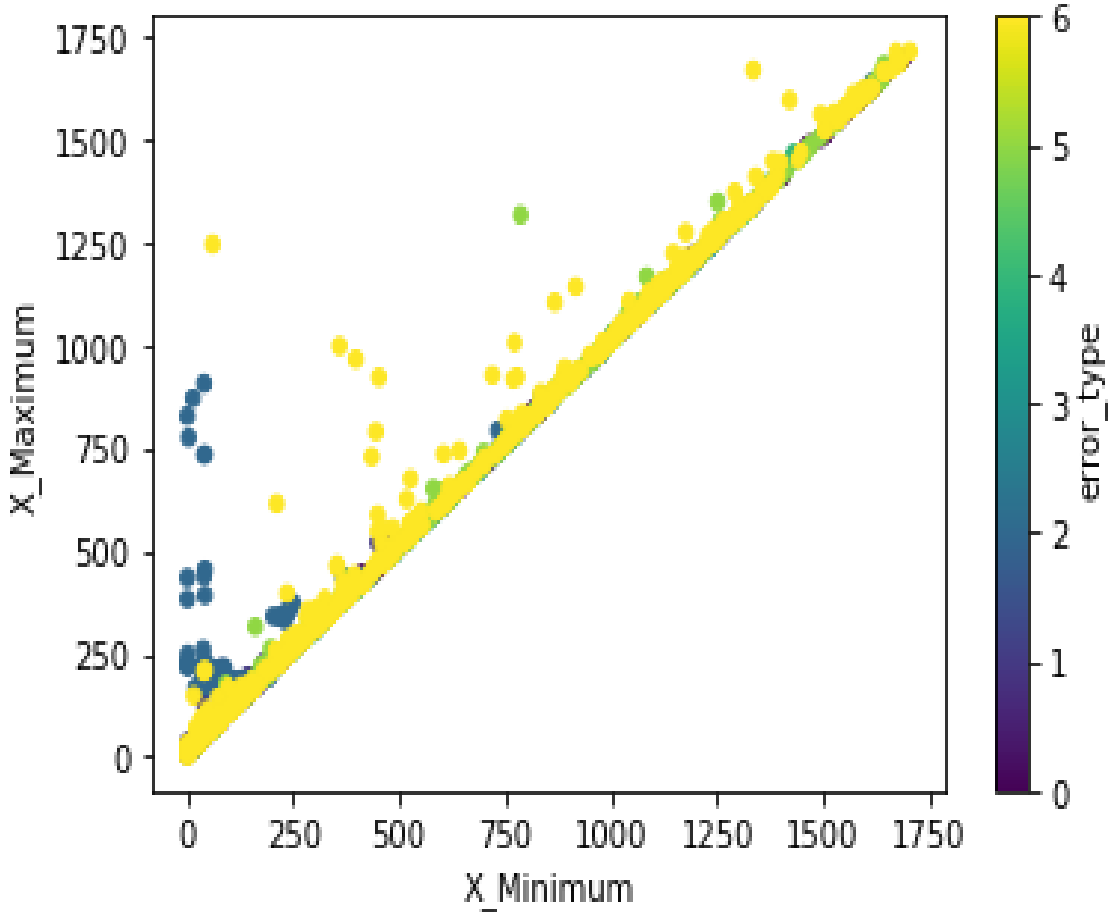
Εικόνα 9. Κατανομή χαρακτηριστικού Steel plate thickness



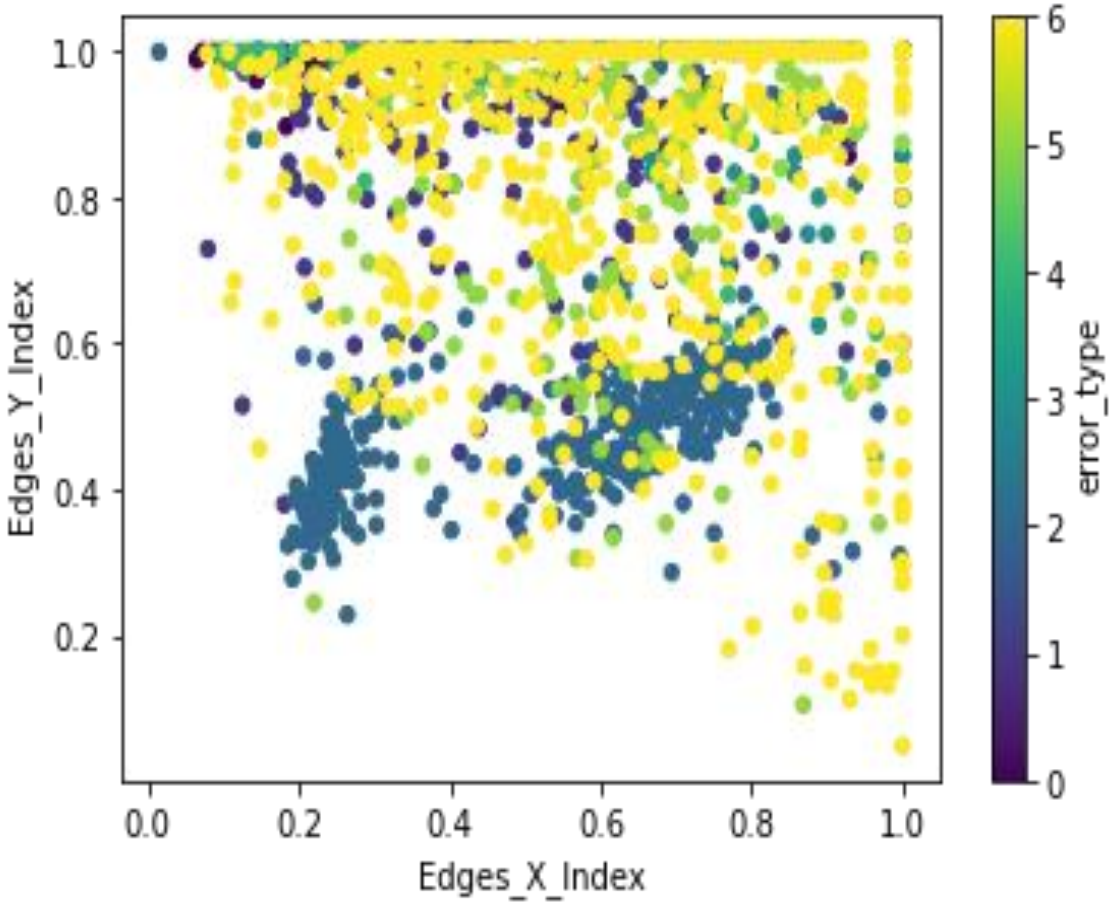
Εικόνα 10. Κατανομή χαρακτηριστικού LogOfAreas

Αρχικά βρέθηκαν οι κατανομές όλων των χαρακτηριστικών και υλοποιήθηκαν 27 διαγράμματα (δεξί διάγραμμα). Στην συνέχεια βρέθηκαν όλες οι κατανομές των χαρακτηριστικών ανά κλάση (αριστερό διάγραμμα). Οπότε για κάθε ένα χαρακτηριστικό πλοταρίστηκαν 7 κατανομές, όσοι δηλαδή οι τύποι των ελαττωμάτων. Αυτό που θέλαμε να δούμε είναι αν τα χαρακτηριστικά ή έστω κάποια από αυτά, λαμβάνουν τιμές σε συγκεκριμένο εύρος αντίστοιχα με την κλάση που είχαν κατηγοριοποιηθεί. Σε περίπτωση που συνέβαινε κάτι τέτοιο θα ήταν ιδιαίτερα θετικό ως προς την απόδοση των μοντέλων στο μετέπειτα στάδιο. Στην Εικόνα 8 μπορούμε να δούμε ότι το χαρακτηριστικό Empty_Index λαμβάνει παρόμοιες τιμές στο ίδιο εύρος για όλους τους τύπους ελαττωμάτων. Αυτό σημαίνει ότι δεν διαφοροποιείται πάρα πολύ για διαφορετικά ελαττώματα κάτι που μειώνει την σημαντικότητα του. Δηλαδή μπορεί να προσεγγίζουν όλες οι κατανομές ανά τύπο ελαττώματος την γκαουσιανή αλλά δεν υπάρχει διακριτική ικανότητα των κλάσεων αφού είναι όλες μαζί. Στην αντίθετη κατεύθυνση φαίνεται να κινείται το χαρακτηριστικό Log_of_Areas στην Εικόνα 10 στο οποίο οι τιμές τους διαφοροποιούνται σε αρκετά μεγάλο βαθμό ως προς τις κλάσεις K_Scratch και Stains. Επίσης, στην Εικόνα 9 το χαρακτηριστικό Steel_plate_thickness φαίνεται να διαφοροποιείται σε αρκετές κλάσεις ως προς τις τιμές του. Αυτό που θα πρέπει να αναφερθεί σε αυτό το σημείο είναι ότι για όλα σχεδόν τα χαρακτηριστικά η κατανομή τους ως προς την κλάση Other_Faults έπεφτε πάνω στις άλλες κατανομές των κλάσεων, κάτι που οδηγεί στο συμπέρασμα ότι θα υπάρχει δυσκολία ως προς την εύρεση της συγκεκριμένης κλάσης με υψηλή απόδοση από τα μοντέλα μηχανικής μάθησης.

Στην συνέχεια δημιουργήθηκαν διαγράμματα για όλα τα χαρακτηριστικά του συνόλου δεδομένων ανά δύο και ανά κλάση. Ο σκοπός ήταν να παρατηρηθούν οι τιμές των χαρακτηριστικών ως προς τύπους των κλάσεων τους καθώς και η διακριτική τους ικανότητα ως προς αυτές. Όπως και στα διαγράμματα των κατανομών η τελευταία κλάση Other_Faults που είναι και η πλειοψηφούσα κλάση του συνόλου δεδομένων φαίνεται να καλύπτει τις τιμές των χαρακτηριστικών για τις άλλες κλάσεις και να δυσχεραίνει τον διαχωρισμό. Γενικά, ο σκοπός ήταν σε αρχικό στάδιο να γίνουν διαγράμματα all vs all για τα χαρακτηριστικά και να δημιουργηθεί ένας πίνακας διαγραμμάτων (n-1) x (n–1) με n=27 . Κάτι τέτοιο δεν έγινε οπότε έγιναν διαγράμματα για τα χαρακτηριστικά ανά δύο. Τα διαγράμματα αυτά μπορούν να βρεθούν στο Παράρτημα 3 - Διαγράμματα Δεδομένων. Παρακάτω, στην Εικόνα 11 και στην Εικόνα 12 παρουσιάζονται ενδεικτικά τα διαγράμματα για κάποια χαρακτηριστικά. Για το X_ Maximum και το X_Minimum παρατηρείται και η γραμμική συσχέτιση που βρέθηκε με το στατιστικό δείκτη του pearson correlation.



Εικόνα 11. Τιμές χαρακτηριστικού X_Maximum και χαρακτηριστικού X_Minimum σε σχέση με τις κλάσεις τους



Εικόνα 12. Τιμές χαρακτηριστικού Edges_Y_Index και χαρακτηριστικού Edges_X_Index σε σχέση με τις κλάσεις τους

3.2 Εύρεση των χαρακτηριστικών με την μεγαλύτερη σημαντικότητα

Έγινε χρήση του μοντέλου μηχανικής μάθησης RF για την εξαγωγή χαρακτηριστικών με την μεγαλύτερη σημαντικότητα (model based feature importances). Ο τρόπος που υπολογίζεται η σημαντικότητα βασίζεται στην γενική λειτουργία του συγκεκριμένου μοντέλου και περιγράφεται αναλυτικά στην αναφορά [61]. Το μοντέλο RF χρησιμοποιεί το μετρικό Gini Impurity σε κάθε κόμβο (node) των Δέντρων Απόφασης [62]. Κατά την εκπαίδευση, υπολογίζεται κατά πόσο μειώνεται αυτό το μετρικό από κάθε επιλογή χαρακτηριστικού και έτσι προκύπτει μια κατάταξη των χαρακτηριστικών.

Η γενική ιδέα στηρίζεται στο πώς μπορεί η ορθότητα του μοντέλου να βελτιωθεί με την αφαίρεση κάθε φορά ενός χαρακτηριστικού και την επανάληψη ελέγχου της τιμής της ορθότητας με την σταδιακή αφαίρεση όλων των χαρακτηριστικών. Η υλοποίηση και ενσωμάτωση της συγκεκριμένης μεθόδου για την παρούσα μεταπτυχιακή εργασία έγινε

σύμφωνα με τις βιβλιοθήκες μηχανικής μάθησης ανοικτού κώδικα της βιβλιοθήκης του scikit-learn.

Παρατηρήθηκε ότι τα αποτελέσματα της παραπάνω τεχνικής είναι παρόμοια με τα αντίστοιχα που μπορούν να βρεθούν στη βιβλιογραφία, από άλλους ερευνητές που ακολούθησαν άλλες μεθόδους για το συγκεκριμένο σύνολο δεδομένων. Πιο συγκεκριμένα, η σημαντικότητα των χαρακτηριστικών που φαίνεται στον Πίνακα 8 συμφωνεί σε ποσοστό πάνω από 87% για τα 13 (48% των features) πρώτα χαρακτηριστικά στην κατάταξη σημαντικότητας με δημοσιευμένα αποτελέσματα [30]. Βέβαια, οι θέσεις στην κατάταξη είναι αρκετά διαφορετικές αλλά αυτό δεν έχει ιδιαίτερη σημασία καθώς ανιχνεύονται συνολικά τα καλύτερα χαρακτηριστικά. Η συγκεκριμένη μέθοδος ακολουθήθηκε με σκοπό να δοκιμαστούν μοντέλα μηχανικής μάθησης με λιγότερα χαρακτηριστικά ως προς τις αποδόσεις τους αλλά και τον χρόνο εκπαίδευσης τους που σαφώς μειώνεται.

Πίνακας 8. Feature Importance Score για RF και GBT

RF Importances	**Feature Importance_Score**	**GBT Importances**	**Feature Importance_Score**
Length_of_Conveyer	0.06602	Log_X_Index	0.166625
Sum_of_Luminosity	0.058524	Steel_Plate_Thickness	0.112757
Outside_X_Index	0.056446	Length_of_Conveyer	0.08971
Pixels_Areas	0.056089	TypeOfSteel_A400	0.066802
Log_X_Index	0.049425	Orientation_Index	0.062462
LogOfAreas	0.047734	Outside_X_Index	0.051323
Steel_Plate_Thickness	0.047015	Edges_Index	0.041721
X_Maximum	0.045889	Square_Index	0.037778
X_Minimum	0.042969	Pixels_Areas	0.036987
Minimum_of_Luminosity	0.040091	Empty_Index	0.034652
Orientation_Index	0.039887	Minimum_of_Luminosity	0.032534
SigmoidOfAreas	0.036814	Y_Minimum	0.030615
Square_Index	0.036646	Luminosity_Index	0.027592
Edges_Index	0.035615	X_Minimum	0.025551
Luminosity_Index	0.033841	Y_Maximum	0.02483
Y_Maximum	0.033052	Edges_Y_Index	0.023612
X_Perimeter	0.031206	TypeOfSteel_A300	0.023428
Y_Minimum	0.030659	Maximum_of_Luminosity	0.020415
Empty_Index	0.030634	X_Maximum	0.019824
Edges_Y_Index	0.030346	LogOfAreas	0.018847
Maximum_of_Luminosity	0.027529	Edges_X_Index	0.017849
Edges_X_Index	0.026698	Sum_of_Luminosity	0.010095
Y_Perimeter	0.026292	X_Perimeter	0.009823
TypeOfSteel_A400	0.024009	SigmoidOfAreas	0.006988
Log_Y_Index	0.022122	Y_Perimeter	0.005527
TypeOfSteel_A300	0.02183	Log_Y_Index	0.001534
Outside_Global_Index	0.00262	Outside_Global_Index	0.000118

Για να γίνει συγκριτική αξιολόγηση της σημαντικότητας των χαρακτηριστικών υλοποιήθηκε κώδικας για την εξαγωγή κατάταξης των χαρακτηριστικών και με το μοντέλο GBT [63]. Η υλοποίηση και ενσωμάτωση της συγκεκριμένης μεθόδου για την παρούσα μεταπτυχιακή εργασία έγινε σύμφωνα με τις βιβλιοθήκες μηχανικής μάθησης ανοικτού κώδικα της βιβλιοθήκης του scikit-learn.

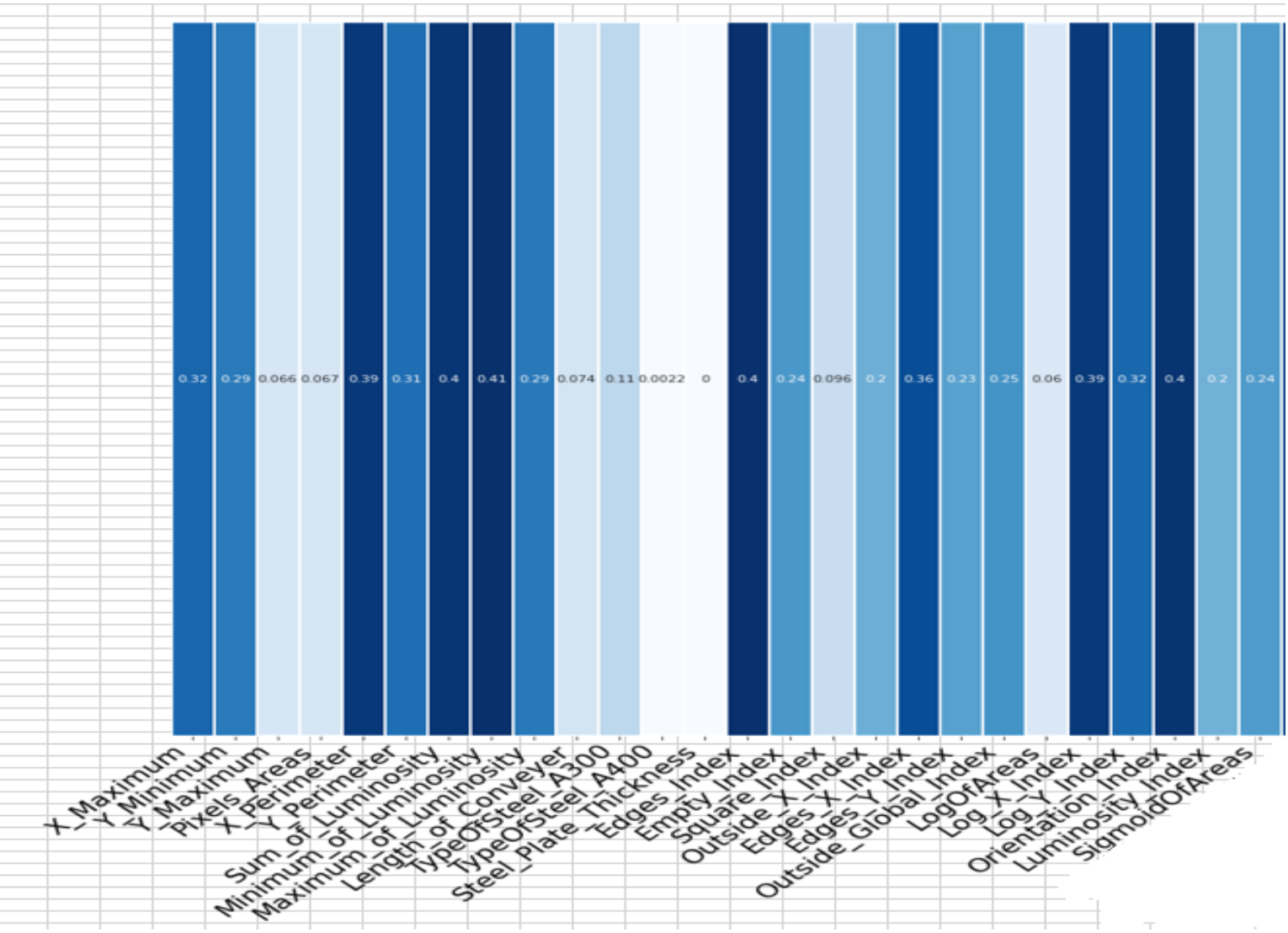
Τα αποτελέσματα εδώ συμφωνούν σε ποσοστό 70% για τα πρώτα 13 χαρακτηριστικά στην κατάταξη με το μοντέλο RF . Εδώ πρέπει να αναφερθεί ότι το μοντέλο RF έδειξε μεγαλύτερη συμφωνία με τα δημοσιευμένα αποτελέσματα. Τέλος, δεν έγινε ρύθμιση στα μοντέλα κατά την διαδικασία εξαγωγής της λίστας με τα καλύτερα χαρακτηριστικά.

Η παραγωγή λίστας κατάταξης χαρακτηριστικών με χρήση μοντέλων μηχανικής μάθησης αντιμετωπίζει την σημαντικότητα των χαρακτηριστικών ως προς όλες τις κλάσεις. Δηλαδή βρίσκει τα καλύτερα χαρακτηριστικά που μπορούν να διαχωρίσουν όλους τους τύπους ελαττωμάτων μαζί. Είναι σαφές ότι αν το πρόβλημα πολλών κλάσεων (multiclass classification) σπάσει σε πολλά δυικά προβλήματα (binary classification) τα χαρακτηριστικά που είναι σημαντικά ως προς τον διαχωρισμό των δυϊκών προβλημάτων μπορεί να είναι διαφορετικά.

Έτσι, έγινε χρήση της τεχνικής αμοιβαίας αβεβαιότητας (mutual information – information gain) για διασταύρωση των αποτελεσμάτων της λίστας κατάταξης των χαρακτηριστικών αφού το πρόβλημα πολλών κλάσεων έσπασε σε 3 δυϊκά προβλήματα για τις πρώτες 6 κλάσεις (Common Faults). Η τεχνική της αμοιβαίας αβεβαιότητας μετράει την ποσότητα της πληροφορίας που μια μεταβλητή περιέχει για μία άλλη μεταβλητή. Με άλλα λόγια, υπολογίζει την πληροφορία με όρους εντροπίας που περιέχει μία μεταβλητή/χαρακτηριστικό για την σωστή ταξινόμηση μίας κλάσης [64].

Τα αποτελέσματα αυτά περιγράφονται στο κεφάλαιο 7 όπου παρουσιάζεται αναλυτικά η δημιουργία ενός προσαρμοσμένου ταξινομητή που έγινε στα πλαίσια της παρούσας μεταπτυχιακής εργασίας.

Παρακάτω, παρουσιάζονται στην Εικόνα 13 ενδεικτικά κάποια αποτελέσματα για την κατάταξη των χαρακτηριστικών ως προς τις κλάσεις K_Scratch και Stains. Στο κεφάλαιο 7 παρουσιάζεται και σε αντίστοιχο διάγραμμα στην Εικόνα 44 η κατάταξη μετά από κατασκευή πολυωνυμικών χαρακτηριστικών (polynomial feature engineering) (n=2) για τα χαρακτηριστικά. Παρατηρείται ότι το χαρακτηριστικό για τον διαχωρισμό των δύο κλάσεων που περιέχει την μεγαλύτερη πληροφορία είναι το Minimum_of_Luminosity το οποίο βρίσκεται στην θέση 7 στην κατάταξη των χαρακτηριστικών από το RF και στην θέση 16 από το GBT. Επίσης, το χαρακτηριστικό Steel_Plate_Thickness που βρίσκεται σε υψηλή κατάταξη στα αποτελέσματα των μοντέλων, εδώ λαμβάνει πολύ χαμηλή τιμή.



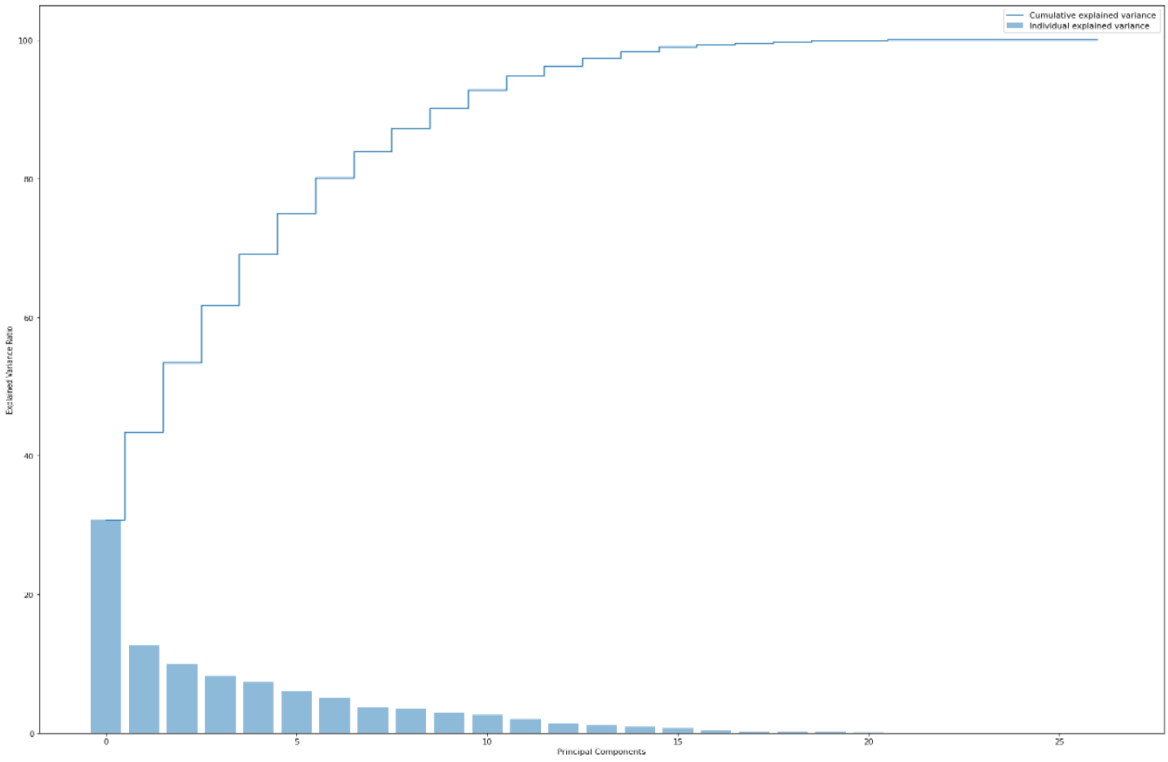
Εικόνα 13. Mutual Information για K_Scratch vs Stains

3.3 Οπτικοποίηση των δεδομένων και μείωση διάστασης

Στην παρούσα μεταπτυχιακή εργασία χρησιμοποιήθηκαν δύο τεχνικές μείωσης διάστασης του προβλήματος με σκοπό την ελάττωση των χαρακτηριστικών και την αξιολόγηση των μοντέλων μηχανικής μάθησης ως προς την απόδοσή τους. Η εφαρμογή μεθόδων μείωσης διαστατικότητας ξεκινά από το πρόβλημα της «κατάρας της διαστατικότητας» (curse of dimensionality) [65]. Σε πολλές εφαρμογές μηχανικής μάθησης τα χαρακτηριστικά είναι περισσότερα από τις παρατηρήσεις (observations) κάτι που καθιστά πολύ δύσκολη την εκπαίδευση των μοντέλων και κατ΄ επέκταση την πρόβλεψη αποτελεσμάτων.

Η πρώτη τεχνική που χρησιμοποιήθηκε είναι η ανάλυση κύριων συνιστωσών (Principal Component Analysis - PCA) που είναι ίσως η πιο διαδεδομένη τεχνική μείωσης διαστατικότητας. Ο ερευνητής που βρήκε την συγκεκριμένη τεχνική είναι ο Άγγλος μαθηματικός Karl Pearson και παρουσιάστηκε το 1901 [66]. Η βασική Αρχή της PCA είναι

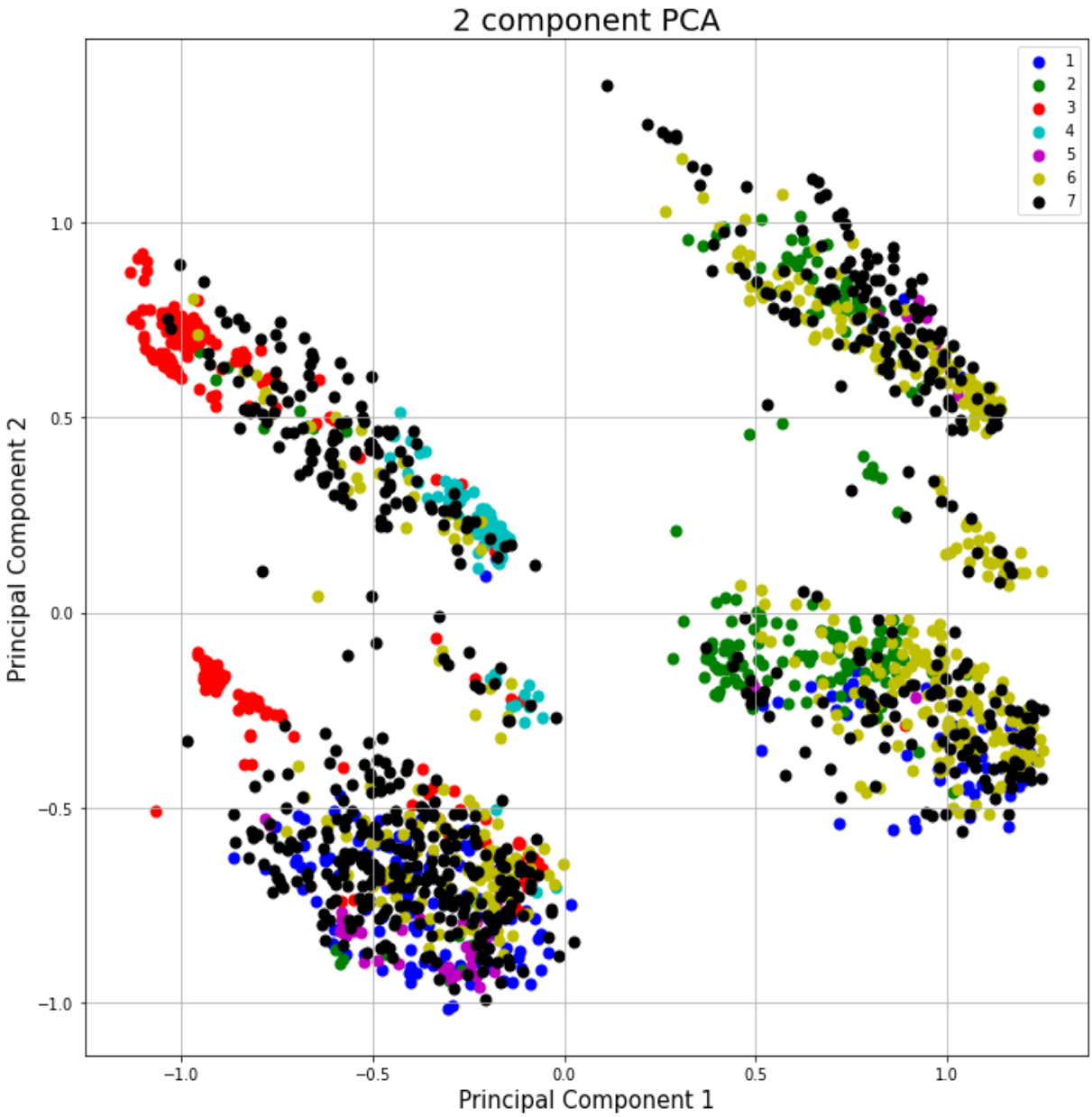
να αναδιατάσσει τα αρχικά δεδομένα και να παράγει καινούργια. Οι νέες μεταβλητές που προκύπτουν δεν έχουν καμία φυσική σημασία και αποτελούν γραμμικούς συνδυασμούς των αρχικών μεταβλητών έτσι ώστε να είναι ασυσχέτιστες μεταξύ τους και να περιέχουν όσο το δυνατό μεγαλύτερο μέρος της διακύμανσης (variance) των αρχικών δεδομένων. Αυτό σημαίνει ότι μεταβλητές με χαμηλή διακύμανση χάνονται, κάτι που δεν είναι πάντα σωστό, καθώς και αυτές οι μεταβλητές μπορεί να περιέχουν σημαντική πληροφορία. Παρακάτω, στην Εικόνα 14 γίνεται παρουσίαση του διαγράμματος Pareto για τις κύριες συνιστώσες.



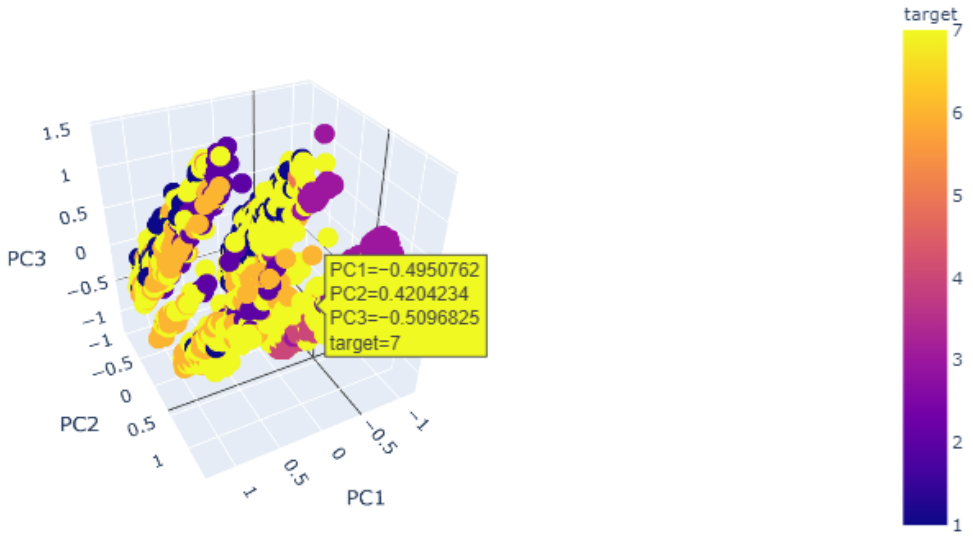
Εικόνα 14. Pareto diagram – PCA

Γενικά η εφαρμογή της PCA εφαρμόζεται για την μείωση της διαστατικότητας του προβλήματος. Βέβαια είναι ασφαλές να υποθέσουμε ότι κατά την ελάττωση των χαρακτηριστικών χάνεται πληροφορία που μπορεί απλά να μην είναι ανιχνεύσιμη στο παρόν σύνολο δεδομένων. Γενικά κατά αυτόν τον μετασχηματισμό των δεδομένων θεωρείται καλή πρακτική να κρατιούνται οι συνιστώσες (components) που περιέχουν το 95% της διακύμανσης και οι υπόλοιπες να αφαιρούνται. Στο διάγραμμα Pareto παρατηρούμε ότι από τα 27 components αρκούν τα 15 για να καλύψουν το ποσοστό αυτό. Εδώ πρέπει να αναφερθεί ότι η χρήση μόνο των 15 αυτών συνιστωσών για την εκπαίδευση των μοντέλων μηχανικής μάθησης είναι ένας αρκετά απλοϊκός τρόπος που συχνά δεν έχει τα καλύτερα αποτελέσματα καθώς χάνεται πληροφορία. Οπότε αν δεν έχουμε πάρα πολλά χαρακτηριστικά αυτό που σε αρχικό στάδιο μπορεί να γίνει είναι η δημιουργία ενός νέου συνόλου δεδομένων που θα περιέχει τα αρχικά μας δεδομένα συν τις κύριες συνιστώσες που καλύπτουν 95% του variance (original data + principal components).

Επίσης, πέρα από την μείωση διαστατικότητας οι τεχνικές αυτές καταφέρνουν κάτι αρκετά σημαντικό. Μπορούν και παρέχουν πληροφορία (insights) για τα δεδομένα στο πλαίσιο της ανάλυσης μέσω οπτικοποίησης. Παρακάτω, στην Εικόνα 15 και στην Εικόνα 16 παρουσιάζονται διαγράμματα για τις 2 πρώτες συνιστώσες (αντιστοιχούν στο περίπου 45% της διακύμανσης) και για τις 3 πρώτες συνιστώσες (αντιστοιχούν στο περίπου 55% της διακύμανσης). Επειδή το εξεταζόμενο πρόβλημα είναι πολυδιάστατο (multidimensional problem , n=27) και η οπτικοποίηση των δεδομένων είναι αδύνατη σε μεγάλους αριθμούς διαστάσεων (n>4 – Tesseract) αυτό που επιτυγχάνουμε με τις συγκεκριμένες τεχνικές είναι να συμπτύσσουμε πληροφορία σε μικρότερες διαστάσεις.



Εικόνα 15. Διάγραμμα PCA - δύο συνιστώσες



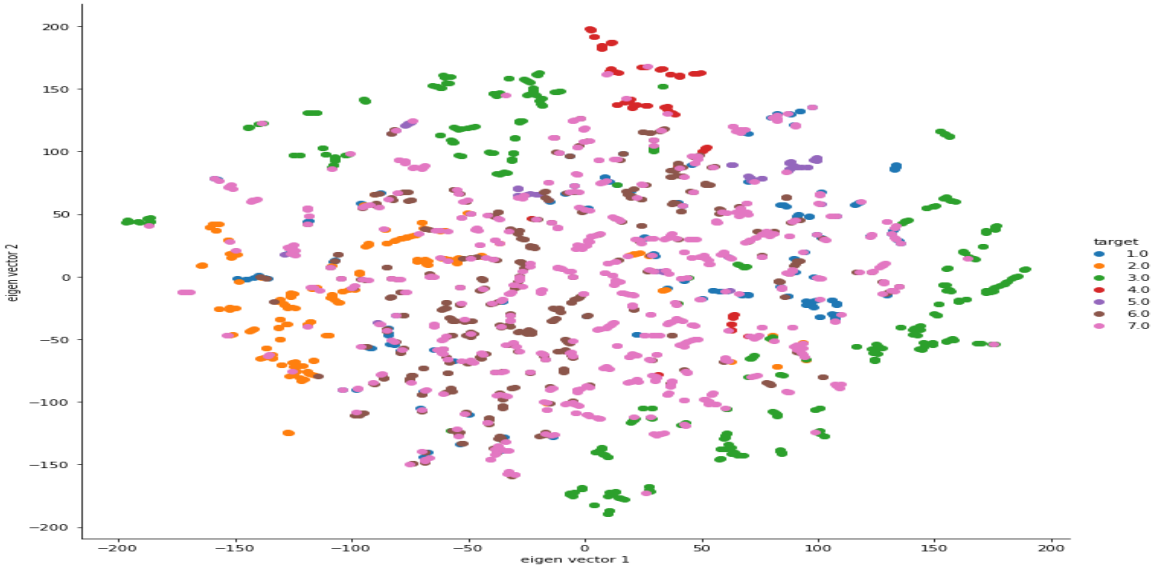
Εικόνα 16. Διάγραμμα PCA - τρεις συνιστώσες

Πίνακας 9. Πρώτες ένδεκα τιμές των συνιστωσών για την κλάση 1

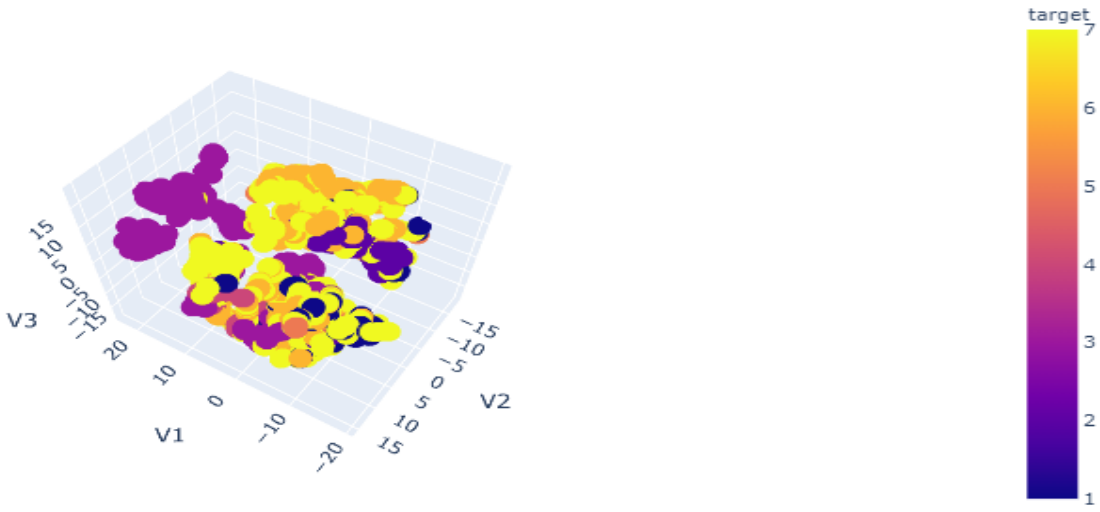
	PC1	PC2	PC3	target
0	0.7391879	-0.2610762	0.8429568	1
1	1.0411911	-0.3956142	0.2186167	1
2	1.1417501	-0.3986473	-0.0457337	1
3	-0.27207	-0.9031682	-0.2421922	1
4	-0.5715842	-0.8772729	0.276824	1
5	-0.5909599	-0.6737849	0.2283981	1
6	-0.6479241	-0.74084	0.5153254	1
7	-0.2491762	-0.5978107	-0.2414704	1
8	-0.3069768	-0.6002995	-0.118938	1
9	-0.6656409	-0.653516	0.6190285	1
10	-0.4912258	-0.5724953	0.2298722	1

Πέρα της PCA χρησιμοποιήθηκε μία τεχνική που ονομάζεται T-SNE (t-Distributed Stochastic Neighbor Embedding). Η συγκεκριμένη μέθοδος ανακαλύφθηκε από τον Laurens van der Maaten ο οποίος αποτελεί και τον Research Director του Facebook AI Research (FAIR) στην Νέα Υόρκη [67]. Η t-SNE αποτελεί μία τεχνική για τη γραφική αναπαράσταση και οπτικοποίηση πολυδιάστατων δεδομένων. Δηλαδή αποτελεί μέθοδο σύμπτυξης πολυδιάστατων δεδομένων σε χώρους μικρότερων διαστάσεων. Σε αντίθεση με την PCA, η συγκεκριμένη τεχνική είναι μη γραμμική τεχνική μείωσης της διαστατικότητας και μειώνει τις διαστάσεις μέχρι 3 και όχι παραπάνω (πχ 4 διαστάσεις). Επειδή η συγκεκριμένη τεχνική έχει υπέρ - παραμέτρους και απαιτείται να γίνει ρύθμιση, δημιουργείται ζήτημα ως προς τους χρόνους εκτέλεσης. Οπότε συνηθίζεται η υλοποίηση αυτή να γίνεται σε γραφικούς

επιταχυντές GPUs (CUDA). Στην παρούσα μεταπτυχιακή εργασία έγινε προσπάθεια της ρύθμισης των παραμέτρων της μέσω διαδικασίας δοκιμής και λάθους (trial and error) και στις Εικόνες 17 και 18 παρουσιάζονται τα διαγράμματα για δύο και για τρεις διαστάσεις. Η ανάπτυξη και εδώ έγινε μέσω της βιβλιοθήκης του scikit [68].



Εικόνα 17. Διάγραμμα T-SNE - δύο vectors



Εικόνα 18. Διάγραμμα t-SNE - τρία vectors

Αυτό που παρατηρείται αρχικά από τα διαγράμματα και με τις δύο μεθόδους είναι ότι υπάρχουν κάποια υπερ-επίπεδα (hyper-planes) και κάποιες κλάσεις φαίνεται να διαχωρίζονται από κάποιες άλλες. Το πρόβλημα που δημιουργείται είναι ότι υπάρχει μεγάλη διασπορά της 7ης κλάσης Other Faults που έχει και τις περισσότερες παρατηρήσεις σε όλο των φάσμα των τιμών των υπόλοιπων κλάσεων. Αυτό σημαίνει ότι η συγκεκριμένη κλάση θα

είναι δύσκολο να ταξινομηθεί, καθώς τα μοντέλα μηχανικής μάθησης δεν θα μπορούν να τη διαχωρίσουν ως προς τις υπόλοιπες. Επιπροσθέτως, εδώ μπορούμε να καταλάβουμε ότι η συγκεκριμένη κλάση μπορεί να περιέχει δεδομένα – τύπους ελαττωμάτων από τις άλλες κλάσεις (Common Faults) και για αυτό οι τιμές της να πέφτουν πάνω στις τιμές των άλλων ελαττωμάτων. Δυστυχώς, δεν υπάρχει σχετική πληροφορία από το ερευνητικό κέντρο Seimeon από όπου προέρχονται τα δεδομένα, καθώς αν κάτι τέτοιο ίσχυε θα μπορούσε το συγκεκριμένο πρόβλημα να αντιμετωπιστεί τελείως διαφορετικά. Τέλος θα πρέπει να αναφερθεί ότι τα διαγράμματα της t-SNE άλλαζαν σε μεγάλο βαθμό με την εισαγωγή διαφορετικών υπέρ - παραμέτρων (perplexity , learning rate, early exaggeration) και τα διαγράμματα που παρουσιάζονται είναι αποτέλεσμα αρκετών διαφορετικών συνδυασμών.

3.4 Μετασχηματισμοί Δεδομένων

Στο συγκεκριμένο κεφάλαιο θα γίνει περιγραφή των μετασχηματισμών των δεδομένων που πραγματοποιήθηκαν για την εισαγωγή τους στα μοντέλα μηχανικής μάθησης. Πέρα από τους μετασχηματισμούς με την ελάττωση διάστασης με PCA και t-SNE υλοποιήθηκαν τεχνικές κατασκευής - παραγωγής χαρακτηριστικών (feature engineering) που ανήκουν στο κομμάτι της προ-επεξεργασίας των δεδομένων. Ο σκοπός των συγκεκριμένων τεχνικών ήταν η επίτευξη υψηλότερων αποδόσεων από τα μοντέλα μηχανικής μάθησης.

3.4.1 Κατασκευή πολυωνυμικών χαρακτηριστικών (Polynomial Feature Engineering)

Η συγκεκριμένη τεχνική αποτελεί από τις πιο βασικές τεχνικές κατασκευής χαρακτηριστικών που υπάρχουν καθώς αυτό που περιλαμβάνει η συγκεκριμένη υλοποίηση είναι η προσθήκη πολυωνύμων των χαρακτηριστικών στα δεδομένα. Η λογική πίσω από αυτό είναι ότι η προσθήκη πολυωνύμων των δεδομένων και η κατασκευή ενός καινούριου συνόλου χαρακτηριστικών μπορεί να βοηθήσει τα μοντέλα μηχανικής μάθησης ως προς την αναγνώριση μη-γραμμικών μοτίβων (non-linear patterns) και τελικά τον καλύτερο διαχωρισμό και ταξινόμηση των ελαττωμάτων [69]. Παρακάτω, παρουσιάζονται μερικοί βασικοί πολυωνυμικοί και εκθετικοί μετασχηματισμοί.

$$x_i^n \text{ , } x_i x_{i+1} \text{, πολυωνυμικός μετασχηματισμός}$$
$$e^{x_i} \text{, εκθετικός μετασχηματισμός}$$
$$\log(x_i) \text{ , λογαριθμικός μετασχηματισμός}$$
$$\text{όπου } i = 1 \text{ μέχρι } 27 \text{ και } n : \text{ο εκθέτης του πολυωνύμου} \quad (9)$$

Στην συγκεκριμένη εργασία προστέθηκαν μόνο τα πολυώνυμα 2ου βαθμού (x_i^2) στο αρχικό σύνολο δεδομένων. Έγιναν και δοκιμές για του 3ου βαθμού αλλά τελικά δεν χρησιμοποιήθηκαν. Γενικά η πρόσθεση πολλών πολυωνύμων υψηλού βαθμού δεν είναι ότι

καλύτερο αφού αφενός αυξάνονται οι διαστάσεις του προβλήματος και αφετέρου χάνεται η φυσική σημασία των χαρακτηριστικών, ειδικά για υψηλά πολυώνυμα (n>4).

Το νέο σύνολο δεδομένων που προέκυψε μετά από την προσθήκη των x_i^2 είναι σαφές ότι θα περιέχει 54 χαρακτηριστικά (27+27) και τις 7 κλάσεις. Είναι κατανοητό ότι η προσθήκη των νέων χαρακτηριστικών μπορεί να βοηθήσει τους αλγορίθμους μηχανικής μάθησης ως προς τις αποδόσεις τους, καθώς από την ανάλυση των δεδομένων δεν υπάρχει καμία ένδειξη ότι σε χαμηλές διαστάσεις (λιγότερες του αρχικού dataset) μπορούν να παραχθούν ακριβή αποτελέσματα. Η προσθήκη μη-γραμμικών όρων των χαρακτηριστικών είναι κάτι που δοκιμάστηκε και βελτίωσε την απόδοση των μοντέλων.

Πίνακας 10. Νέο σύνολο δεδομένων με 54 χαρακτηριστικά και 7 κλάσεις

	X_Minimum	X_Maximum	Y_Minimum	Y_Maximum	Pixels_Areas	X_Perimeter	Y_Perimeter	Sum_of_Luminosity	Minimum_of_Luminosity
0	42	50	270900	270944	267	17	44	24220	76
1	645	651	2538079	2538108	108	10	30	11397	84
2	829	835	1553913	1553931	71	8	19	7972	99
3	853	860	369370	369415	176	13	45	18996	99
4	1289	1306	498078	498335	2409	60	260	246930	37
...	...	...	...	...	...	...	...	...	...
1936	249	277	325780	325796	273	54	22	35033	119
1937	144	175	340581	340598	287	44	24	34599	112
1938	145	174	386779	386794	292	40	22	37572	120
1939	137	170	422497	422528	419	97	47	52715	117
1940	1261	1281	87951	87967	103	26	22	11682	101

1941 rows × 61 columns

3.4.2 RBF Kernel και PCA

Ένας άλλος γνωστός μετασχηματισμός που έγινε στην μεταπτυχιακή εργασία είναι ο RBF Kernel (radial basis function) ή Gaussian Kernel [70]. Ο συγκεκριμένος μετασχηματισμός των δεδομένων έγινε στο πλαίσιο υλοποίησης του προσαρμοσμένου ταξινομητή που παρουσιάζεται στο κεφάλαιο 7. Η εξίσωση για την παραγωγή του συγκεκριμένου kernel είναι η εξής:

$$k(x, y) = \exp(-\gamma \|x - y\|^2) \text{ όπου } x, y \text{ τα διανύσματα των χαρακτηριστικών και } \gamma = \sigma^2 \quad (10)$$

Γενικά, χρησιμοποιώντας τον παραπάνω μετασχηματισμό κατασκευάζεται ένα καινούριο σύνολο δεδομένων n x n, όπου n ο αριθμός των παρατηρήσεων του προβλήματος. Αυτό σημαίνει ότι αριθμός των στηλών (χαρακτηριστικά) με τον αριθμό των γραμμών (παρατηρήσεις) είναι ίδιος κάτι που δημιουργεί πρόβλημα ως προς την κατάρα της διαστατικότητας (curse of dimensionality). Στην διαγώνιο έχουμε μονάδες καθώς η απόσταση είναι μηδενική και υψώνεται στον εκθετικό βαθμό.

Η χρήση ad hoc του συγκεκριμένου kernel για την είσοδο των μετασχηματισμένων δεδομένων στα μοντέλα μηχανικής μάθησης δεν εφαρμόστηκε, αφού αυτό που έγινε είναι η χρήση της ελάττωσης διάστασης με PCA μετά τον συγκεκριμένο μετασχηματισμό. Πιο συγκεκριμένα, αυτό που έγινε ήταν να υλοποιηθεί κώδικας για την εφαρμογή του μετασχηματισμού των δεδομένων εκπαίδευσης με RBF Kernel και την παραγωγή ενός νέου dataset με nxn διαστάσεις. Στην συνέχεια έγινε ο μετασχηματισμός μέσω PCA και κρατήθηκαν τα 20 πρώτα χαρακτηριστικά που έφταναν το 95% της διακύμανσης. Τα αποτελέσματα δεν ήταν τα πιο ικανοποιητικά μετά την εφαρμογή της μάθησης οπότε και τελικά δεν χρησιμοποιήθηκαν. Η εφαρμογή και εδώ έγινε μέσω της βιβλιοθήκης του scikit [71].

Κεφάλαιο 4: ΑΝΑΠΤΥΞΗ ΜΟΝΤΕΛΩΝ

4.1 Εισαγωγή

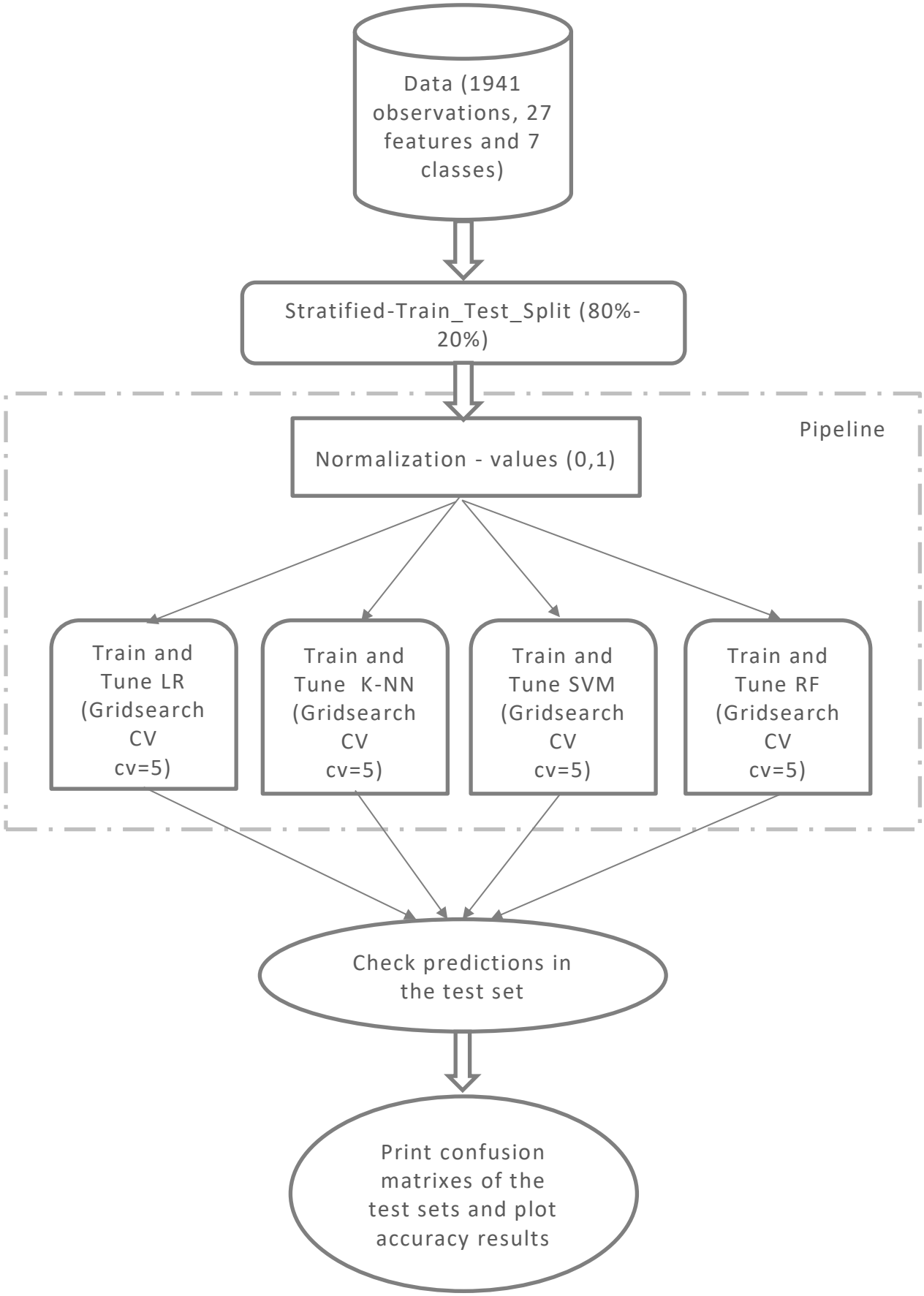
Στο συγκεκριμένο κεφάλαιο γίνεται αναφορά των σεναρίων προ-επεξεργασίας των δεδομένων που χρησιμοποιήθηκαν για την ανάπτυξη των μοντέλων μηχανικής μάθησης με ρύθμιση των υπέρ-παραμέτρων τους. Τα μοντέλα που αναπτύχθηκαν αξιολογήθηκαν ως προς την απόδοση τους για κάθε διαφορετικό σενάριο. Πέρα από το κομμάτι της προ-επεξεργασίας των δεδομένων σε όλα τα μοντέλα, εκτός από αυτά που χρησιμοποιήθηκε η μέθοδος PCA, πραγματοποιήθηκε ρύθμιση μέσω της Gridsearch CV.

4.2 Μέθοδοι προ-επεξεργασίας των δεδομένων

Θα αναλυθούν 4 διαφορετικοί τρόποι προ-επεξεργασίας των δεδομένων που πραγματοποιήθηκαν με σκοπό την παραγωγή αποτελεσμάτων για την σύγκριση των 4 μοντέλων. Η προ-επεξεργασία των δεδομένων αποτελεί πολύ σημαντικό κομμάτι της μηχανικής μάθησης καθώς η σωστή εφαρμογή διαφορετικών τεχνικών προετοιμασίας των δεδομένων και των μετασχηματισμών τους μπορεί να οδηγήσει σε καλύτερα αποτελέσματα από τα μοντέλα που χρησιμοποιούνται. Από την ανάλυση του προβλήματος έχει γίνει κατανοητό ότι η τελευταία κλάση των ελαττωμάτων είναι προβληματική καθώς το εύρος τιμών της απλώνεται σε όλες τις άλλες κλάσεις οπότε είναι ασφαλές να υποθέσουμε ότι η προ-επεξεργασία των δεδομένων είναι απαραίτητη, καθώς η είσοδος των πρωτογενών τιμών τους στα μοντέλα δεν μπορεί να έχει τα καλύτερα δυνατά αποτελέσματα.

4.2.1 Απλή ρύθμιση υπέρ – παραμέτρων

Το συγκεκριμένο σενάριο αποτελεί το πιο απλό που πραγματοποιήθηκε στο πλαίσιο αυτής της μεταπτυχιακής εργασίας. Τα δεδομένα αρχικά διαβάστηκαν και τοποθετήθηκαν σε dataframes (pandas). Σε επόμενο στάδιο, χωρίστηκαν σε 2 υποσύνολα, το υποσύνολο εκπαίδευσης (train set) που περιλάμβανε το 80% των δεδομένων και το υποσύνολο ελέγχου (test set) που περιλάμβανε το υπόλοιπο 20% των δεδομένων. Κατασκευάστηκε pipeline στο οποίο τα δεδομένα εκπαίδευσης κανονικοποιούντουσαν (normalization) πριν την είσοδο τους στα μοντέλα. Για όλα τα μοντέλα πραγματοποιήθηκε ρύθμιση των παραμέτρων τους με την GridSearch CV με cv=5 (cross – validations). Τα pipelines στην python είναι αυτοματοποιημένα workflows που βοηθούν στην μηχανική μάθηση ως προς τον όγκο του κώδικα σε αρχικό επίπεδο αλλά και ως προς την ακολουθία των μετασχηματισμών που πραγματοποιούνται. Το pipeline κατασκευάστηκε για να υπάρχει συμφωνία των δεδομένων και της κανονικοποίησης τους ως προς το κάθε fold. Τα μοντέλα που χρησιμοποιήθηκαν για την ταξινόμηση είναι τα Random Forest, Support Vector Machines, k-Nearest-Neighbor και Logistic Regression. Παρακάτω, στην Εικόνα 19 παρουσιάζεται το λογικό διάγραμμα του κώδικα.



Εικόνα 19. Διάγραμμα κώδικα με απλή ρύθμιση παραμέτρων

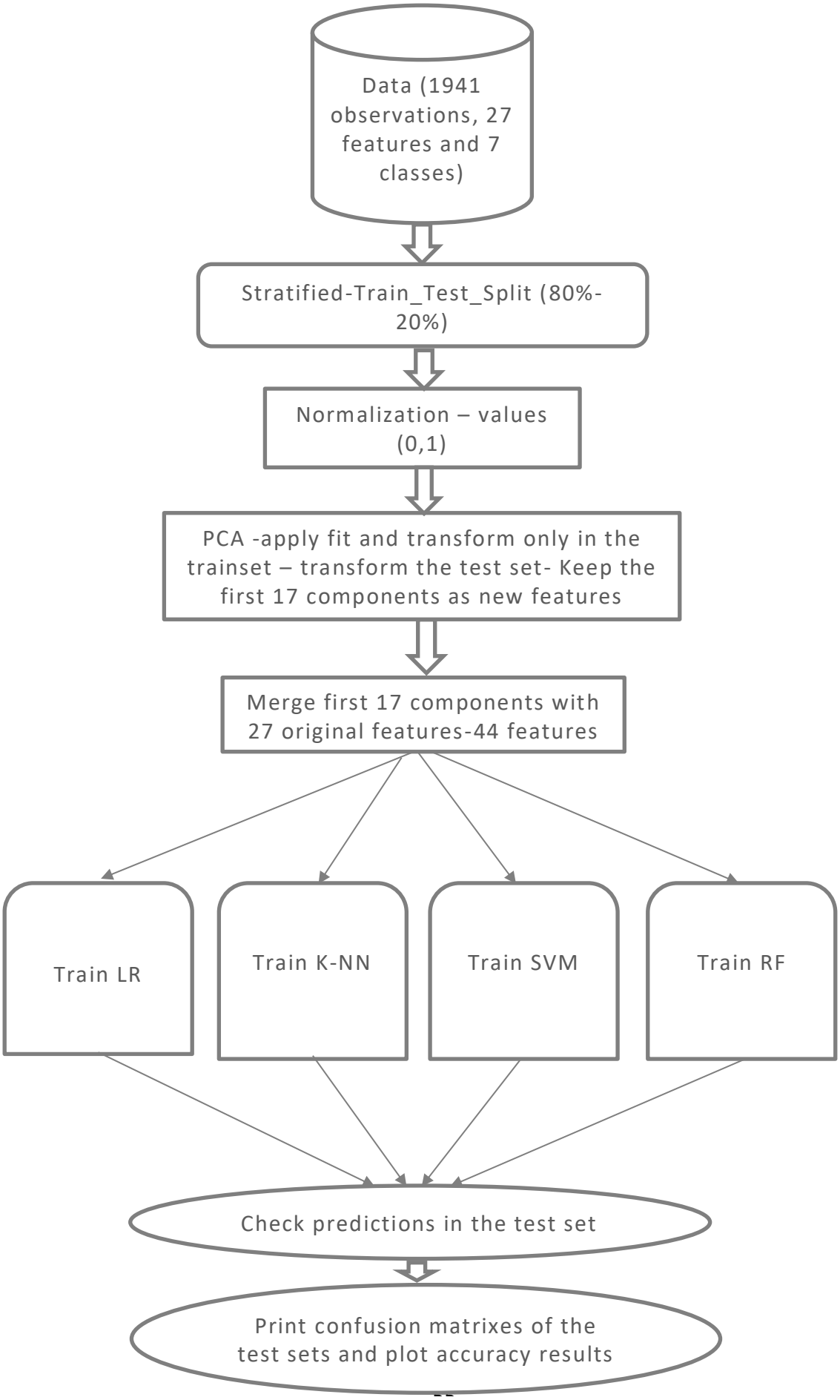
4.2.2 Μείωση Διαστατικότητας και εφαρμογή μοντέλων

Στην συγκεκριμένη ενότητα υλοποιήθηκε το σενάριο της μείωσης διαστατικότητας με χρήση της μεθόδου PCA στο κομμάτι της προετοιμασίας των δεδομένων (preproccessing). Κρατήθηκαν οι 17 πρώτες κύριες συνιστώσες (components).

Τα δεδομένα χωρίστηκαν σε υποσύνολο εκπαίδευσης και υποσύνολο ελέγχου με ποσοστό 80% - 20% (stratified split), στη συνέχεια κανονικοποιήθηκαν (MinMaxScaler) και εφαρμόστηκε η μέθοδος PCA. Πριν την είσοδο τους στα μοντέλα οι κύριες συνιστώσες ενώθηκαν με το αρχικό σετ δεδομένων και συνεπώς τελικά προέκυψαν 44 χαρακτηριστικά. Στην συγκεκριμένη περίπτωση τα μοντέλα ρυθμίστηκαν με επαναληπτική διαδικασία δοκιμής-και-λάθους. Το λογικό διάγραμμα του κώδικα παρατίθεται στην Εικόνα 20. Η t-SNE δεν χρησιμοποιήθηκε καθώς το scikit δεν έχει αναπτύξει σχετική βιβλιοθήκη για μετασχηματισμό δεδομένων (transform). Η t-SNE χρησιμοποιείται μόνο σε όλο το σύνολο δεδομένων (fit_transform) κάτι που είναι καλό για οπτικοποίηση των δεδομένων και μόνο. Επιπλέον η t-SNE βγάζει το πολύ 3 vectors (μέγιστος χώρος μειωμένων διαστάσεων = 3), που σημαίνει μεγάλη απώλεια πληροφορίας για τα μοντέλα αφού έχουμε αρχικά 27 χαρακτηριστικά.

Πίνακας 11. Πρώτες 10 τιμές των 17 συνιστωσών

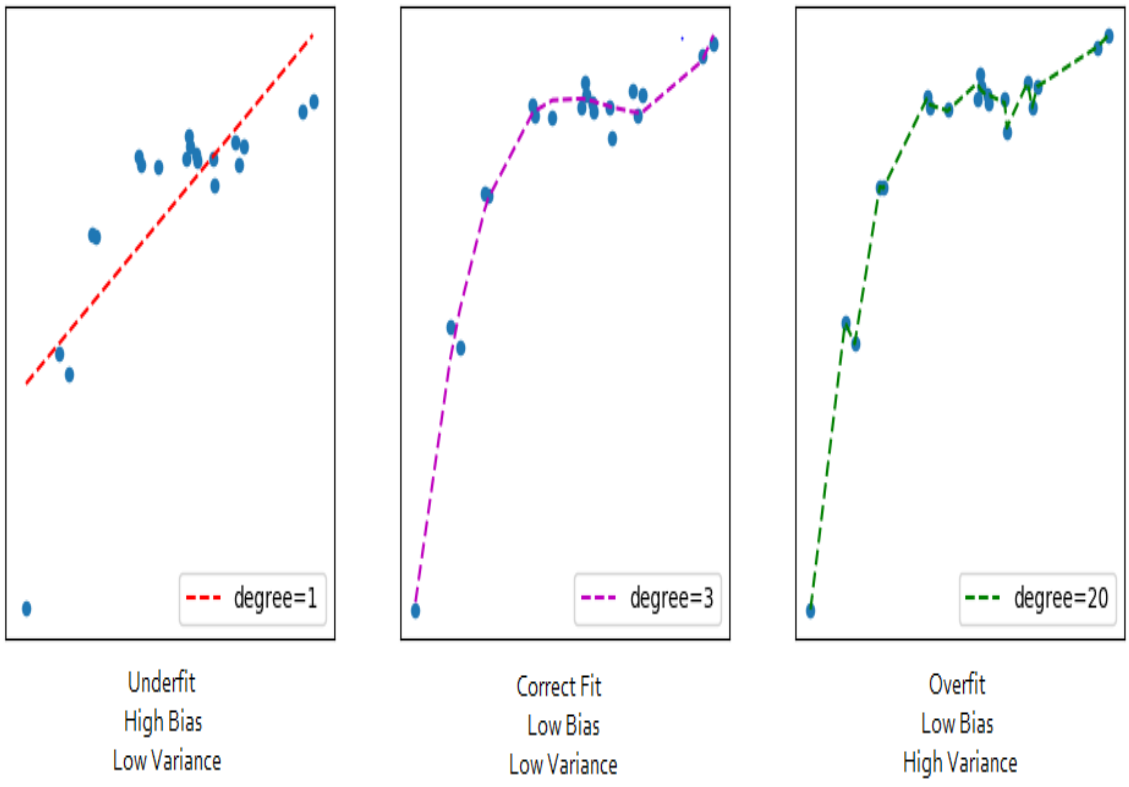
PCA 0	PCA 1	PCA 2	PCA 3	PCA 4	PCA 5	PCA 6	PCA 7	PCA 8	PCA 9	PCA 10	PCA 11	PCA 12	PCA 13	PCA 14	PCA 15	PCA 16
-1.01	0.59	-0.31	0.06	0.41	0.20	0.21	0.12	0.19	-0.20	-0.22	0.11	0.02	0.06	0.01	0.01	-0.02
-0.91	-0.15	0.64	-0.05	-0.11	0.64	0.06	-0.02	-0.13	0.05	0.18	0.15	0.00	0.00	0.02	0.05	-0.01
-0.82	0.35	0.14	-0.49	-0.07	0.26	-0.19	-0.31	-0.07	-0.12	0.11	0.17	0.02	-0.10	-0.03	-0.02	-0.01
-0.22	-0.69	-0.51	0.09	-0.22	0.10	-0.22	0.16	-0.06	0.11	-0.02	-0.12	0.02	0.03	-0.03	-0.03	0.01
-0.19	0.17	-1.10	0.09	-0.43	0.05	-0.16	-0.18	0.08	-0.16	-0.02	-0.02	0.01	-0.04	-0.10	0.02	-0.01
0.63	0.98	0.12	-0.35	-0.21	-0.42	-0.23	0.09	0.00	0.13	0.04	0.10	-0.21	-0.03	0.04	0.02	0.03
-0.21	-0.88	-0.43	0.21	-0.41	-0.23	-0.38	-0.04	-0.12	-0.25	0.11	0.09	-0.07	0.10	0.01	0.03	-0.04
-0.50	-0.55	-0.04	0.17	0.53	0.35	-0.26	0.31	0.17	0.15	-0.04	-0.16	0.04	-0.09	0.03	0.00	0.01
-1.00	0.75	0.19	0.05	-0.10	-0.04	0.06	0.07	-0.05	0.14	-0.17	-0.01	0.01	0.01	-0.01	0.00	-0.01
-0.63	-0.77	0.31	0.59	0.09	-0.10	-0.16	-0.02	0.00	0.02	-0.36	0.13	0.21	0.14	0.03	0.20	-0.06



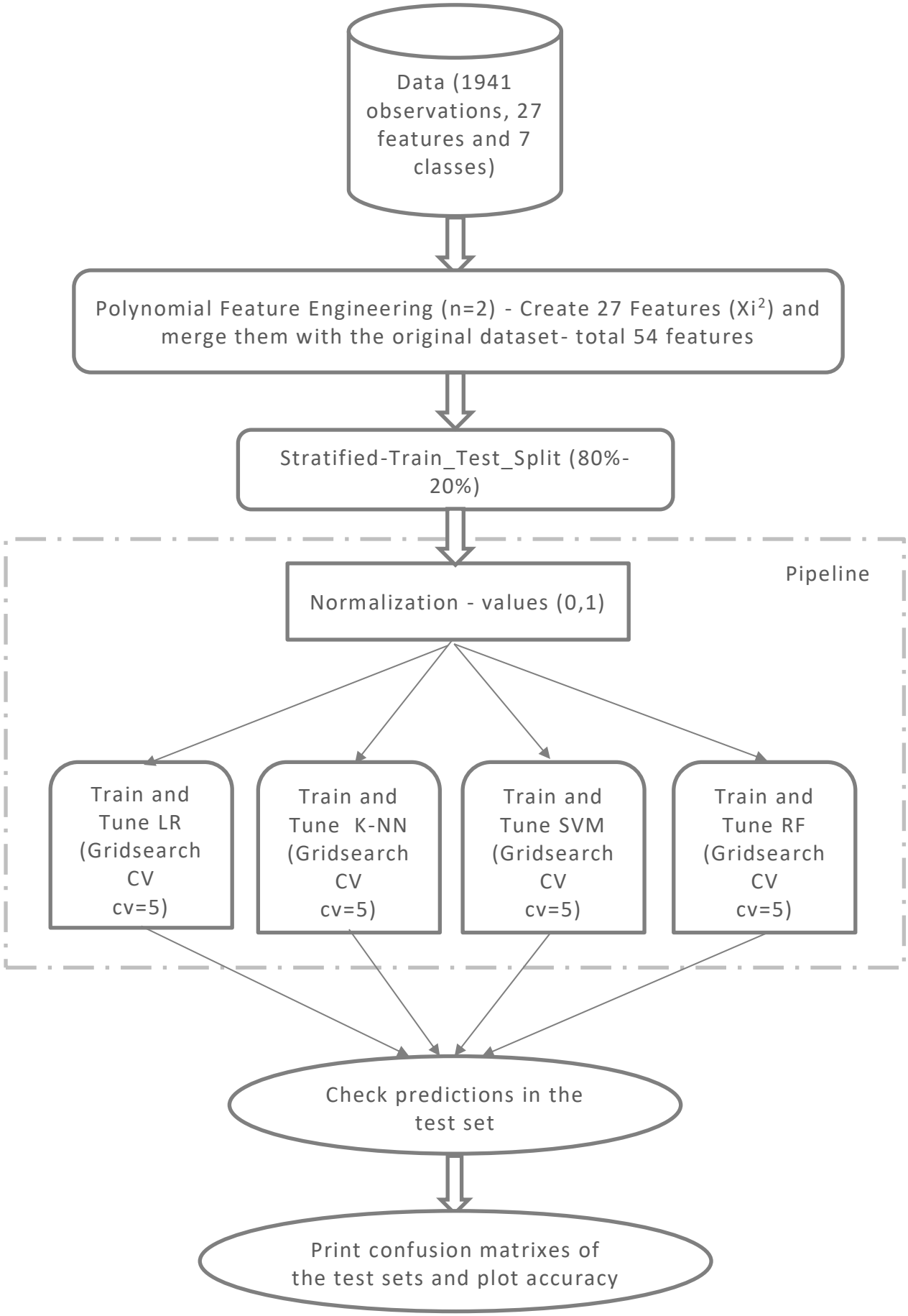
Εικόνα 20. Διάγραμμα με PCA

4.2.3 Πολυωνυμική προ-επεξεργασία των δεδομένων

Στο συγκεκριμένο κεφάλαιο έγινε πολυωνυμική προ-επεξεργασία των δεδομένων. Προστέθηκαν πολυώνυμα δευτέρου βαθμού (Xi ^{2} – όπου Xi τα 27 χαρακτηριστικά) στο αρχικό σετ δεδομένων. Οπότε προέκυψε ένα νέο σετ δεδομένων με 54 χαρακτηριστικά. Είναι σαφές από την ανάλυση των δεδομένων ότι για να επιτευχθεί καλύτερη ταξινόμηση πρέπει να εφαρμοστούν μετασχηματισμοί που θα μεγαλώσουν την διάσταση του προβλήματος. Σε αυτό το σημείο, αναφέρουμε ότι πραγματοποιήθηκαν και δοκιμές με προσθήκη πολυώνυμων τρίτου βαθμού αλλά τελικά δεν χρησιμοποιήθηκαν για την αναπαραγωγή αποτελεσμάτων. Ως προς την διαδικασία που ακολουθήθηκε τα δεδομένα χωρίστηκαν σε train set και test set με ποσοστό 80% - 20% (stratified split). Κατασκευάστηκε συνάρτηση για αυτοματοποιήμενη ακολουθία μετασχηματισμών (pipeline) στην οποία τα δεδομένα για την εκπαίδευση κανονικοποιούντουσαν (normalization) πριν την είσοδος τους στα μοντέλα. Για όλα τα μοντέλα πραγματοποιήθηκε ρύθμιση των παραμέτρων τους με την GridSearch CV με cv=5 (cross – validations) .Τα μοντέλα που χρησιμοποιήθηκαν για την ταξινόμηση είναι τα Random Forest, Support Vector Machines, k-Nearest-Neighbor και Logistic Regression. Παρακάτω, στην Εικόνα 22 παρουσιάζεται και σε διάγραμμα ο κώδικας.



Εικόνα 21. Polynomial feature engineering για n=1, n=3, n=20



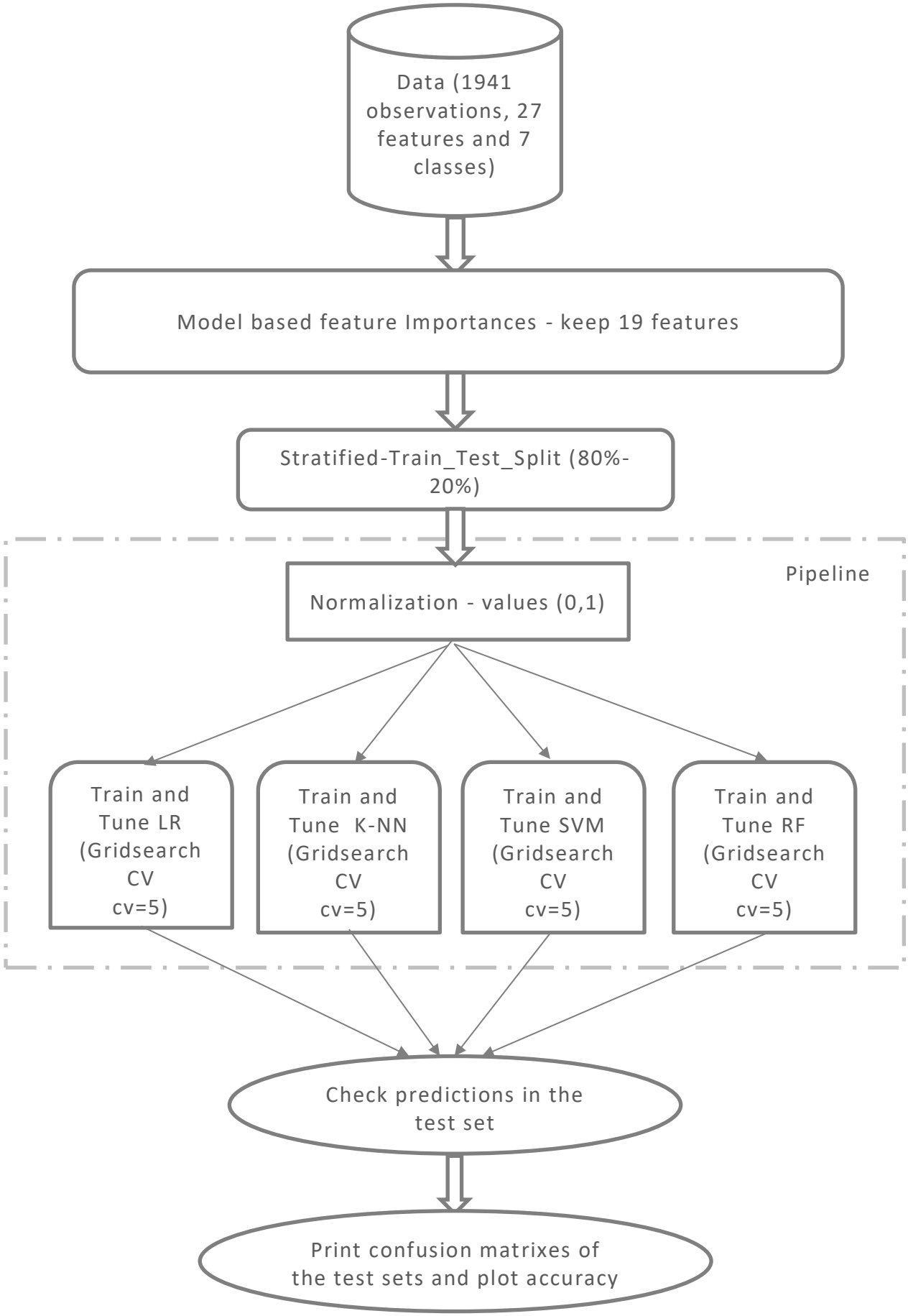
Εικόνα 22. Διάγραμμα με polynomial feature enginnering

4.2.4 Επιλογή Χαρακτηριστικών και Ρύθμιση Παραμέτρων

Στην συγκεκριμένη ενότητα επιλέχθηκαν μερικά χαρακτηριστικά για την είσοδο στα μοντέλα καθώς δεν χρησιμοποιήθηκαν και τα 27. Η επιλογή των χαρακτηριστικών που αφαιρέθηκαν έγινε σύμφωνα με τα αποτελέσματα για την σημαντικότητα τους από το μοντέλο RF (model based feature Importances) που παρουσιάστηκε στην Ενότητα 3.2. Για την σημαντικότητα των συγκεκριμένων μεταβλητών υπάρχει και σε αρκετά μεγάλο ποσοστό συμφωνία σχετικά με την βιβλιογραφία [30]. Οπότε μετά την αφαίρεση των 8 χαρακτηριστικών που μπορούν να φανούν στον Πίνακα 12, τελικά προέκυψαν 19 χαρακτηριστικά. Ακολουθήθηκε η ίδια διαδικασία ως προς τα βήματα που περιγράφτηκαν προηγουμένως και παρακάτω παρουσιάζεται στην Εικόνα 23 σε διάγραμμα ο κώδικας που υλοποιήθηκε.

Πίνακας 12. Χαρακτηριστικά χαμηλής σημαντικότητας σύμφωνα με το RF

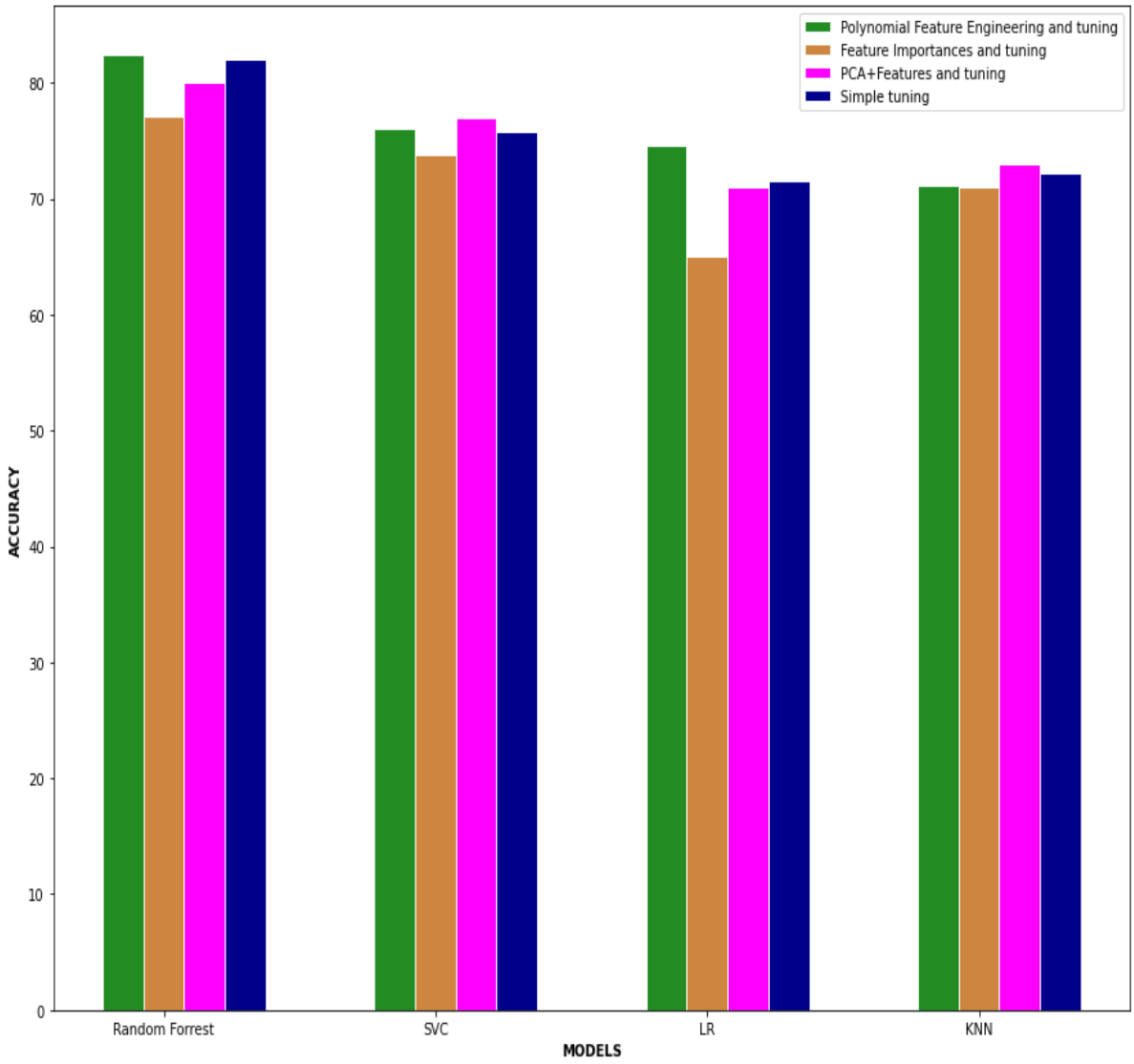
Features that droped based on RF score importance
Edges_Y_Index
Maximum_of_Luminosity
Edges_X_Index
Y_Perimeter
TypeOfSteel_A400
Log_Y_Index
TypeOfSteel_A300
Outside_Global_Index



Εικόνα 23. Κώδικας για feature importances - ml models

4.3 Αποτελέσματα μοντέλων

Στο παρόν κεφάλαιο θα παρουσιαστούν τα αποτελέσματα για όλα τα διαφορετικά σενάρια προ-επεξεργασίας των δεδομένων που έχουν αναλυθεί. Συγκεκριμένα όμως για κάθε σενάριο και κάθε διαφορετικό μοντέλο παρουσιάζεται σε πρωτογενή μορφή από την Python η έκθεση ταξινόμησης που περιλαμβάνει τις μετρικές: ορθότητα, ακρίβεια, ανάκληση , αρμονικός μέσος και τους πίνακες σύγχυσης (confusion matrix) στο Παράρτημα 4. Παρακάτω, εμφανίζονται στην Εικόνα 24 τα συγκεντρωτικά αποτελέσματα των μοντέλων (matplotlib) με όλες τις διαφορετικές προ-επεξεργασίες των δεδομένων που πραγματοποιήθηκαν. Παρατηρείται ότι το καλύτερο μοντέλο είναι, σε όλες τις περιπτώσεις, το Random Forest. Επιπλέον, η καλύτερη επίδοση για το τεστ ελέγχου όσον αφορά τη μετρική της συνολικής ακρίβειας, προκύπτει για την περίπτωση της κατασκευής πολυωνυμικών χαρακτηριστικών (n = 2) και είναι 0.825. Τέλος, επισυνάπτεται ο συγκεντρωτικός Πίνακας 13 με τον αριθμό των συνδυασμών για την ρύθμιση παραμέτρων, τους χρόνους για τη ρύθμιση καθώς και τις καλύτερες παραμέτρους που προέκυψαν.



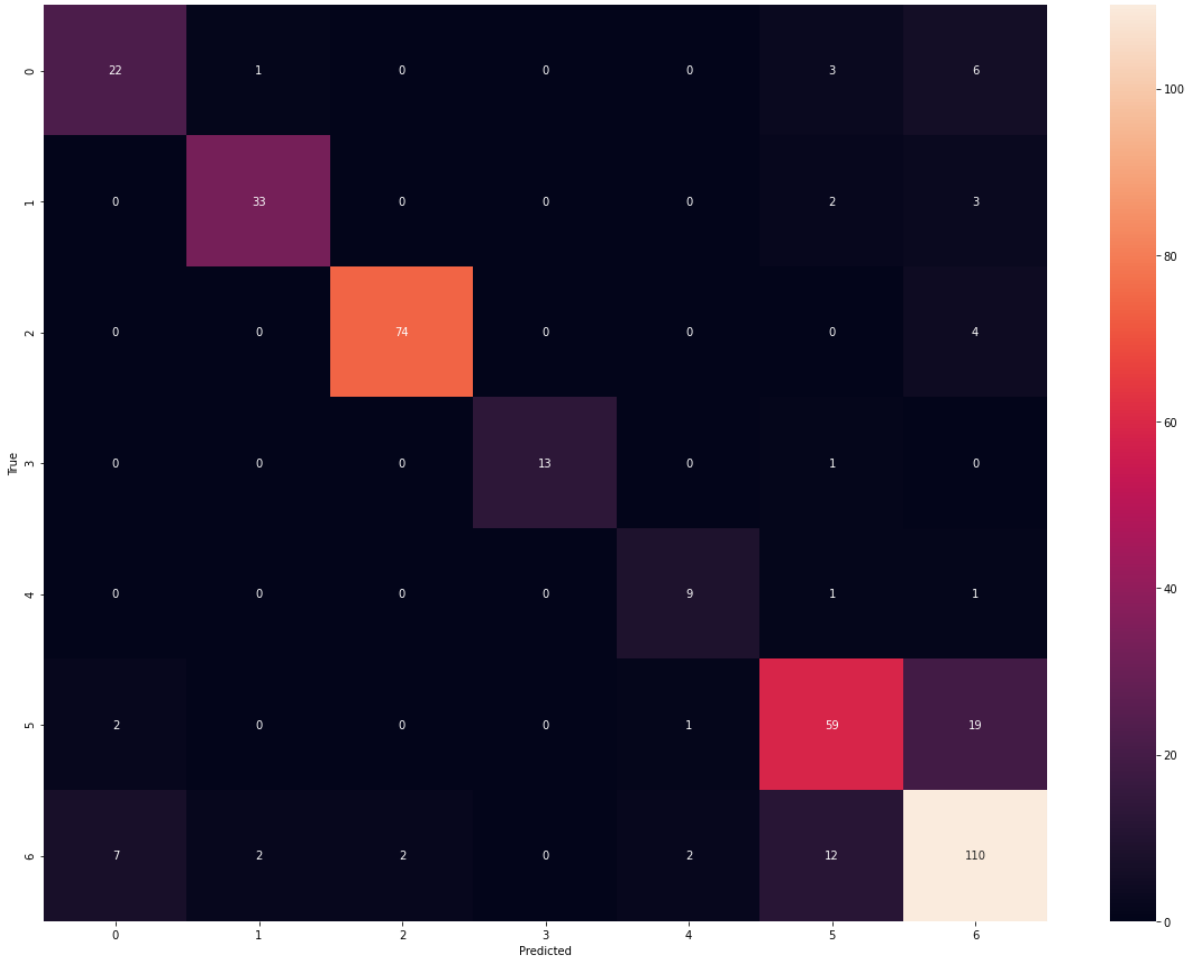
Εικόνα 24. Συγκεντρωτικά αποτελέσματα μοντέλων

Πίνακας 13. Συγκεντρικός πίνακας χρόνων εκπαίδευσης και βέλτιστων παραμέτρων

	Polynomial Feature Engineering (n=2)	PCA	Feature Importances	Simple Tuning
	Best Parameters and tuning time			
Random Forest	Done 2400 out of 2400 | elapsed: 23.7min finished `{'bootstrap': False, 'criterion': 'entropy', 'max_depth': 100, 'max_features': 'auto', 'min_samples_leaf': 1, 'min_samples_split': 10, 'n_estimators': 500}	n_estimators=300, random_state=42)	Done 2400 out of 2400 | elapsed: 23.3min finished {'bootstrap': False, 'criterion': 'entropy', 'max_depth': 100, 'max_features': 'auto', 'min_samples_leaf': 1, 'min_samples_split': 10, 'n_estimators': 125}	Done 2400 out of 2400 | elapsed: 28.2min finished {'bootstrap': False, 'criterion': 'entropy', 'max_depth': 100, 'max_features': 'sqrt', 'min_samples_leaf': 1, 'min_samples_split': 10, 'n_estimators': 375}
SVC	Done 250 out of 250 | elapsed: 26.0s finished {'C': 20, 'gamma': 0.1, 'kernel': 'rbf'}	(C=26, gamma=0.25, random_state=42)	Done 250 out of 250 | elapsed: 17.8s finished {'C': 30, 'gamma': 0.25, 'kernel': 'rbf'}	Done 250 out of 250 | elapsed: 21.1s finished {'C': 23, 'gamma': 0.15, 'kernel': 'rbf'}
LR	Done 420 out of 420 | elapsed: {'C': 10.0, 'penalty': 'l2', 'solver': 'lbfgs'}	(C=1000.0, random_state=42, solver='newton-cg')	Done 420 out of 420 | elapsed: 25.9s finished {'C': 100.0, 'penalty': 'l1', 'solver': 'saga'}	Done 420 out of 420 | elapsed: 35.1s finished {'C': 100.0, 'penalty': 'l2', 'solver': 'newton-cg'}
k-NN	Done 640 out of 640 | elapsed: 23.0s finished {'algorithm': 'auto', 'leaf_size': 10, 'n_neighbors': 5, 'weights': 'distance'}	(leaf_size=10, n_neighbors=4, weights='distance')	Done 640 out of 640 | elapsed: 14.3s finished{'algorithm': 'auto', 'leaf_size': 10, 'n_neighbors': 3, 'weights': 'distance'}	Done 640 out of 640 | elapsed: 17.0s finished {'algorithm': 'auto', 'leaf_size': 10, 'n_neighbors': 5, 'weights': 'distance'}

Train Accuracy	: 0.999
Test Accuracy	: 0.823

Classification Report:				
	precision	recall	f1-score	support
1	0.71	0.69	0.70	32
2	0.92	0.87	0.89	38
3	0.97	0.95	0.96	78
4	1.00	0.93	0.96	14
5	0.75	0.82	0.78	11
6	0.76	0.73	0.74	81
7	0.77	0.81	0.79	135
accuracy			0.82	389
macro avg	0.84	0.83	0.83	389
weighted avg	0.82	0.82	0.82	389



Εικόνα 25. Random Forest - Polynomial Feature Engineering and Tuning – Best Results

Κεφάλαιο 5: ΣΥΝΘΕΤΙΚΑ ΔΕΔΟΜΕΝΑ

5.1 Εισαγωγή

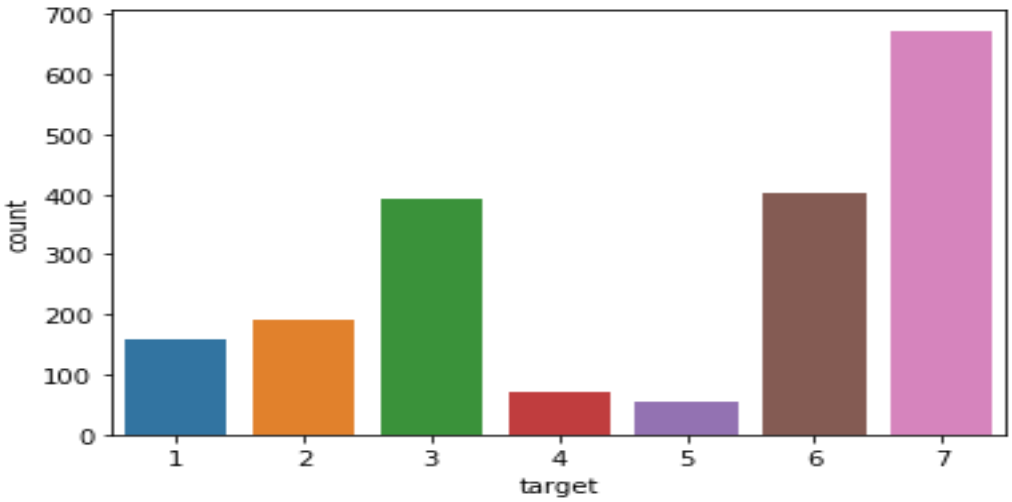
Το πρόβλημα της ανομοιογένειας των δεδομένων, σχετικά με τις κλάσεις, δημιουργεί δυσκολίες στα μοντέλα μηχανικής μάθησης να εκπαιδευτούν και να παράξουν αποτελέσματα με μεγάλη συνολική ακρίβεια και ορθότητα. Ο αριθμός των δειγμάτων ανά κλάση είναι τελείως διαφορετικός με αποτέλεσμα τα μοντέλα να εκπαιδεύονται σε τελείως διαφορετικούς αριθμούς δειγμάτων ανά κάθε κλάση (imbalanced learning). Το πρόβλημα γίνεται ακόμα πιο δύσκολο καθώς η κλάση με τις περισσότερες παρατηρήσεις (majority class) είναι αυτή που περιέχει τα πιο προβληματικά δεδομένα διότι απλώνεται στο εύρος τιμών των άλλων. Επιπροσθέτως, ο αριθμός των παρατηρήσεων του συνόλου δεδομένων μπορεί να θεωρηθεί από σχετικά μικρός εως μεσαίος για ένα πρόβλημα με 7 κλάσεις κάτι που αποτελεί ένα επιπρόσθετο πρόβλημα που καλούνται τα μοντέλα να λύσουν. Το πρόβλημα των μη σταθμισμένων κλάσεων στην περίπτωση εφαρμογών του πραγματικού κόσμου αντιπροσωπεύει ένα σημαντικό ζήτημα με πολυποίκιλες επιπτώσεις που απαιτούν προσεκτικές διερευνήσεις.

5.2 Δημιουργία συνθετικών δεδομένων

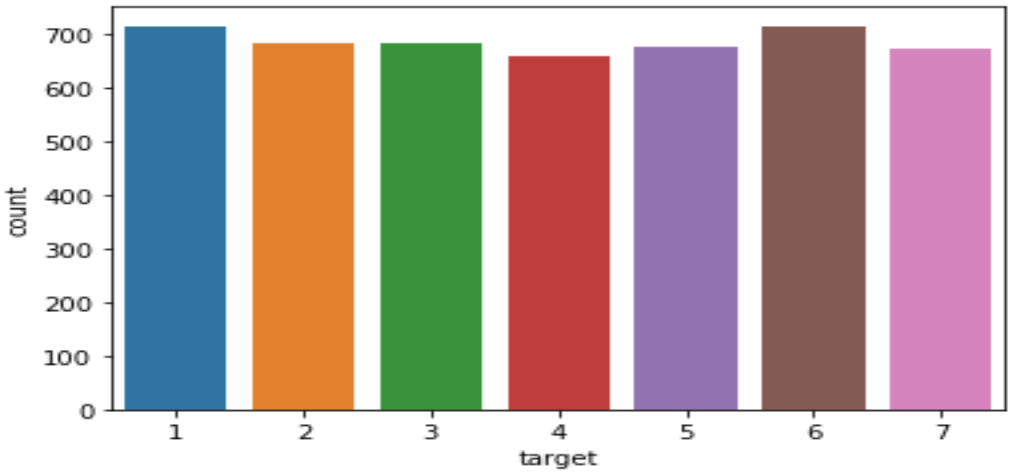
Στο πλαίσιο ανασκόπησης της βιβλιογραφίας πολλοί ερευνητές που ασχολήθηκαν με το υπάρχον σύνολο δεδομένων (Faulty Steel Plastes) επέλεξαν να χρησιμοποιήσουν τεχνικές για την δημιουργία συνθετικών δεδομένων και την εξισορρόπηση των δεδομένων ανά κλάση. Σε δημοσίευση που έγινε το 2020 [30] χρησιμοποιήθηκε η τεχνική SMOTE [72] η οποία γενικά είναι κατασκευασμένη για προβλήματα δυαδικής ταξινόμησης (binary classification) αλλά παραμετροποιήθηκε ώστε να εφαρμοστεί στο συγκεκριμένο πρόβλημα (multiclass classification). Εδώ θα πρέπει να αναφερθεί ότι η τεχνική εφαρμόστηκε στο σύνολο των δεδομένων και όχι μόνο στο υποσύνολο εκπαίδευσης κάτι που δεν θεωρείται καλή πρακτική καθώς οι προβλέψεις και η παραγωγή αποδόσεων των μοντέλων δεν είναι καλό να στηρίζεται πάνω σε συνθετικά δεδομένα. Σε αυτή την κατεύθυνση για λόγους σύγκρισης αποτελεσμάτων και μόνο, κινηθήκαμε και στην παρούσα εργασία. Επιλέχθηκε να γίνει χρήση της τεχνικής Προσαρμοστικής Συνθετικής Δειγματοληψίας (Adaptive Synthetic Sampling - Adasyn) για την εξισορρόπηση των δειγμάτων των κλάσεων. Η συγκεκριμένη τεχνική, πρώτα υπολογίζει για όλες τις μειοψηφούσες κλάσεις τον λόγο τους προς την πλειοψηφούσα κλάση, στη συνέχεια βρίσκει τον αριθμό των συνθετικών δειγμάτων που πρέπει να παραχθούν για την συγκεκριμένη κλάση μειοψηφίας και βρίσκει για κάθε δείγμα της κλάσης τους k κοντινότερους γείτονες (k-N-N) καθώς και τα δεδομένα που πρέπει να παραχθούν. Αυτή η διαδικασία επαναλαμβάνεται για όλες τις κλάσεις. Περισσότερες πληροφορίες σχετικά με την Adasyn μπορούν να βρεθούν στην αναφορά [73]. Χρησιμοποιήθηκε και εδώ η αντίστοιχη βιβλιοθήκη του imbalanced - learn [74].

5.3 Προετοιμασία δεδομένων

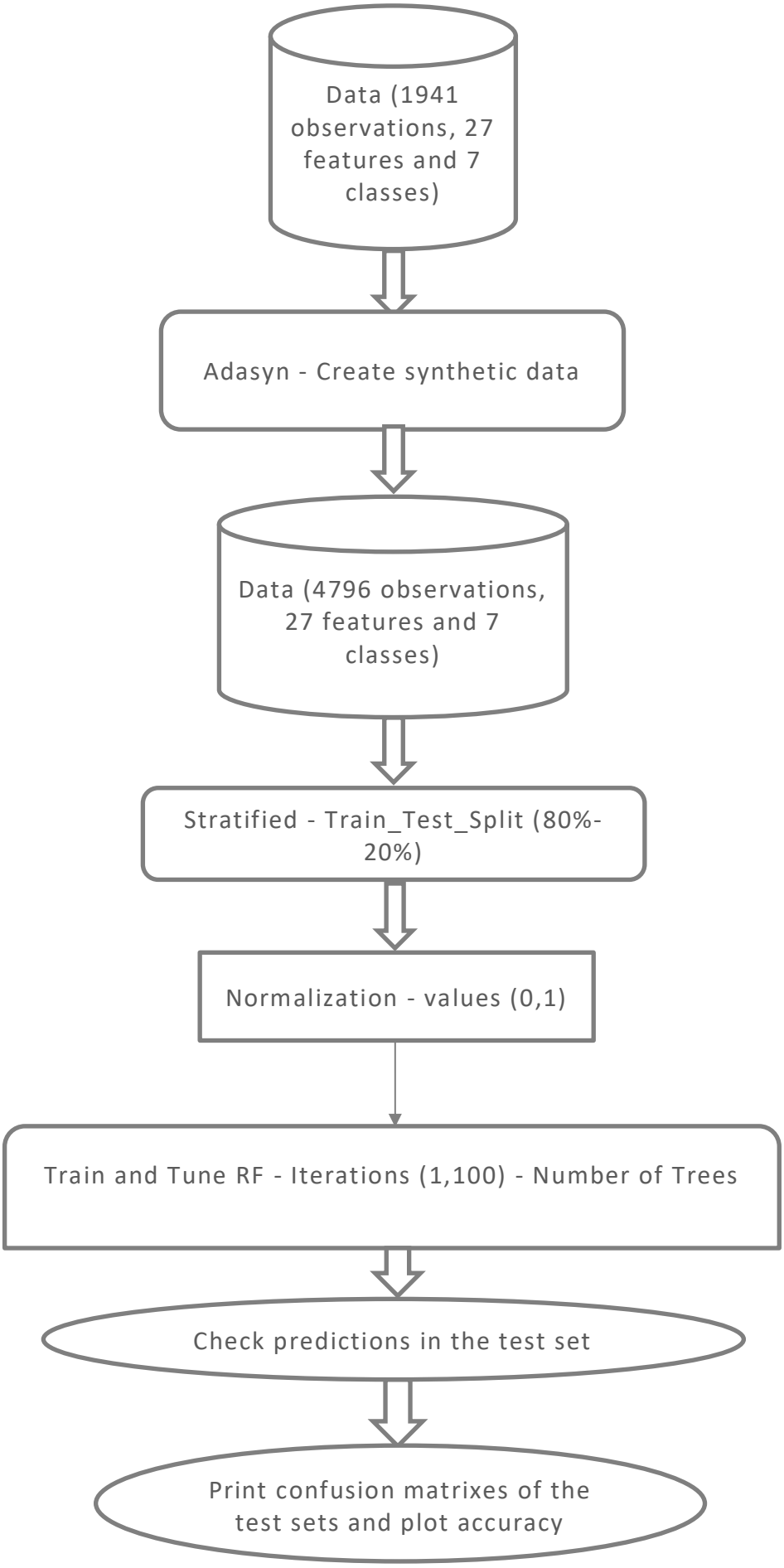
Εφαρμόστηκε η μέθοδος Adasyn σε όλο το αρχικό σύνολο δεδομένων με σκοπό την δημιουργία συνθετικών δεδομένων ώστε οι παρατηρήσεις για όλες τις κλάσεις να είναι σχεδόν ίδιες. Πιο συγκεκριμένα, το αρχικό σύνολο δεδομένων αποτελείται από 1941 παρατηρήσεις. Με την μέθοδο Adasyn που εφαρμόστηκε το νέο σύνολο δεδομένων αποτελείται από 4796 παρατηρήσεις. Στην συνέχεια το σύνολο δεδομένων χωρίστηκε σε υποσύνολα εκπαίδευσης και ελέγχου σε αναλογία 80% - 20%. Έγινε κανονικοποίηση των δεδομένων σε εύρος τιμών (0,1) και στην συνέχεια εκπαιδεύτηκε ένα Random Forest στο οποίο πραγματοποιήθηκε απλή ρύθμιση μόνο ως προς τις τιμές των δέντρων που χρησιμοποιούνται με επαναλήψεις από (1,100). Παρακάτω, στην Εικόνα 26 και στην Εικόνα 27 παρουσιάζονται τα δεδομένα και οι κλάσεις τους πριν και μετά την εφαρμογή της Adasyn.



Εικόνα 26. Κλάσεις αρχικού dataset - count observations per class



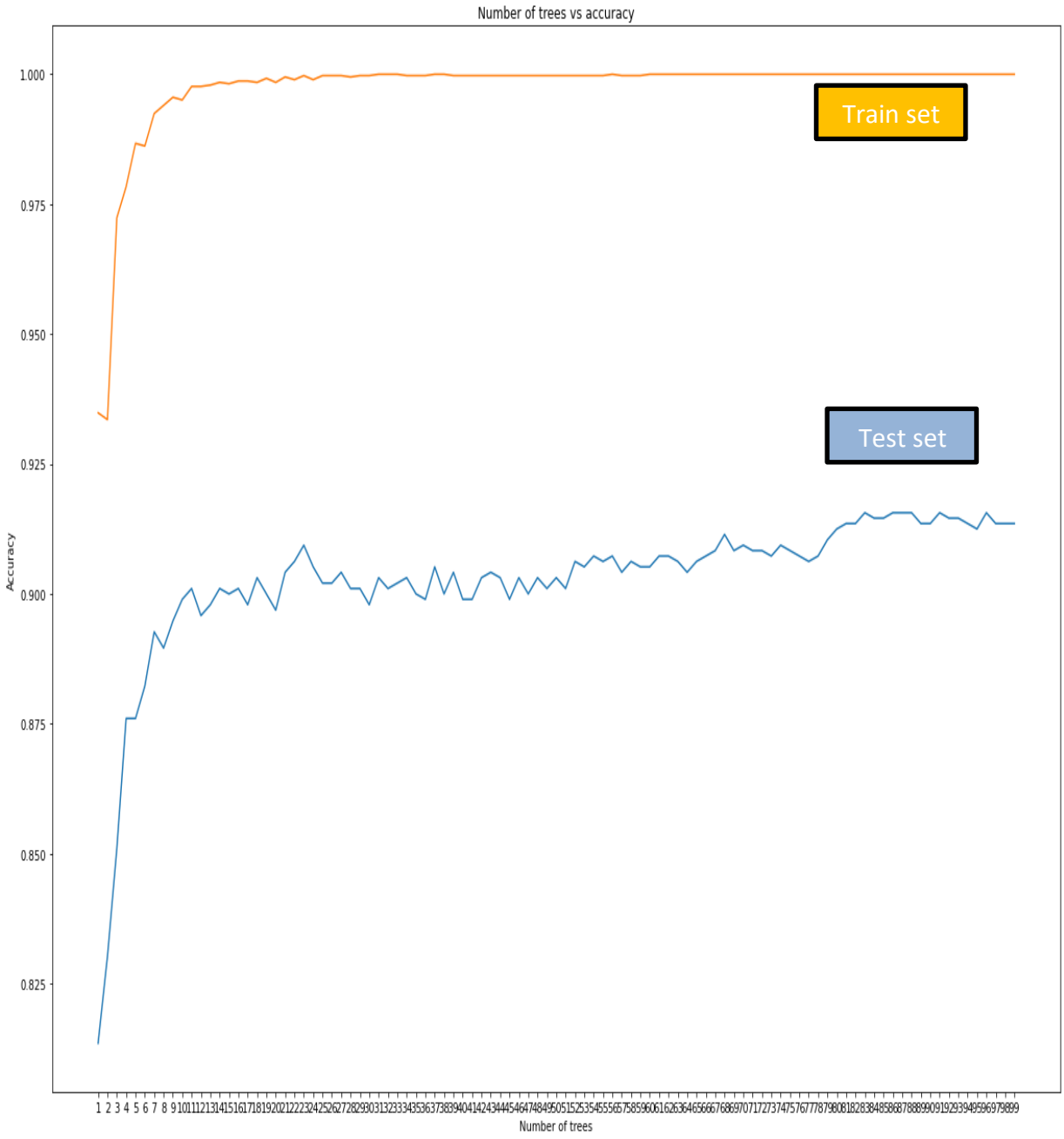
Εικόνα 27. Κλάσεις dataset με συνθετικά δεδομένα - count observations per class



Εικόνα 28. Λογικό διάγραμμα κώδικα ADASYN

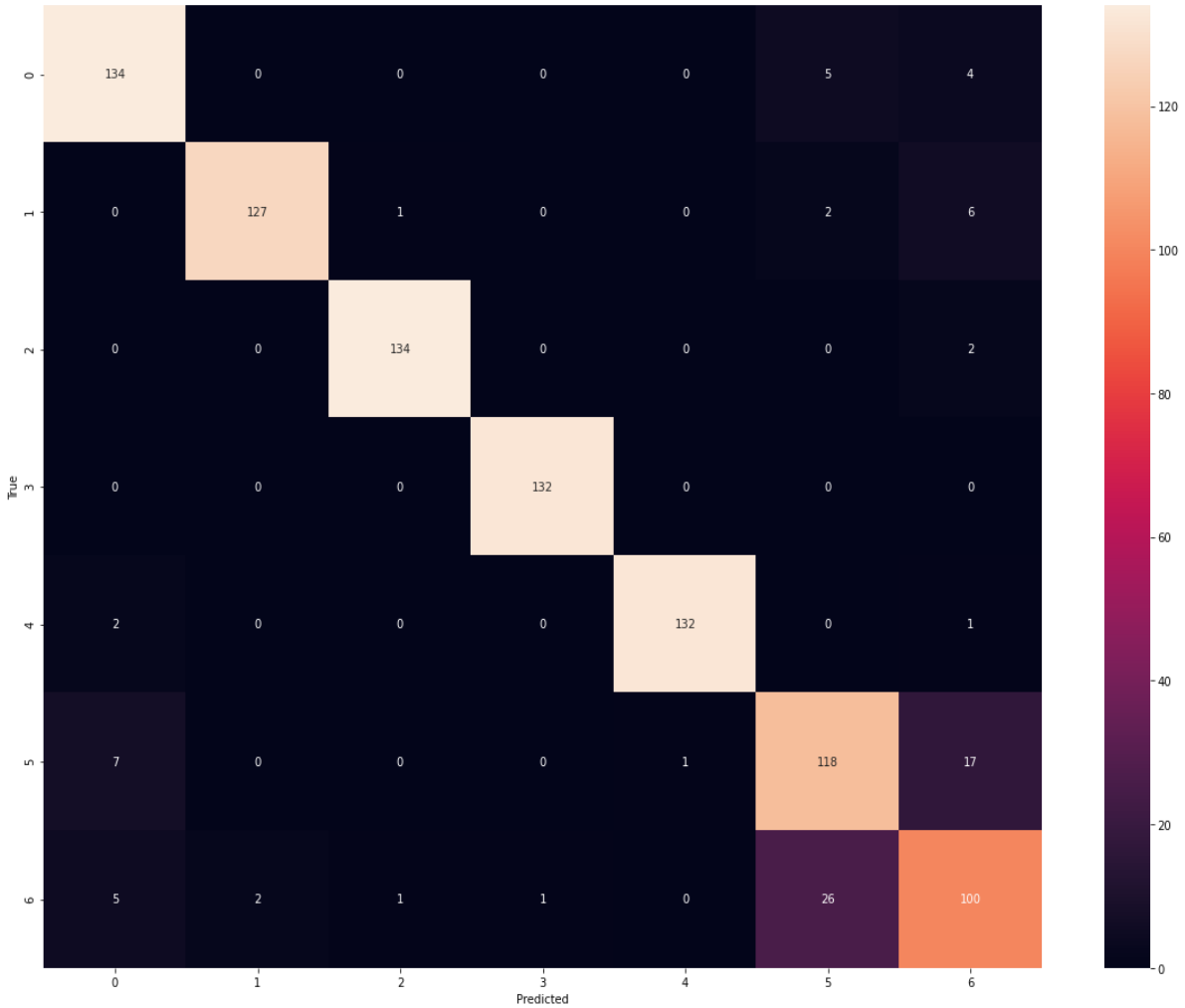
5.4 Αποτελέσματα Μοντέλου

Παρακάτω, στην Εικόνα 29 φαίνονται τα αποτελέσματα για το accuracy στο σετ εκπαίδευσης και στο σετ ελέγχου. Παρατηρούμε ότι για 80 δέντρα απόφασης το accuracy ξεπερνάει το 91% στο τεστ σετ. Το συνολικό αποτέλεσμα είναι πολύ ικανοποιητικό. Σε αυτό το σημείο πρέπει να αναφερθεί ότι η καλή πρακτική είναι να εφαρμόζεται η τεχνική παραγωγής συνθετικών δεδομένων μόνο στο σετ εκπαίδευσης. Ο λόγος είναι ότι υπάρχει διάχυση πληροφορίας (information leakage) καθώς τα νέα δεδομένα που παράγονται βγαίνουν και σύμφωνα με τιμές του test set. Βέβαια η παραγωγή συνθετικών δεδομένων δεν είναι ένα απλό oversampling που απλά αντιγράφονται τιμές των παρατηρήσεων κάτι που θα περιείχε περισσότερη διάχυση πληροφορίας.



Εικόνα 29. Αποτελέσματα - train set - test set – Adasyn

Classification Report:				
	precision	recall	f1-score	support
1	0.91	0.94	0.92	143
2	0.98	0.93	0.96	136
3	0.99	0.99	0.99	136
4	0.99	1.00	1.00	132
5	0.99	0.98	0.99	135
6	0.78	0.83	0.80	143
7	0.77	0.74	0.75	135
accuracy			0.91	960
macro avg	0.92	0.91	0.91	960
weighted avg	0.91	0.91	0.91	960



Εικόνα 30. Confusion Matrix - Adasyn

Κεφάλαιο 6: ΠΡΟΣΑΡΜΟΣΜΕΝΟΣ ΤΑΞΙΝΟΜΗΤΗΣ

6.1 Εισαγωγή

Στο συγκεκριμένο κεφάλαιο αναλύεται η διαδικασία που ακολουθήθηκε για την δημιουργία ενός προσαρμοσμένου ταξινομητή για την βελτίωση της μετρικής της συνολικής ακρίβειας και ορθότητας. Όπως φαίνεται από την ανάλυση των δεδομένων που έχει γίνει, το συγκεκριμένο σύνολο δεδομένων έχει πρόβλημα λόγω των μη σταθμισμένων κλάσεων. Επίσης, η τελευταία κλάση Other Faults που έχει τις περισσότερες παρατηρήσεις έχει πολύ μεγάλη διασπορά σε όλο το φάσμα των τιμών των υπόλοιπων κλάσεων. Αυτό μας δείχνει ότι η συγκεκριμένη κλάση είναι δύσκολο να ταξινομηθεί καθώς τα μοντέλα δεν θα μπορούν να την διαχωρίσουν. Επιπροσθέτως, εδώ μπορούμε να καταλάβουμε ότι η συγκεκριμένη κλάση μπορεί να περιέχει δεδομένα – τύπους ελαττωμάτων από τις άλλες κλάσεις (Common Faults) και για αυτό οι τιμές της να πέφτουν πάνω στις τιμές των άλλων ελαττωμάτων. Επιλέχθηκε το πρόβλημα να σπάσει σε 3 στάδια ταξινόμησης (classication stages). Στη συνέχεια, παρουσιάζεται αναλυτικά η μέθοδος που αναπτύχθηκε καθώς σκοπός ήταν να δημιουργηθεί ένας πλήρως αυτοματοποιημένος προσαρμοσμένος ταξινομητής που σε αυτά τα 3 στάδια ταξινόμησης θα κατηγοριοποιεί με υψηλή ακρίβεια και τις 7 κλάσεις του προβλήματος.

6.2 Διαδικασία επίλυσης σε τρία επίπεδα ταξινόμησης

Η διαδικασία που έγινε ήταν το πρόβλημα να επιμεριστεί σε μικρότερες ταξινομήσεις. Παρόμοια λογική ακολουθείται όταν ένα πρόβλημα πολλαπλών κλάσεων χωρίζεται σε πολλά δυαδικά προβλήματα (One vs All). Πιο συγκεκριμένα, το πρώτο στάδιο ταξινόμησης (First Stage Classification) ήταν να χωριστούν τα δεδομένα στις δύο πιο γενικές κλάσεις του συνόλου δεδομένων, δηλαδή τις Common Faults vs Other Faults. Η κλάση των κοινών τύπων ελαττωμάτων (Common Faults) περιέχει μέσα όλες τις άλλες 6 κλάσεις. Οπότε κατασκευάστηκε ένα νέο αρχείο csv, στο οποίο όλα τα δεδομένα των 6 πρώτων κλάσεων ενοποιήθηκαν και πήραν κοινή τιμή ως προς τον στόχο (target). Στη συνέχεια, πραγματοποιείται δυαδική ταξινόμηση ανάμεσα στους τύπους ελαττωμάτων Common Faults και Other Faults. Εδώ θα θέλαμε να αναφερθεί ότι πολλοί που ασχολήθηκαν με το συγκεκριμένο πρόβλημα δεν το λύνουν σαν ένα πρόβλημα πολλαπλών κλάσεων αλλά το αντιμετωπίζουν σαν πρόβλημα δυαδικής ταξινόμησης [11] με την ίδια τεχνική που περιγράφεται για το πρώτο στάδιο ταξινόμησης μόνο.

Στο δεύτερο στάδιο ταξινόμησης έχουμε ήδη ταξινομήσει την κλάση Other Faults του προβλήματος. Αυτό που συμβαίνει εδώ είναι η δημιουργία τριών Φανταστικών κλάσεων (Imaginary Classes) που η καθεμία ουσιαστικά είναι η ένωση δύο τύπων ελαττωμάτων από τις 6 απομένουσες κλάσεις που πρέπει να ταξινομηθούν. Η πρώτη κλάση A περιέχει όλες τις παρατηρήσεις των κλάσεων Pastry και Dirtiness, η δεύτερη φανταστική κλάση B περιέχει όλες τις παρατηρήσεις των κλάσεων Z_Scratch και Bumps και η τρίτη Φανταστική κλάση περιέχει όλες τις παρατηρήσεις των κλάσεων K_Scratch και Stains. Αυτό ουσιαστικά πραγματοποιείται

διότι θέλουμε να δημιουργήσουμε 3 δυαδικά προβλήματα για τις 6 κλάσεις που μένουν ως προς ταξινόμηση. Μετά το πρώτο στάδιο ταξινόμησης θα μπορούσαμε απλά να σπάσουμε το πρόβλημα σε 3 δυαδικές ταξινομήσεις όπου οι κλάσεις διαχωρίζονται σύμφωνα με την ανάλυση των δεδομένων που έχει γίνει αυτό όμως θα ήταν λάθος καθώς σε περίπτωση που θα λαμβάναμε δεδομένα για να ταξινομήσουμε τις 7 κλάσεις του προβλήματος και να εφαρμόσουμε τον προσαρμοσμένο ταξινομητή που κατασκευάστηκε δεν θα μπορούσαμε να γνωρίζουμε στο δεύτερο στάδιο ποιο από τα 3 δυαδικά classification να κάνουμε. Οπότε η δημιουργία τριών Φανταστικών κλάσεων ήταν κάτι απαραίτητο που έπρεπε να πραγματοποιηθεί.
Εδώ είναι σημαντικό να αναφερθεί ότι οι Φανταστικές Κλάσεις που έγιναν με την ένωση των τύπων ελαττωμάτων δεν είναι καθόλου τυχαίες. Το πρώτο πράγμα που κοιτάξαμε ήταν οι κλάσεις που θα ενωθούν για να φτιάξουν τις 3 νέες κλάσεις να είναι αρκετά διαχωρίσιμες ανά δύο μεταξύ τους ώστε στο τρίτο στάδιο να μπορούν να διαχωριστούν. Αυτό βέβαια είναι το ένα μόνο κομμάτι που μας απασχόλησε. Το άλλο είναι ότι οι 3 Φανταστικές κλάσεις A, B, C θα πρέπει να είναι και αυτές διαχωρίσιμες, δηλαδή οι ενώσεις ανά δύο των υπόλοιπων 6 κλάσεων θα πρέπει να διαχωρίζονται και αυτές για να πετύχουμε υψηλές αποδόσεις στο δεύτερο στάδιο της ταξινόμησης.

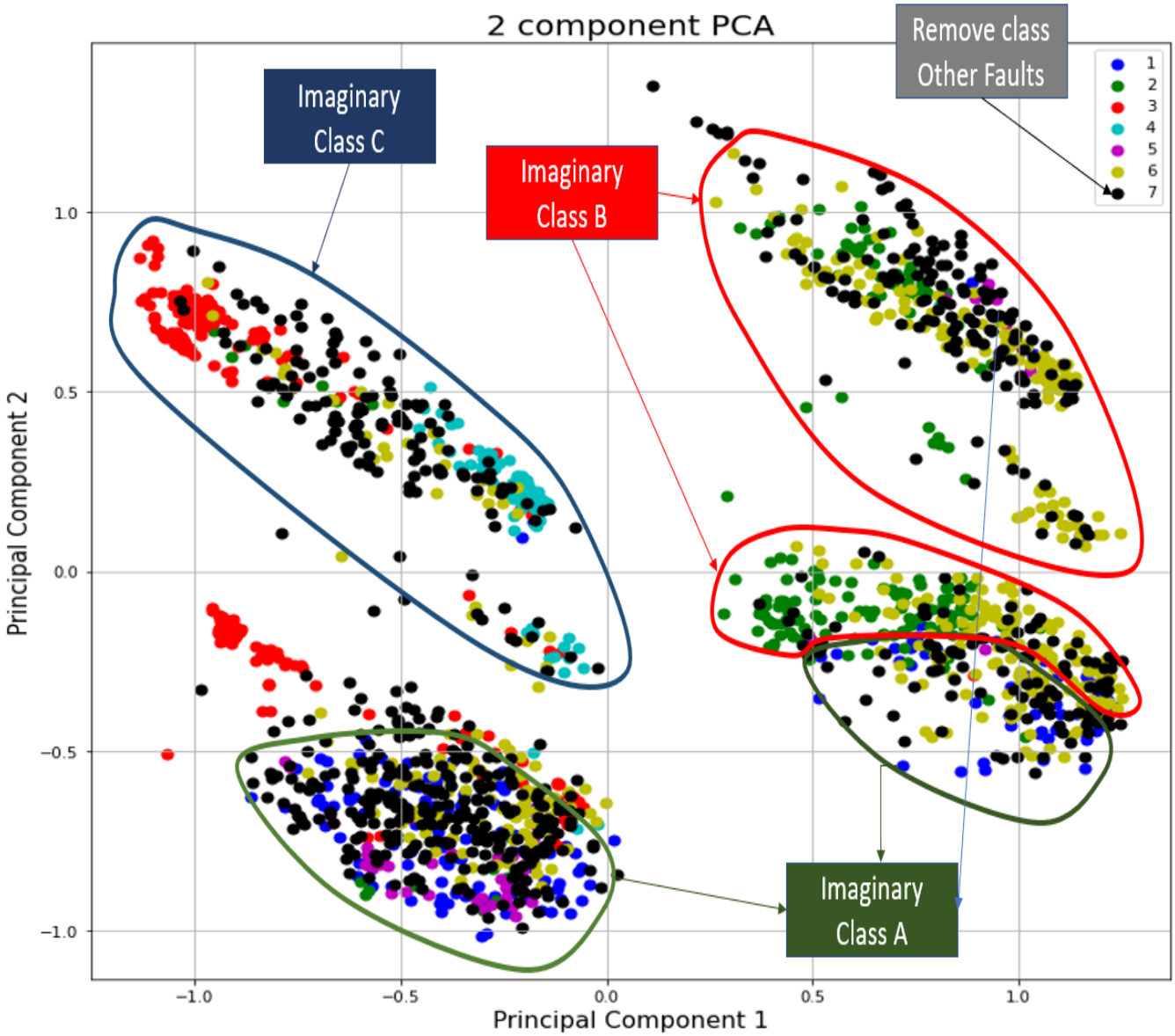
Σύμφωνα με το δυωνυμικό θεώρημα και απλή συνδυαστική έχουμε το εξής:

$$\binom{n}{k} = \frac{n!}{k!(n-k)!} = \frac{6!}{2!4!} = 15 \quad (11)$$

Από τους 15 συνδυασμούς που υπήρξαν έγιναν πειράματα με δοκιμή και λάθος ως προς τις αποδόσεις της ταξινόμησης και στο δεύτερο και στο τρίτο στάδιο. Να αναφέρουμε ότι οι επιλογές ως προς τις ενώσεις των κλάσεων που έγιναν στηρίζονται και στην ανάλυση των δεδομένων καθώς και στις οπτικοποιήσεις που πραγματοποιήθηκαν (PCA, t – SNE) και δεν χρειάστηκε να χρησιμοποιηθούν και οι 15 συνδυασμοί. Το ιδανικό όπως καταλαβαίνει κανείς είναι να γίνουν ενώσεις κλάσεων που φαίνονται να είναι κοντά μεταξύ τους αλλά διαχωρίσιμες ανά δύο συμπεριλαμβανομένου βέβαια ότι και οι 3 ενώσεις είναι διαχωρίσιμες μεταξύ τους ώστε να έχουμε υψηλά αποτελέσματα και στο δεύτερο και στο τρίτο στάδιο ταξινόμησης. Το κομμάτι της ανάλυσης που χρησιμοποιήθηκε μπορεί να φανεί στο διάγραμμα της Εικόνας 31.

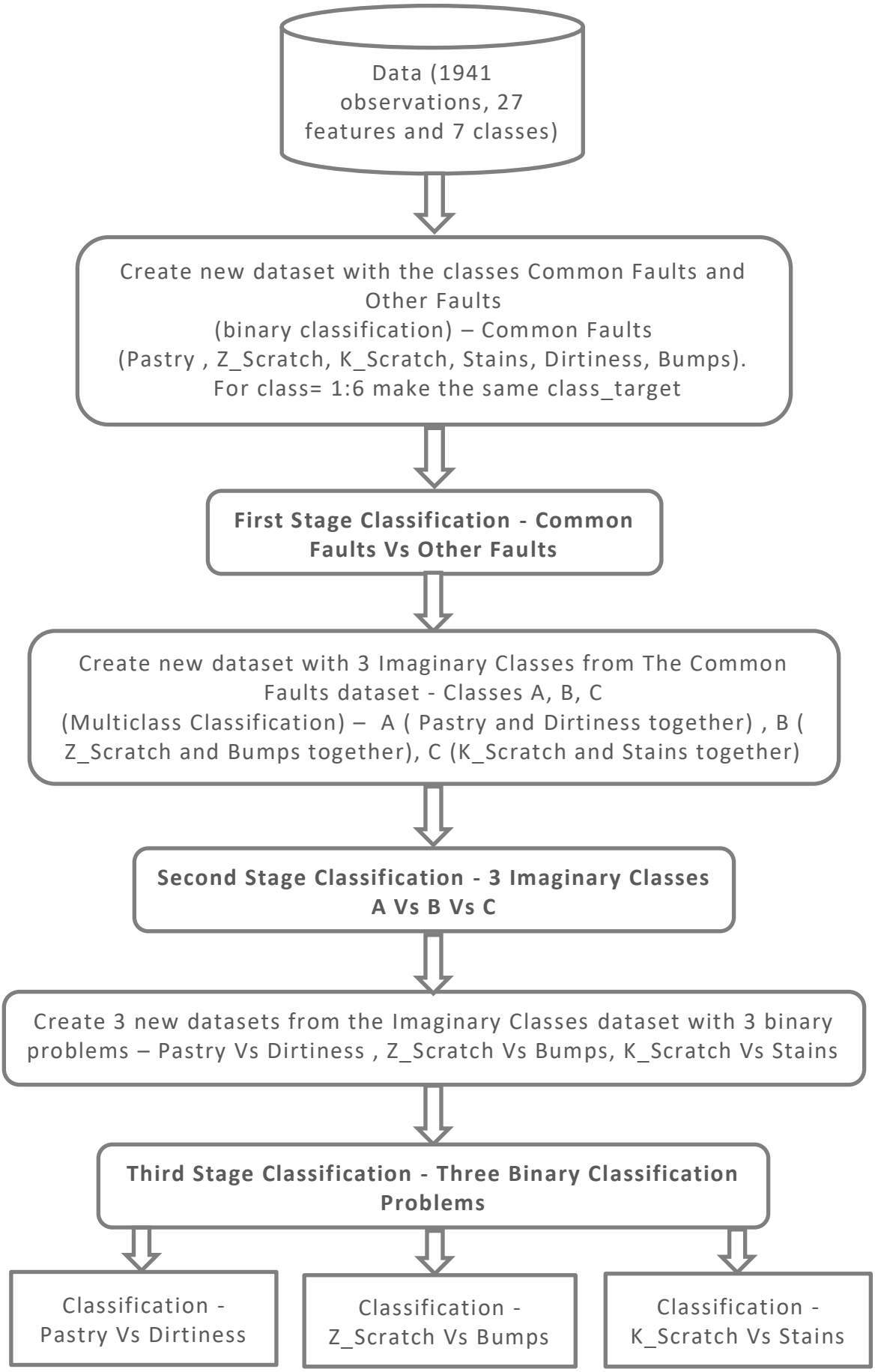
Πιο συγκεκριμένα, στην Εικόνα 31 φαίνεται ο τρόπος που γίνεται η επιλογή των ενώσεων των κλάσεων. Παρατηρήθηκε ότι $6^{η}$ κλάση Bumps με την $2^{η}$ κλάση Z_Scratch είναι αρκετά κοντά μεταξύ τους αλλά φαίνονται ότι διαχωρίζονται. Επίσης, αν ενωθούν οι δύο αυτές κλάσεις φαίνεται σε μεγάλο βαθμό να διαχωρίζονται από τις υπόλοιπες αν εξαιρεθεί η τελευταία κλάση Other Faults. Παρομοίως τα ίδια μπορούμε να υποθέσουμε και για την $1^{η}$ κλάση Pastry αν ενωθεί με την $5^{η}$ κλάση Dirtiness καθώς και για την $3^{η}$ κλάση K_Scratch αν ενωθεί με την $4^{η}$ κλάση Stains.

Επιπροσθέτως, παρατηρείται κάποια διασπορά της $6^{ης}$ κλάσης Bumps στην Φανταστική κλάση A, κάτι που μας δείχνει ότι θα πρέπει να περιμένουμε χαμηλότερα αποτελέσματα για τον διαχωρισμό της από τις άλλες Φανταστικές κλάσεις. Για τις άλλες δύο Φανταστικές κλάσεις παρατηρούμε πολύ μικρή διασπορά παρατηρήσεων των άλλων κλάσεων, κάτι που θα έχει θετικό πρόσημο για τα αποτελέσματα των μοντέλων.



Εικόνα 31. Επιλογή ένωσης κλάσεων μέσω οπτικοποίησης με PCA

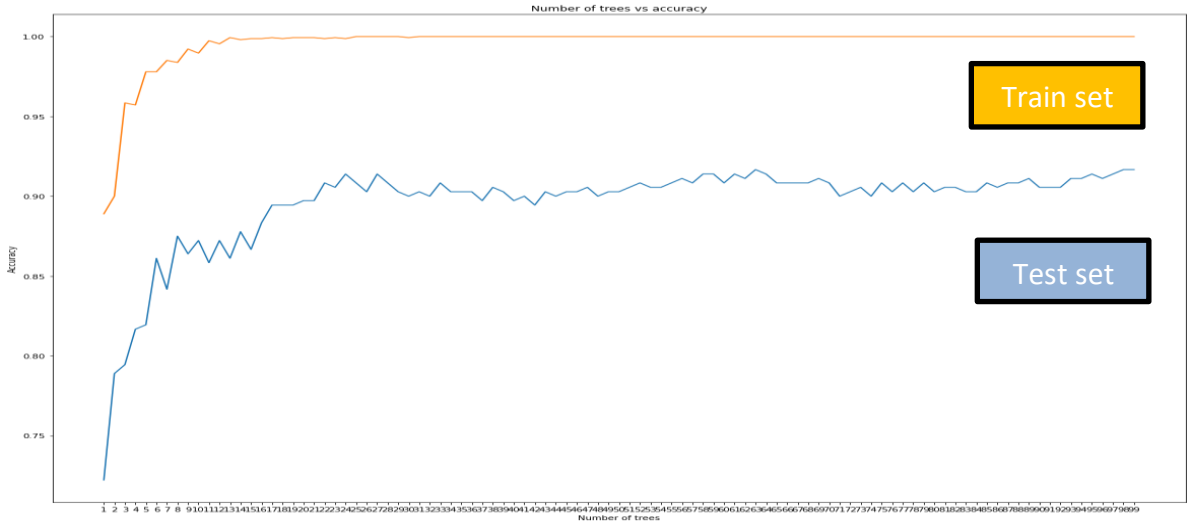
Όταν μπαίνουμε στο τρίτο στάδιο ταξινόμησης έχουμε ταξινομήσει και τα δεδομένα των κοινών τύπων ελαττωμάτων ως προς τις 3 Φανταστικές Κλάσεις. Για κάθε μία από αυτές απλά κάνουμε μία δυαδική ταξινόμηση για να σπάσουμε την ένωση τους. Έτσι στο τέλος έχουμε ταξινομήσει και τις 7 κλάσεις του συνόλου δεδομένων. Τα αποτελέσματα ήταν ιδιαίτερα ικανοποιητικά και θα παρουσιαστούν στο τέλος του Κεφαλαίου. Στην συνέχεια υπάρχει το λογικό διάγραμμα του κώδικα στην Εικόνα 32 για τον προσαρμοσμένο ταξινομητή, καθώς και γίνεται αναλυτική παρουσίαση της προ-επεξεργασίας των δεδομένων και των μοντέλων που χρησιμοποιήθηκαν στα 3 Στάδια Ταξινόμησης.



Εικόνα 32. Custom Classifier - 3 Stages of Classification

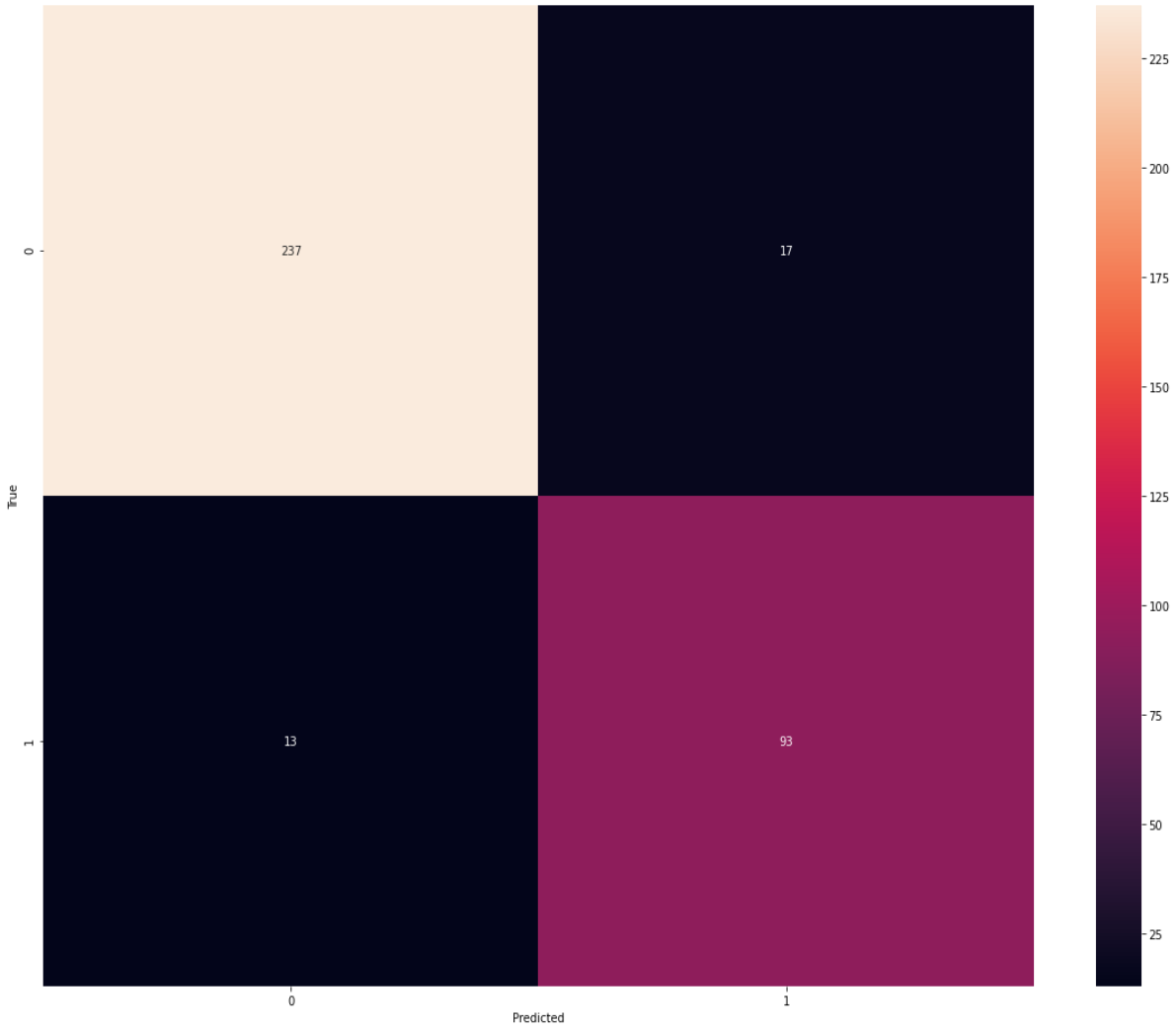
6.2.1 Πρώτο Στάδιο Ταξινόμησης – Common Faults vs Other Faults

Το σύνολο δεδομένων χωρίστηκε σε Common Faults και Other Faults όπου η νέα κλάση των κοινών τύπων ελαττωμάτων περιείχε όλες τις 6 πρώτες κλάσεις. Συγκεκριμένα, από τις 1941 παρατηρήσεις, η κλάση Common Faults έχει 1268 δείγματα και η κλάση Other Faults έχει 673 δείγματα. Οπότε η νέα κλάση που δημιουργήθηκε αποτελεί πλέον την πλειοψηφούσα κλάση του συνόλου δεδομένων. Εδώ πριν τα δεδομένα χωριστούν με την train_test_split πραγματοποιήθηκε αφαίρεση ακραίων τιμών (outliers) με την DBSCAN (Density-Based Spatial Clustering of Applications with Noise). Παρόμοια μεθοδολογία έχει γίνει για το παρόν σύνολο δεδομένων και στην αναφορά [24]. Η τεχνική DBSCAN πραγματοποιεί ανίχνευση ακραίων τιμών καθώς προσπαθεί να ομαδοποιήσει (clustering) τα δεδομένα σε λειτουργικά και σε προβληματικά δημιουργώντας ομάδες – γειτονιές για τις παρατηρήσεις βάσει της ακτίνας και των αριθμών των σημείων που δίνονται ως παράμετροι. Τα δεδομένα που δεν μπορούν να ενταχθούν στην ομαδοποίηση ορίζονται ως προβληματικά (outliers). Στην παρούσα εργασία πραγματοποιήθηκαν δύο προσεγγίσεις , η «ελαφριά» προσέγγιση (light approach) και η «βαριά» προσέγγιση (heavy approach) όπου αλλάζοντας τις παραμέτρους της DBSCAN στην πρώτη αφαιρούσαμε λιγότερα outliers και στην δεύτερη περισσότερα. Για την επίτευξη υψηλότερων αποδόσεων αφαιρέθηκαν από το σετ δεδομένων 56 παρατηρήσεις και ακολουθήθηκε η «βαριά» προσέγγιση. Εδώ πρέπει να αναφερθεί ότι τα 56 δείγματα είναι λιγότερα από το 5% του συνόλου δεδομένων όπως ορίζει η καλή πρακτική για τηνα αφαίρεση ακραίων τιμών. Βέβαια η αφαίρεση των δεδομένων έγινε σε δύο στάδια. Στο πρώτο στάδιο αφαιρέθηκαν 30 παρατηρήσεις από όλο το σύνολο δεδομένων και στη συνέχεια τα δεδομένα χωρίστηκαν σε υποσύνολα εκπαίδευσης και ελέγχου με ποσοστό 80 % - 20% . Στο δεύτερο στάδιο, αφαιρέθηκαν ακόμα 26 παρατηρήσεις μόνο από το υποσύνολο ελέγχου. Τα δεδομένα κανονικοποιήθηκαν σε εύρος (0,1) και στη συνέχεια αναπτύχθηκε μοντέλο μηχανικής μάθησης Random Forest με απλή ρύθμιση μόνο ως προς την παράμετρο των αριθμών των δέντρων (1,100 – number of iterations for different trees). Παρακάτω, στην Εικόνα 33 και στην Εικόνα 34 παρουσιάζονται τα αποτελέσματα για τη συνολική ακρίβεια στα υποσύνολα εκπαίδευσης και ελέγχου, καθώς και η σχετική έκθεση ταξινόμησης με τον πίνακα σύγχυσης (0-κλάση Common Faults , 1-κλάση Other Faults).



Εικόνα 33. Αποτελέσματα – Πρώτο Στάδιο – Custom Classifier

Classification Report:				
	precision	recall	f1-score	support
0	0.95	0.93	0.94	254
1	0.85	0.88	0.86	106
accuracy			0.92	360
macro avg	0.90	0.91	0.90	360
weighted avg	0.92	0.92	0.92	360



Εικόνα 34 Πίνακας Σύγχυσης – Πρώτο Στάδιο Ταξινόμησης

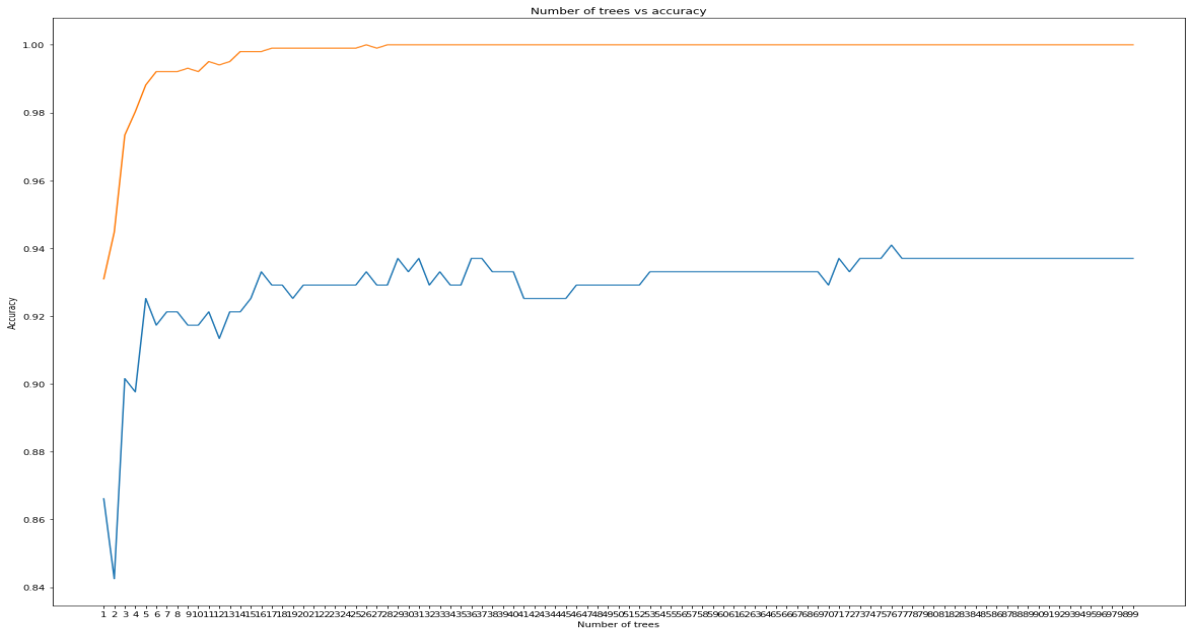
Τα αποτελέσματα είναι ιδιαίτερα ικανοποιητικά καθώς έχουμε συνολική ακρίβεια 91% και βλέπουμε ότι η κλάση Other Faults (1) έχει ακρίβεια 0.85 ενώ η κλάση Common Faults (0) έχει ακρίβεια 95% .

6.2.2 Δεύτερο Στάδιο Ταξινόμησης – Φανταστικές Κλάσεις

Στο δεύτερο στάδιο ταξινόμησης όπως αναφέρθηκε πραγματοποιείται η κατηγοριοποίηση τριών Φανταστικών Κλάσεων που δημιουργήθηκαν. Συγκεκριμένα, η κάθε μία από αυτές αποτελεί την ένωση δύο κλάσεων του προβλήματος και όπως αναφέρθηκε προηγουμένως οι επιλογές που έγιναν δεν ήταν καθόλου τυχαίες. Οπότε το σύνολο δεδομένων του Common Faults (1268 δείγματα) χωρίστηκε σε τρείς νέες Φανταστικές κλάσεις σύμφωνα με τον Πίνακα 14. Η κλάση A έχει 213 δείγματα , η κλάση B έχει 592 δείγματα και η κλάση C έχει 463. Επίσης, εδώ πρέπει να αναφερθεί ότι θα μπορούσε να γίνει και μία άλλη προσέγγιση του χωρισμού των κλάσεων ώστε να αποφευχθεί και πάλι το πρόβλημα της μη στάθμισής τους, δηλαδή να ομαδοποιηθούν οι παλιές κλάσεις με τέτοιο τρόπο ώστε οι Φανταστικές να έχουν πολύ κοντινό αριθμό παρατηρήσεων. Πραγματοποιήθηκε λοιπόν μία τέτοια προσέγγιση αλλά δεν τελεσφόρησε καθώς τα πιο σημαντικά κριτήρια της ομαδοποίησης είναι στο οι ενώσεις των κλάσεων να είναι διαχωρίσιμες μεταξύ τους αλλά και οι ίδιες οι κλάσεις όταν σπάσει η ένωση να μπορούν να διαχωριστούν. Τα δεδομένα εδώ χωρίστηκαν σε υποσύνολα εκπαίδευσης και ελέγχου με ποσοστό 80% - 20%. Στην συνέχεια κανονικοποιήθηκαν και πραγματοποιήθηκε κατασκευή πολυωνυμικών χαρακτηριστικών (n=2). Τα μοντέλο που χρησιμοποιήθηκε ήταν ένα Random Forest με απλή ρύθμιση ως προς τον αριθμό των δέντρων απόφασης του (1,100). Παρακάτω, στην Εικόνα 35 και στην Εικόνα 36 παρουσιάζονται τα αποτελέσματα.

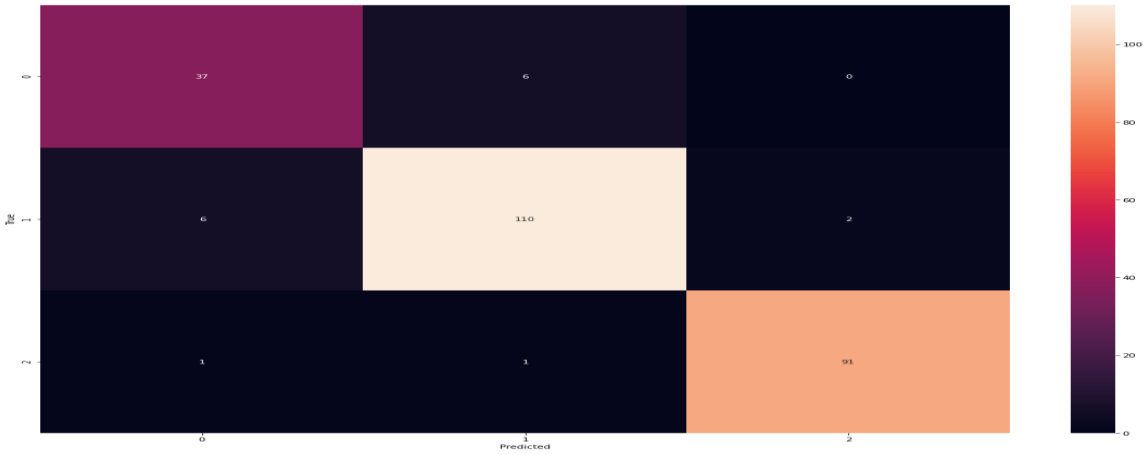
Πίνακας 14. Φανταστικές Κλάσεις - Δεύτερο Στάδιο Ταξινόμησης

New Imaginary Classes	**Dataset Classes**
Imaginary Class A	Pastry ∪ Dirtiness
Imaginary Class B	Z_Scratch ∪ Bumps
Imaginary Class C	K_Scratch ∪ Stains



Εικόνα 35. Αποτελέσματα - Train Set - Test Set- Δεύτερο Στάδιο Ταξινόμησης

Classification Report:				
	precision	recall	f1-score	support
1	0.84	0.86	0.85	43
2	0.94	0.93	0.94	118
3	0.98	0.98	0.98	93
accuracy			0.94	254
macro avg	0.92	0.92	0.92	254
weighted avg	0.94	0.94	0.94	254



Εικόνα 36. Πίνακας Σύγχυσης - Δεύτερο Στάδιο Ταξινόμησης

Σύμφωνα με τα αποτελέσματα παρατηρήθηκε υψηλή απόδοση και στο δεύτερο στάδιο της ταξινόμησης. Πιο συγκεκριμένα , πέρα από την πρώτη Φανταστική κλάση που σημείωσε το χαμηλότερο αποτέλεσμα ακρίβειας (84%), οι άλλες δύο Φανταστικές κλάσεις φαίνεται να

διαχωρίζονται πολύ καλά με ποσοστά 94% και 98% αντίστοιχα. Αυτό μας δείχνει ότι η ένωση της κλάσης Pastry μαζί με την κλάση Dirtiness παρουσίαζει μια μικρή δυσκολία σε σχέση με των διαχωρίσμό των άλλων δύο φανταστικών κλάσεων. Κάτι τέτοιο είναι φυσικό να οφείλεται και στην διασπορά της κλάσης Bumps στην κλάση Pastry που φαίνεται στην Εικόνα 31.

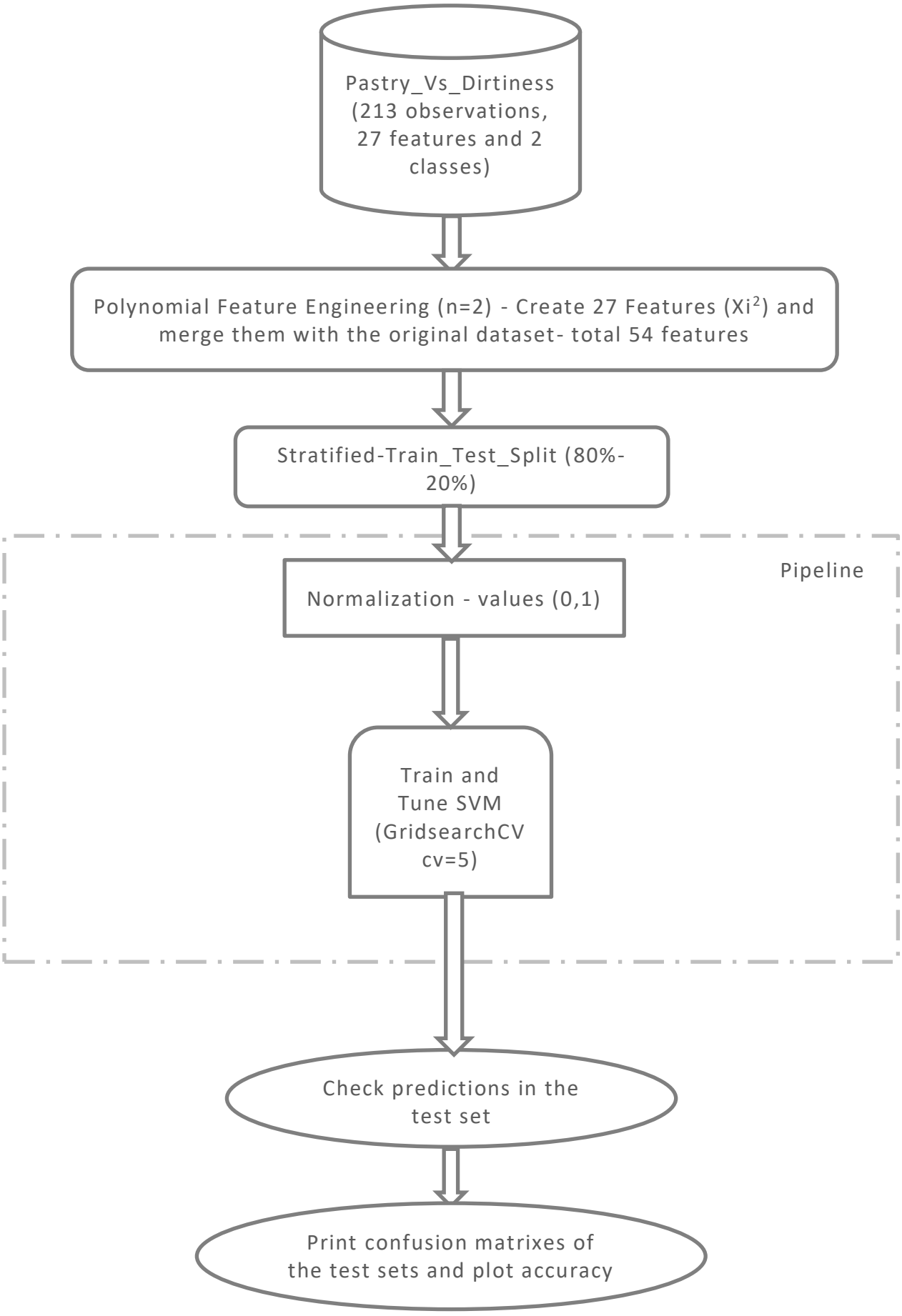
6.2.3 Τρίτο Στάδιο Ταξινόμησης – Τρεις Δυαδικές Ταξινομήσεις

Πραγματοποιήθηκαν 3 δυαδικές ταξινομήσεις αντίστοιχα πάνω στα σύνολα δεδομένων των Φανταστικών κλάσεων. Δηλαδή οι ταξινομήσεις που έγιναν ήταν πάνω στις κλάσεις Pastry_Vs_Dirtiness, Zscratch_vs_Bumps, Kscratch_Vs_Stains. Συγκεκριμένα και στις τρείς περιπτώσεις τα δεδομένα χωρίστηκαν σε ποσοστό 80% - 20% για τα υποσύνολα εκπαίδευσης και ελέγχου. Επίσης, έγινε κανονικοποίηση και κατασκευή πολυωνυμικών χαρακτηριστικών (n=2). Το μοντέλο που επιλέχθηκε να χρησιμοποιηθεί είναι ένα SVM στο οποίο πραγματοποιήθηκε και ρύθμιση με την GridSearchCV (cv=5). Η χρήση του μοντέλου RF δεν επιλέχθηκε καθώς θέλαμε να δείξουμε ότι ακόμα και με ένα πιο απλό γραμμικό μοντέλο όπως ένα SVM μπορούν να επιτευχθούν υψηλά αποτελέσματα. Επιπλέον, η χρήση ενός ensemble για τόσα λίγα δεδομένα θεωρήθηκε μη κατάλληλη επιλογή. Στην συνέχεια στον Πίνακα 15 παρουσιάζονται τα αποτελέσματα του tuning για το SVM για όλα τα binary classifications που έγιναν.

Πίνακας 15. Τρίτο Στάδιο Ταξινόμησης - Παράμετροι και χρόνοι SVM

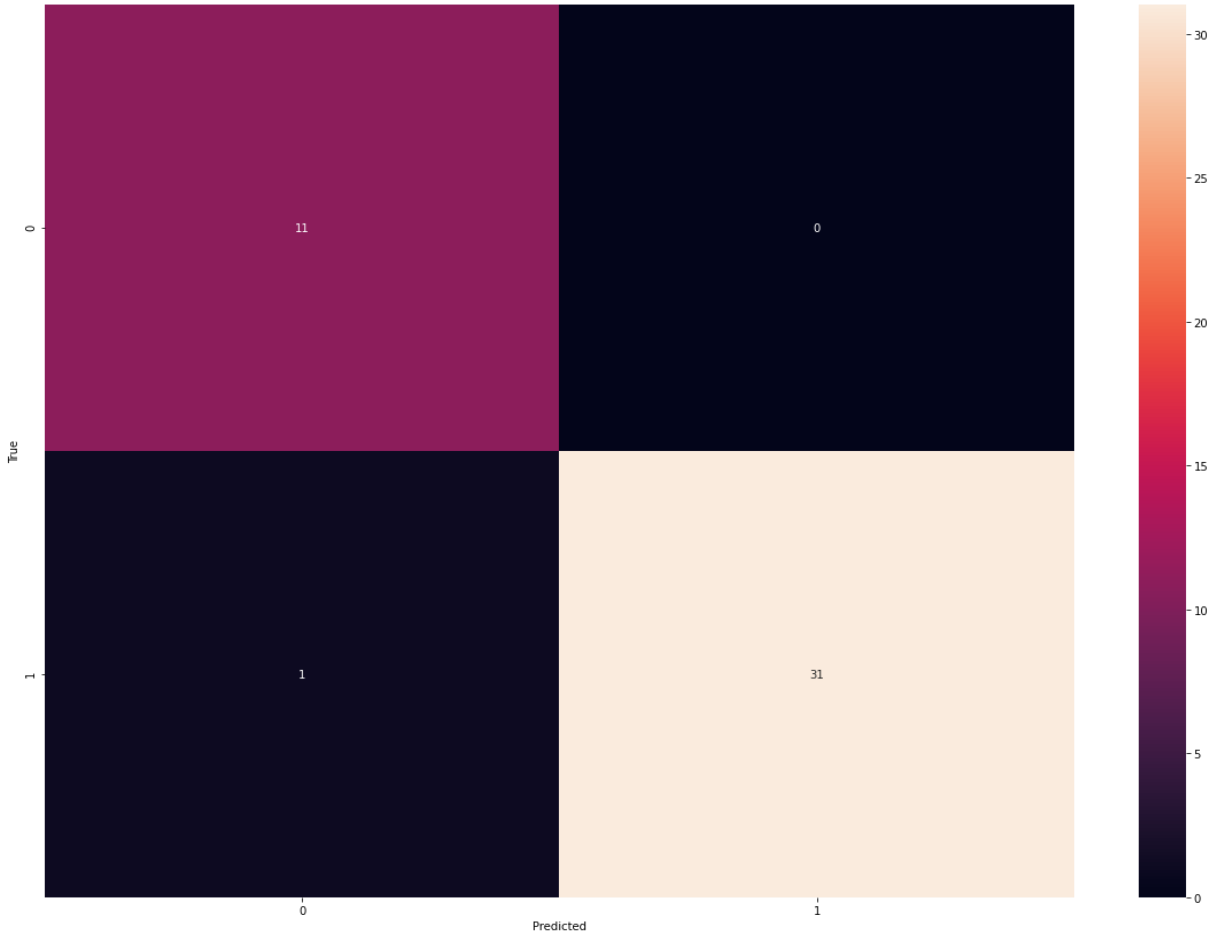
Binary Classifications	**Combinations and Time elapsed**	**Parameters**
Pastry Vs Dirtiness	Done 250 out of 250 | elapsed: 0.8s finished	{'C': 4, 'gamma': 0.05, 'kernel': 'rbf'}
Zscratch_vs_Bumps	Done 250 out of 250 | elapsed: 4.0s finished	'C': 4, 'gamma': 0.05, 'kernel': 'rbf'}
Kscratch_Vs_Stains	Done 250 out of 250 | elapsed: 1.5s finished	{'C': 1, 'gamma': 0.05, 'kernel': 'rbf'}

Παρακάτω παρουσιάζονται σε λογικά διαγράμματα τα 3 μοντέλα που κατασκευάστηκαν καθώς και η αντίστοιχη προ-επεξεργασία των δεδομένων που έγινε. Στην συνέχεια παρουσιάζονται και οι αντίστοιχοι πίνακες σύγχυσης στην Εικόνα 38, στην Εικόνα 40 και Εικόνα 42 καθώς και οι εκθέσεις ταξινόμησης. Όπως φαίνεται τα αποτελέσματα είναι ιδιαίτερα ικανοποιητικά για όλες τις δυαδικές ταξινομήσεις αφού ακόμα και η μικρότερη συνολική ακρίβεια είναι 0.95.

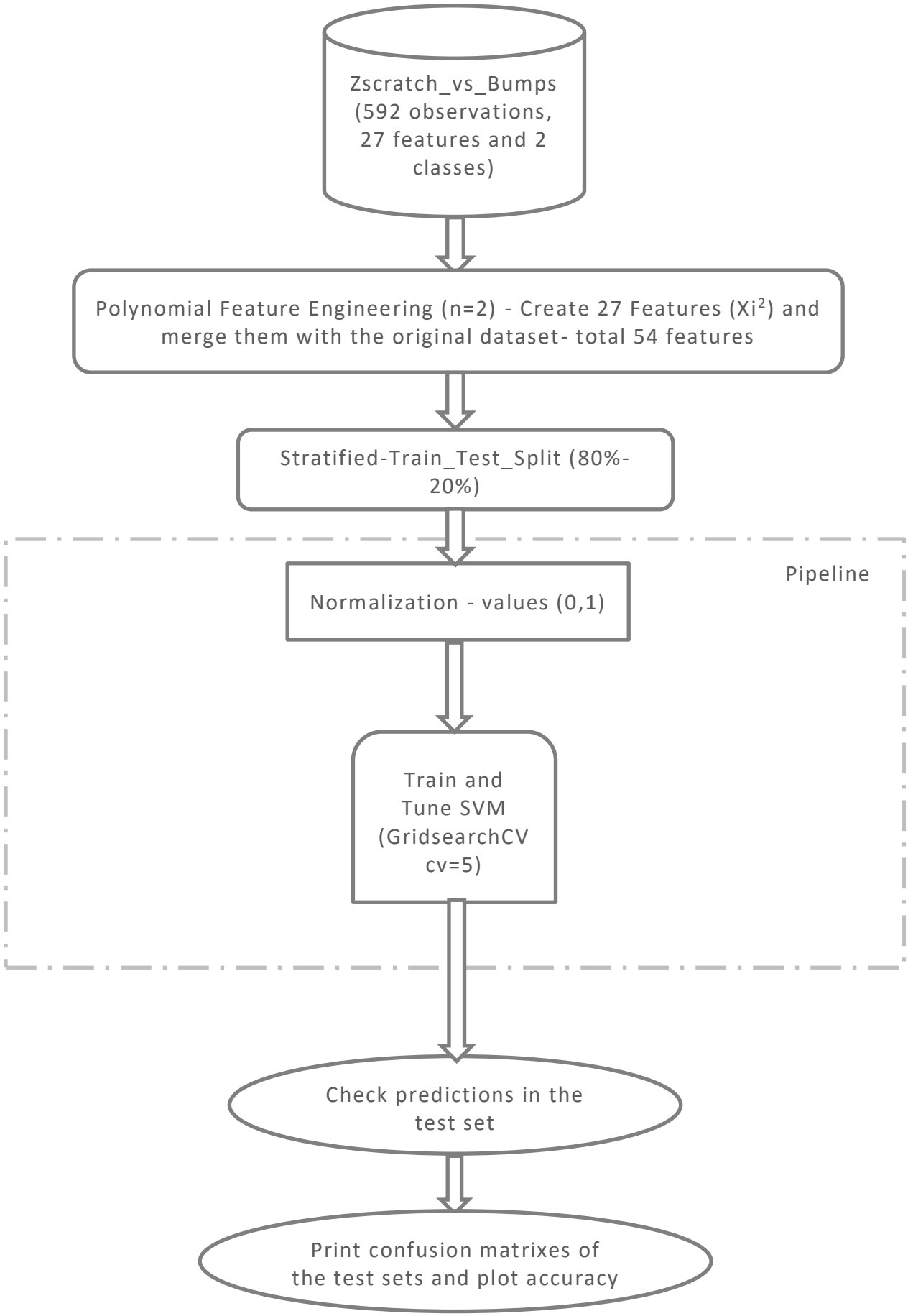


Εικόνα 37. Τρίτο Στάδιο Ταξινόμησης - Pastry Vs Dirtiness

Classification Report:				
	precision	recall	f1-score	support
0	0.92	1.00	0.96	11
1	1.00	0.97	0.98	32
accuracy			0.98	43
macro avg	0.96	0.98	0.97	43
weighted avg	0.98	0.98	0.98	43

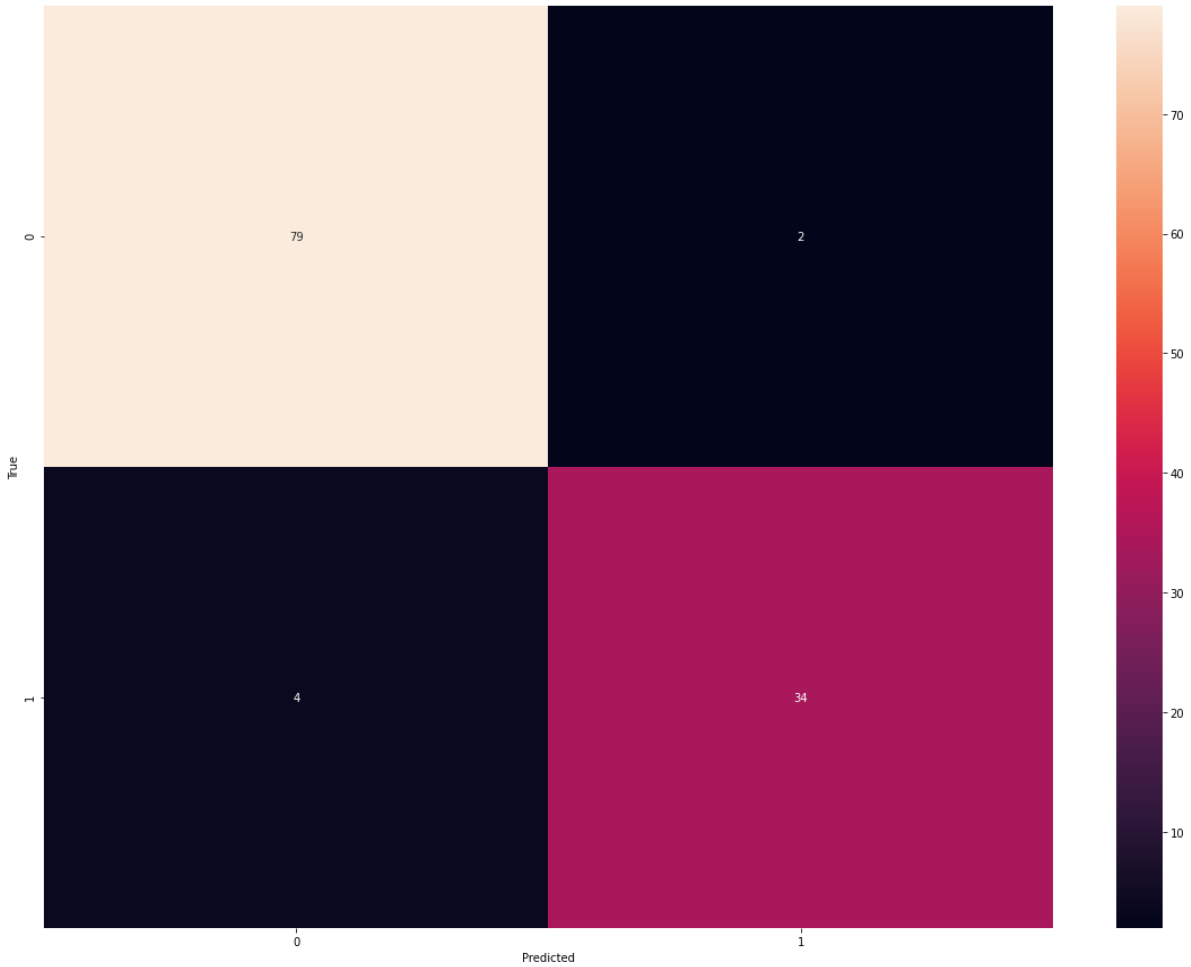


Εικόνα 38. Τρίτο Στάδιο Ταξινόμησης – Πίνακας Σύγχυσης - Pastry Vs Dirtiness

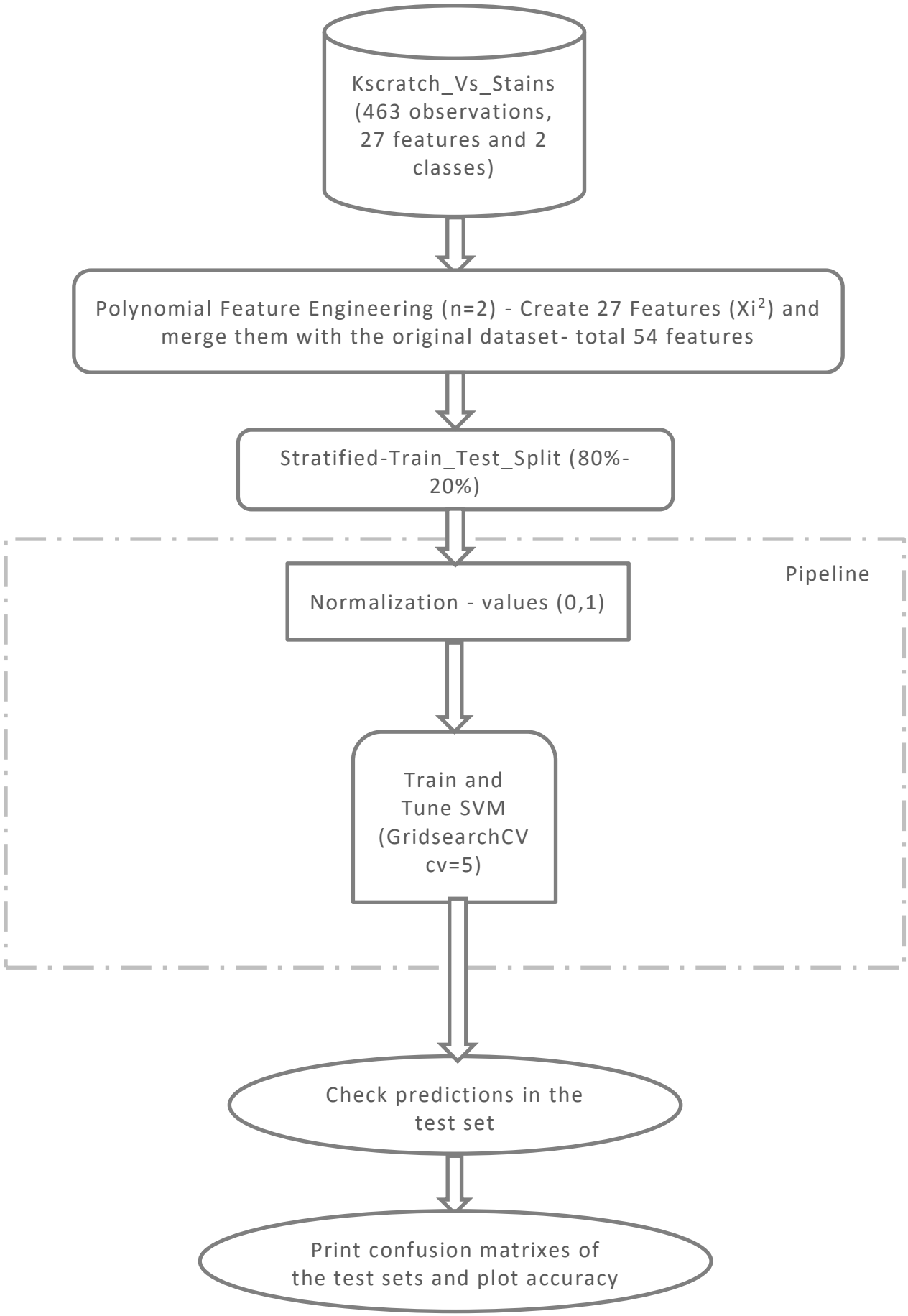


Εικόνα 39. Τρίτο Στάδιο Ταξινόμησης – Z_Scratch Vs Bumps

Classification Report:				
	precision	recall	f1-score	support
0	0.95	0.98	0.96	81
1	0.94	0.89	0.92	38
accuracy			0.95	119
macro avg	0.95	0.94	0.94	119
weighted avg	0.95	0.95	0.95	119

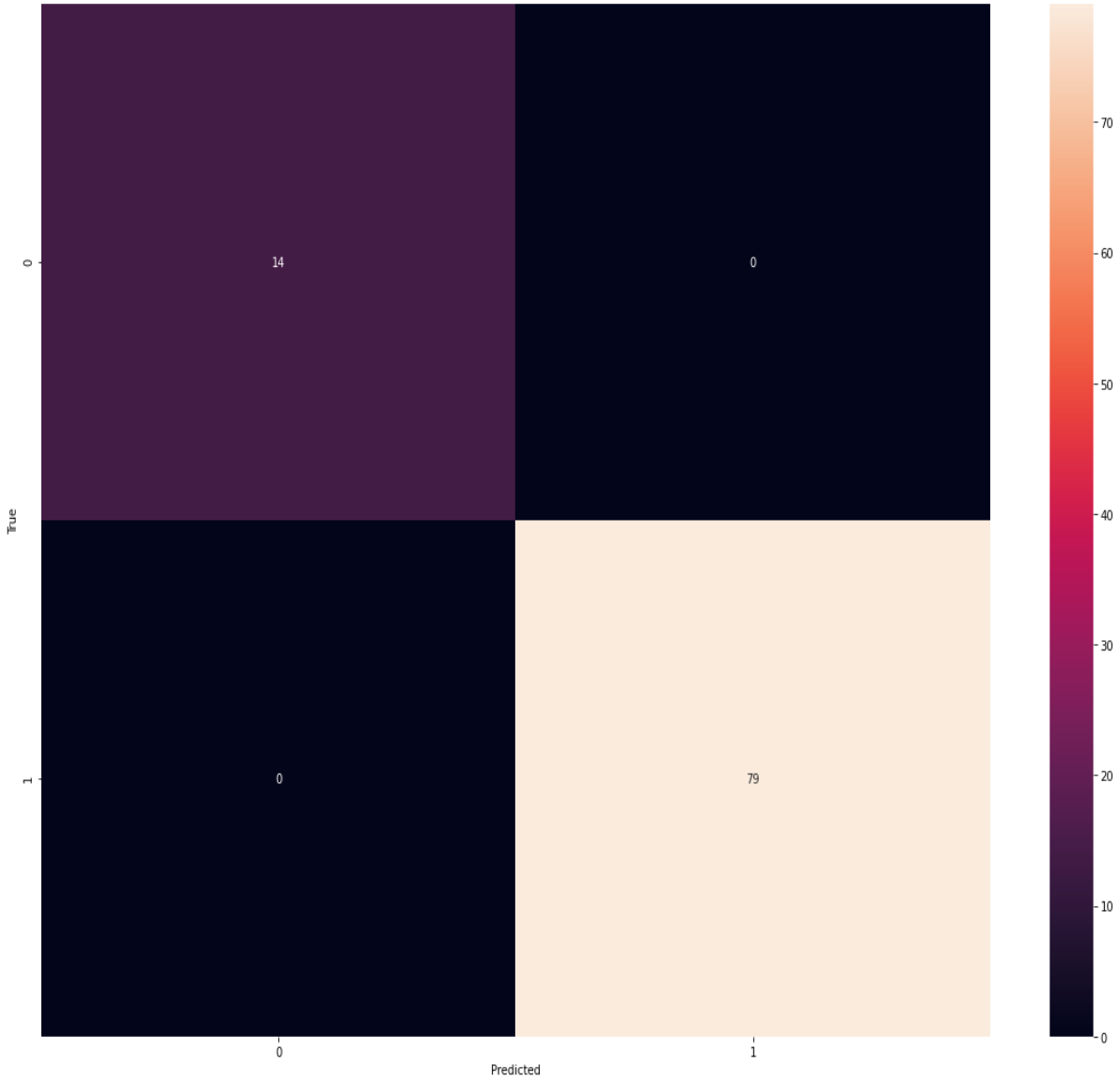


Εικόνα 40. Τρίτο Στάδιο Ταξινόμησης - Πίνακας Σύγχυσης - Z_Scratch Vs Bumps



Εικόνα 41. Τρίτο Στάδιο Ταξινόμησης - Πίνακας Σύγχυσης - Z_Scratch Vs Bumps

Classification Report:				
	precision	recall	f1-score	support
0	1.00	1.00	1.00	14
1	1.00	1.00	1.00	79
accuracy			1.00	93
macro avg	1.00	1.00	1.00	93
weighted avg	1.00	1.00	1.00	93



Εικόνα 42. Τρίτο Στάδιο Ταξινόμησης - Πίνακας Σύγχυσης – K_Scratch Vs Stains

6.3 Συγκεντρωτικά αποτελέσματα και Mutual Information

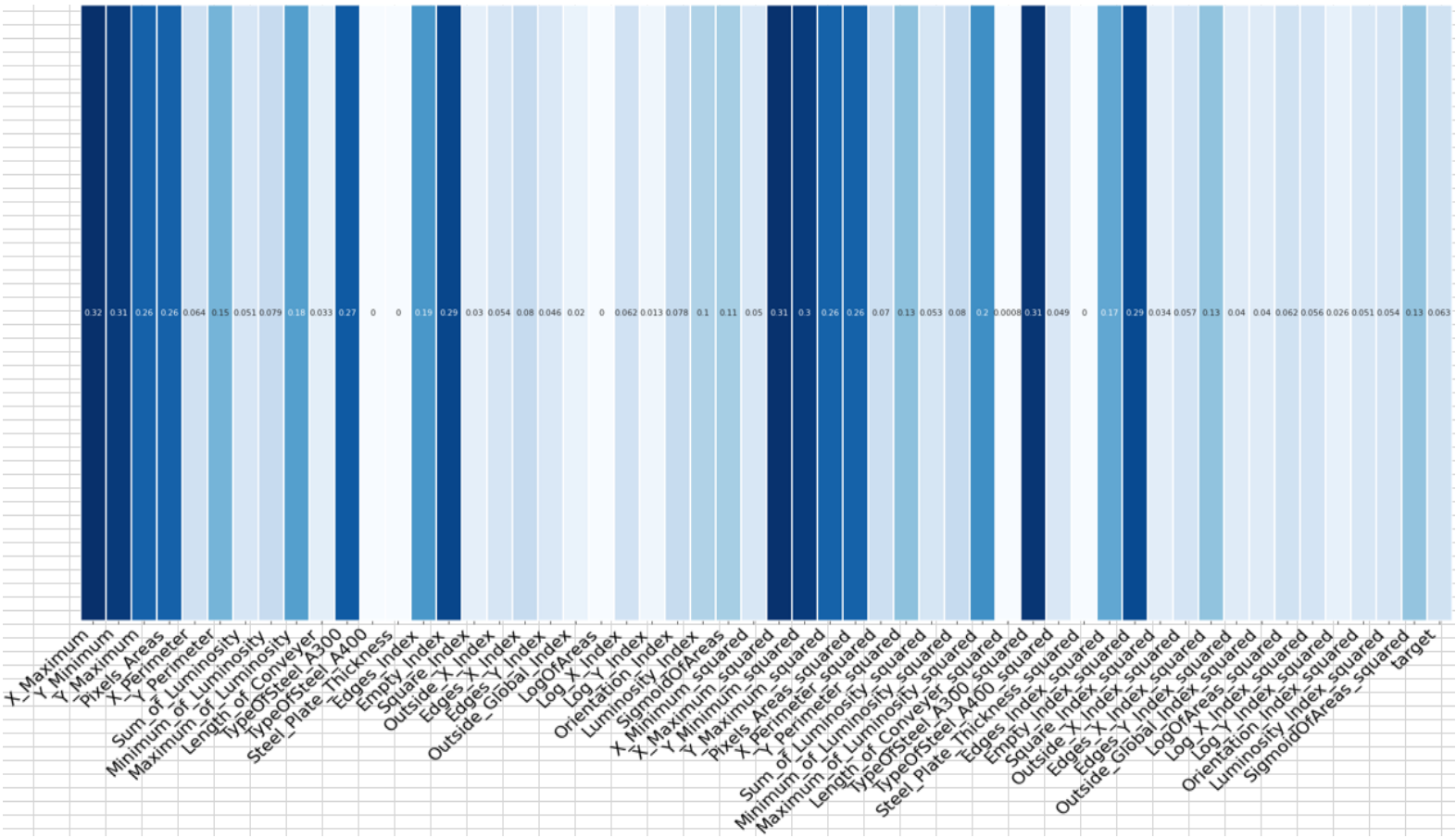
Ο προσαρμοσμένος ταξινομητής που κατασκευάστηκε εμφάνισε πολύ καλά αποτελέσματα που είναι συγκρίσιμα ή/και καλύτερα από τα αποτελέσματα της υπάρχουσας βιβλιογραφίας [13] [26] [30]. Ο συγκεκριμένος ταξινομητής λειτουργεί σειριακά, μέσω των 3 σταδίων ταξινόμησης που αναπτύχθηκαν. Οπότε τα συγκεντρωτικά του αποτελέσματα θα πρέπει να εξεταστούν με προσοχή. Αυτό που παρατηρήθηκε είναι ότι οι πιο πολλές κλάσεις είναι διαχωρίσιμες μεταξύ τους σε μεγάλο βαθμό και αυτό φαίνεται από τους πίνακες σύγχυσης των δυαδικών ταξινομήσεων. Επιπροσθέτως, πρέπει να αναφερθεί ότι οι παρατηρήσεις που περνούσαν από το ένα στάδιο στο άλλο ήταν ολόκληρα τα υποσύνολα των σετ δεδομένων και όχι αυτά που προέκτυπταν από τα μοντέλα του κάθε σταδίου. Κάτι τέτοιο, πραγματοποιηθήκε γιατι αποτελεί πιο συντηρητική πρακτική. Τα αποτελέσματα που προκύπτουν έχουν γίνει για την δυσμενέστερη περίπτωση καθώς λανθασμένες ταξινομήσεις παρατηρήσεων των προηγούμενων σταδίων επιτρέπεται να μεταφέρονται στα επόμενα στάδια.

Πίνακας 16. Συνολικά Αποτελέσματα - Προσαρμοσμένος ταξινομητής

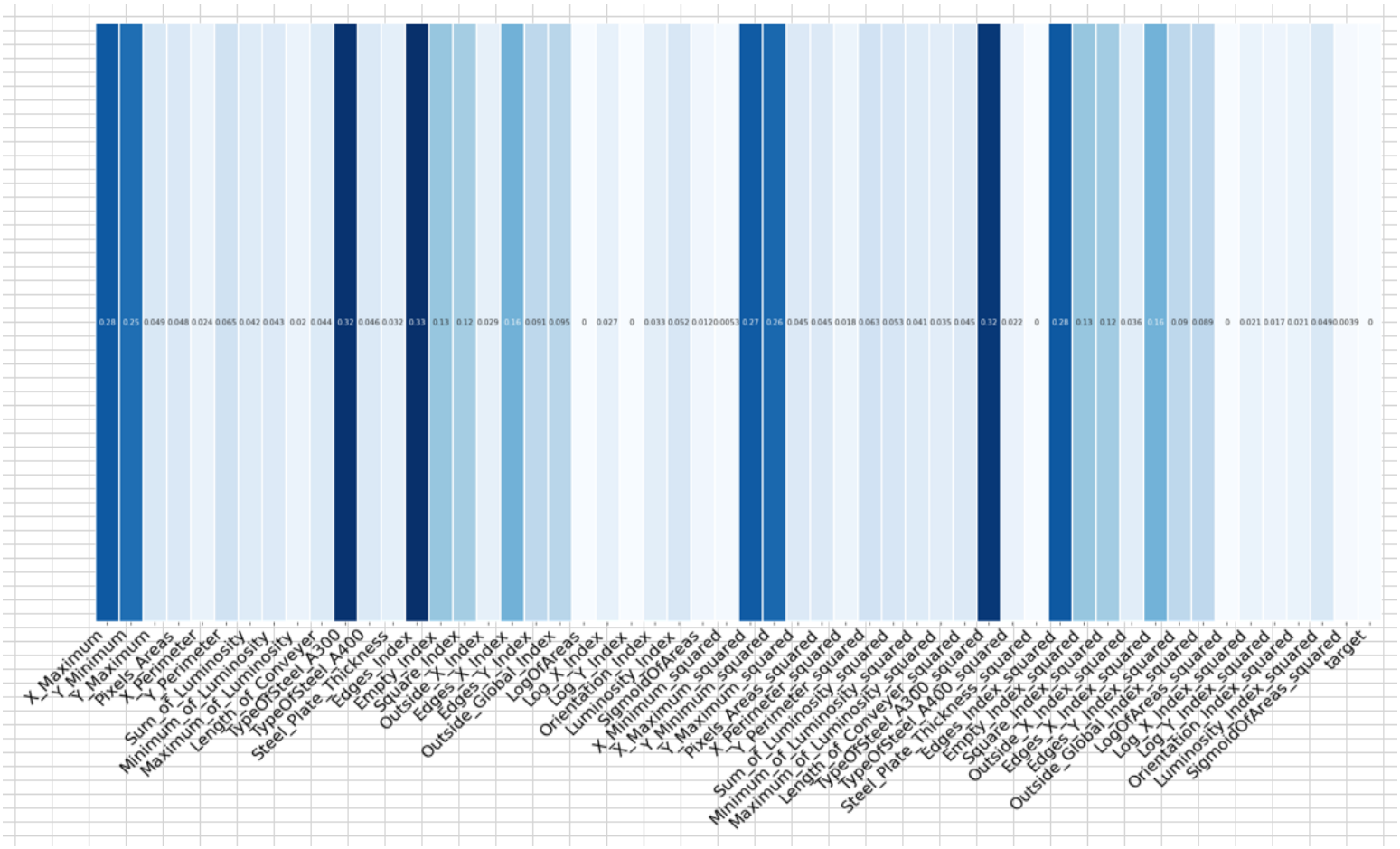
Three Stages of Classification	**Accuracy**
First stage – Common Faults Vs Other Faults	0.92
Second stage – Imaginary Classes - A Vs B Vs C	0.94
Third Stage – Pastry Vs Dirtiness	0.98
Third Stage – Z_Scratch Vs Bumps	0.95
Third Stage – K_Scratch Vs Stains	1.00

Παρακάτω, υπολογίζονται οι δείκτες σημαντικότητας με mutual information των χαρακτηριστικών για το πρώτο και το τρίτο στάδιο που έγιναν με τον προσαρμοσμένο ταξινομητή. Αυτό πραγματοποιήθηκε καθώς θέλαμε να επιβεβαιώσουμε ότι η σημαντικότητα των χαρακτηριστικών επηρεάζεται σε μεγάλο βαθμό ως προς τι κλάσεις του προβλήματος πολλαπλών κλάσεων. Δηλαδή κάποια χαρακτηριστικά μπορεί να είναι σημαντικά ως προς κάποιες κλάσεις αλλά όχι ως προς κάποιες άλλες. Έτσι η αφαίρεση χαρακτηριστικών από το αρχικό πρόβλημα που έγινε με το RF (model based feature importances) μπορεί να μετράει

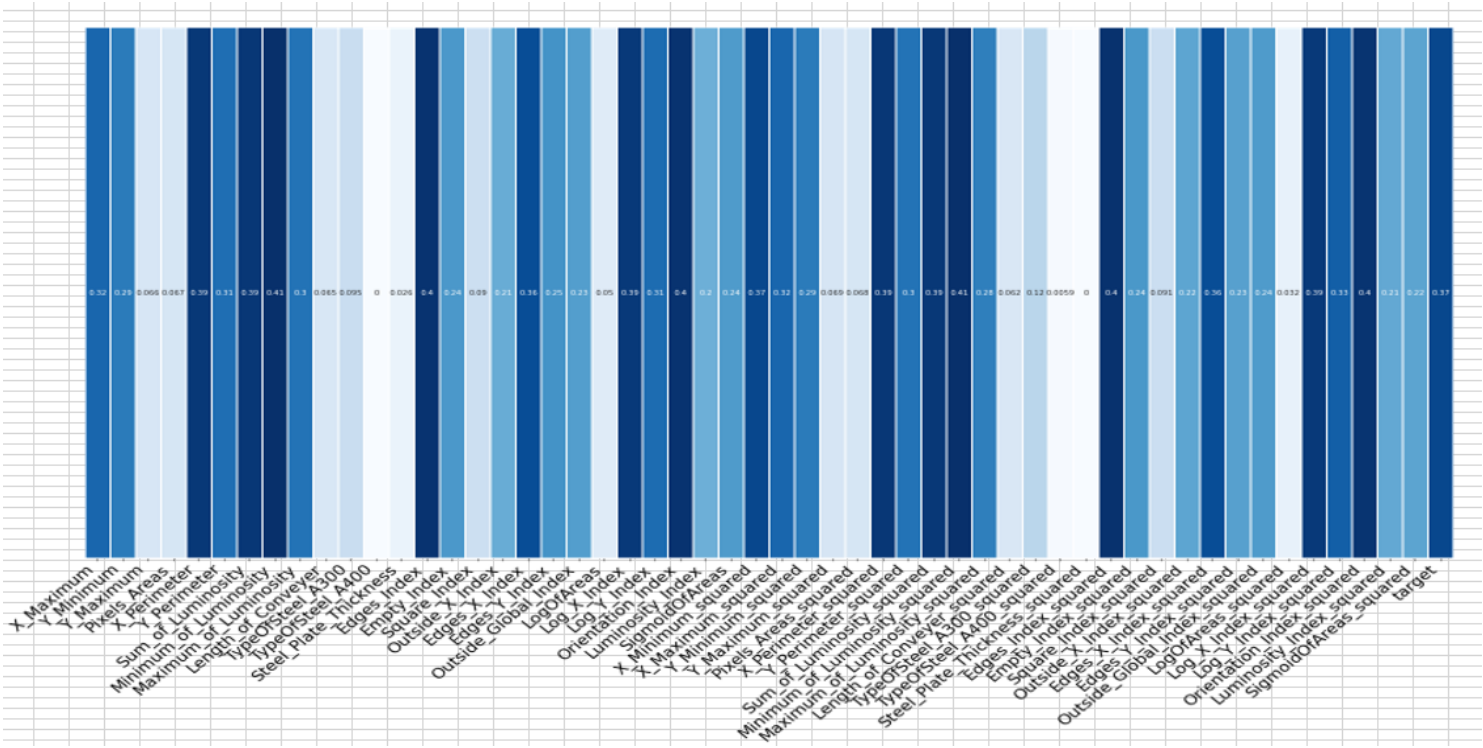
την σημαντικότητα των χαρακτηριστικών ως προς όλες τις κλάσεις αλλά αν γίνει ένας επιμερισμός του προβλήματος πολλαπλών κλάσεων σε πολλά δυαδικά προβλήματα, τα σημαντικά χαρακτηριστικά είναι αρκετά διαφορετικά. Δηλαδή όπως θα περίμενε κανείς η σημαντικότητα των χαρακτηριστικών βλέπουμε ότι αλλάζει σε μεγάλο βαθμό ανάλογα με τις κλάσεις που πάμε να διαχωρίσουμε. Πιο συγκεκριμένα, αυτό που παρατηρούμε είναι ότι για τις κλάσεις Pastry και Dirtiness τα πιο σημαντικά χαρακτηριστικά είναι το X_Maximum, το X_Minimum, το Empty_Index και τετράγωνα τους. Ενώ στην περίπτωση των κλάσεων Z_Scratch και Bumps τα πιο σημαντικά χαρακτηριστικά φαίνονται να είναι το Type_Of_Steel_A300 και το Edges_Index και τα τετράγωνα τους. Τέλος, για τις κλάσεις K_Scratch και Stains το πιο σημαντικό χαρακτηριστικά σύμφωνα με την τεχνική mutual information που έγινε είναι το Minimum_Of_Luminosity και το τετράγωνο του που στις δύο προηγούμενες περιπτώσεις είχε χαμηλό δείκτη σημαντικότητας. Κάτι τέτοιο είναι φυσικό να συμβαίνει σε ένα πρόβλημα πολλαπλών κλάσεων καθώς κάποια χαρακτηριστικά διαχωρίζουν κάποιες κλάσεις καλύτερα από κάποιες άλλες και αντίστοιχα. Στην συνέχεια βρίσκονται τα αντίστοιχα διαγράμματα στις Εικόνες 43-45.



Εικόνα 43. Mutual Information - Pastry Vs Dirtiness



Εικόνα 44. Mutual Information - Z_Scratch Vs Bumps



Εικόνα 45. Mutual Information - K_Scratch Vs Stain

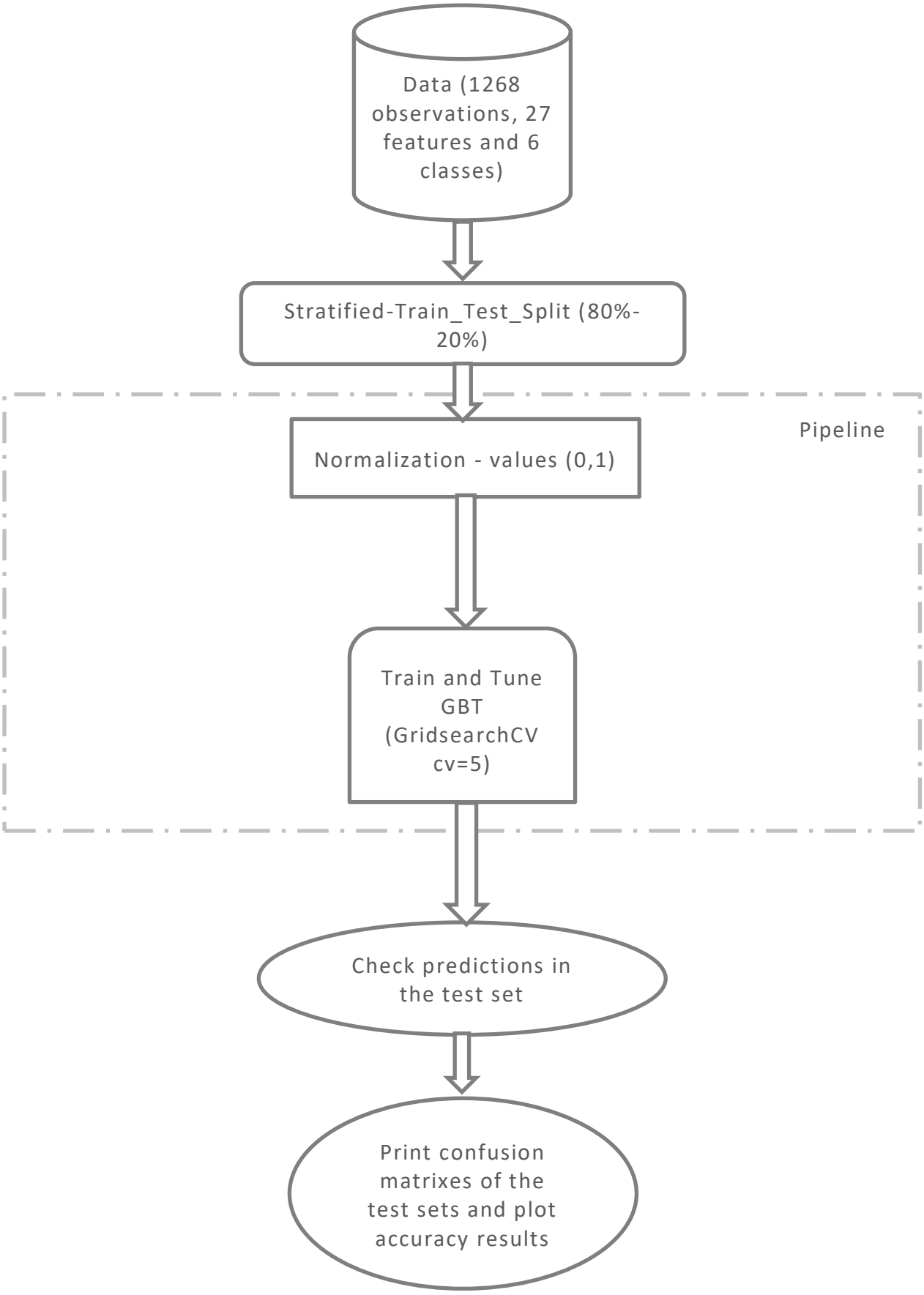
Κεφάλαιο 7: ΤΡΟΠΟΠΟΙΗΣΗ ΤΟΥ ΠΡΟΒΛΗΜΑΤΟΣ

7.1 Εισαγωγή

Η τελευταία κλάση του προβλήματος Other Faults, όπως και το όνομα της δηλώνει, δεν περιέχει πληροφορία σχετικά με το είδος του σφάλματος στις πλάκες χάλυβα ενώ οι προηγούμενες 6 κλάσεις του σετ δεδομένων αποτελούν διακριτά συγκεκριμένα σφάλματα. Πραγματοποιήθηκε η εφαρμογή του μοντέλου μηχανικής μάθησης GBT με αφαίρεση της τελευταίας κλάσης Other Faults. Σκοπός ήταν να δούμε τα αποτελέσματα του μοντέλου χωρίς την τελευταία κλάση κάνοντας την υπόθεση ότι η κλάση Other Fault αποτελεί τύπο ελαττώματος που συνδυάζει τουλάχιστον άλλους δύο τύπους ελαττωμάτων των προηγούμενων κλάσεων. Επιπλέον, τα διαγράμματα που έχουν παραχθεί για την οπτικοποίηση των δεδομένων μέσω PCA και t – SNE μας δείχνουν ότι η τελευταία κλάση δεν είναι σε κανένα βαθμό διαχωρίσιμη από τις υπόλοιπες, καθώς οι τιμές της παρουσιάζονται και στις άλλες κλάσεις. Αυτό σημαίνει ότι τα χαρακτηριστικά του προβλήματος δεν διαφοροποιούνται αρκετά ως προς την τελευταία κλάση σε σχέση με το πώς διαφοροποιούνται ως προς τις υπόλοιπες. Κάτι τέτοιο θα μπορούσε να συμβαίνει εφόσον η τελευταία κλάση περιείχε πολλούς τύπους διακριτών ελαττωμάτων.

7.2 Διαδικασία Επίλυσης με Gradient Boosting Trees

Τα δεδομένα αρχικά διαβάστηκαν και τοποθετήθηκαν σε πίνακες δεδομένων (dataframes - pandas) αφού αφαιρέθηκε η τελευταία κλάση. Σε επόμενο στάδιο χωρίστηκαν (train_test_split) σε σετ εκπαίδευσης (80%) και σετ ελέγχου (20%). Κατασκευάστηκε ακολουθία μετασχηματισμών (pipeline) στην οποία τα δεδομένα για την εκπαίδευση κανονικοποιούντουσαν (normalization) πριν την είσοδος τους στο μοντέλο. Για το GBT πραγματοποιήθηκε ρύθμιση των παραμέτρων του με την GridSearch CV με cv=5 (cross – validations). Να αναφερθεί ότι λόγω της ενίσχυσης το συγκεκριμένο μοντέλο ήταν το πιο χρονοβόρο από την πλευρά της εκπαίδευσης και της ρύθμισης των παραμέτρων ακόμα και με μία λιγότερη κλάση που αποτελεί βεβαίως και την κλάση με τις πιο πολλές παρατηρήσεις (majority class). Παρακάτω, στην Εικόνα 46 παρουσιάζεται λογικό διάγραμμα του κώδικα.



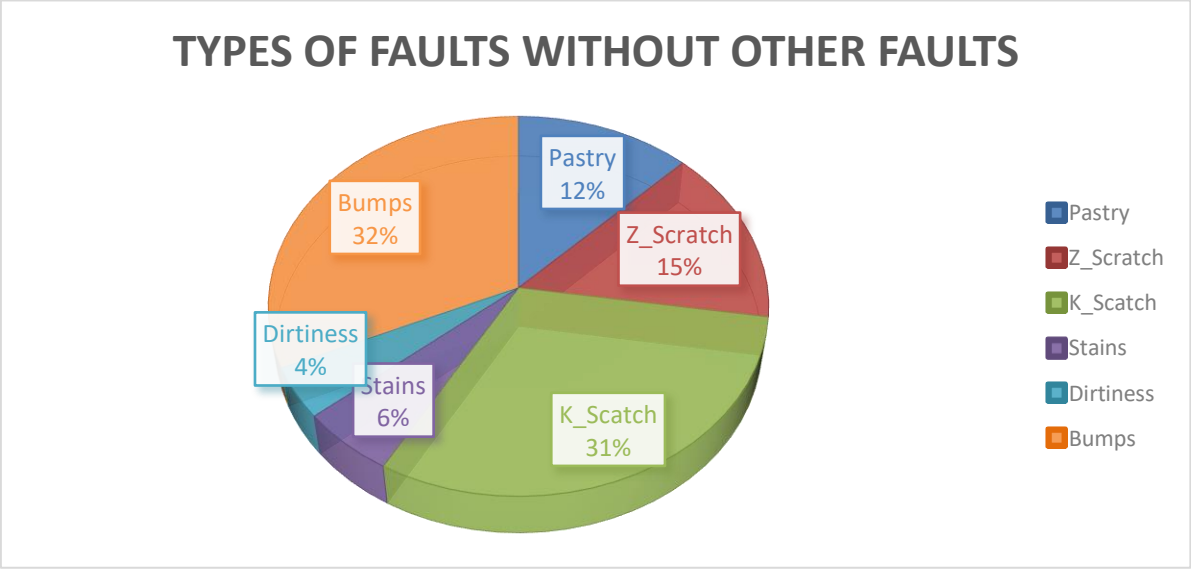
Εικόνα 46. Λογικό διάγραμμα κώδικα - GBT

7.3 Αποτελέσματα

Παρατηρήθηκαν πολύ καλά αποτελέσματα για το tuned GBT χωρίς την τελευταία κλάση, καθώς η ορθότητα ήταν 0.930. Στην συνέχεια στον Πίνακα 17 παρουσιάζεται ο πίνακας σύγχυσης του μοντέλου, όπως και οι αριθμοί συνδυασμών, χρόνοι εκπαίδευσης αλλά και το classification report για τις 6 κλάσεις. Σημαντικό είναι να αναφερθεί ότι το συγκεκριμένο μοντέλο θα μπορούσε να συνδυαστεί για την ταξινόμηση και των 7 τύπων ελαττωμάτων με τον προσαρμοσμένο ταξινομητή στο δεύτερο στάδιο ταξινόμησης. Τέλος, από τα αποτελέσματα καταλαβαίνουμε ότι η πρώτη κλάση Pastry έχει αρκετά μικρότερο ποσοστό σωστής ταξινόμησης από όλες τις υπόλοιπες που το GBT βρίσκει με υψηλές αποδόσεις. Αυτό είναι κάτι το οποίο το παρατηρούμε και στα αποτελέσματα του Κεφάλαιου 4. Βεβαία παρατηρείται σημαντική βελτίωση της Pastry (4-5%) στην απόδοση με την αφαίρεση της κλάσης Other Faults.

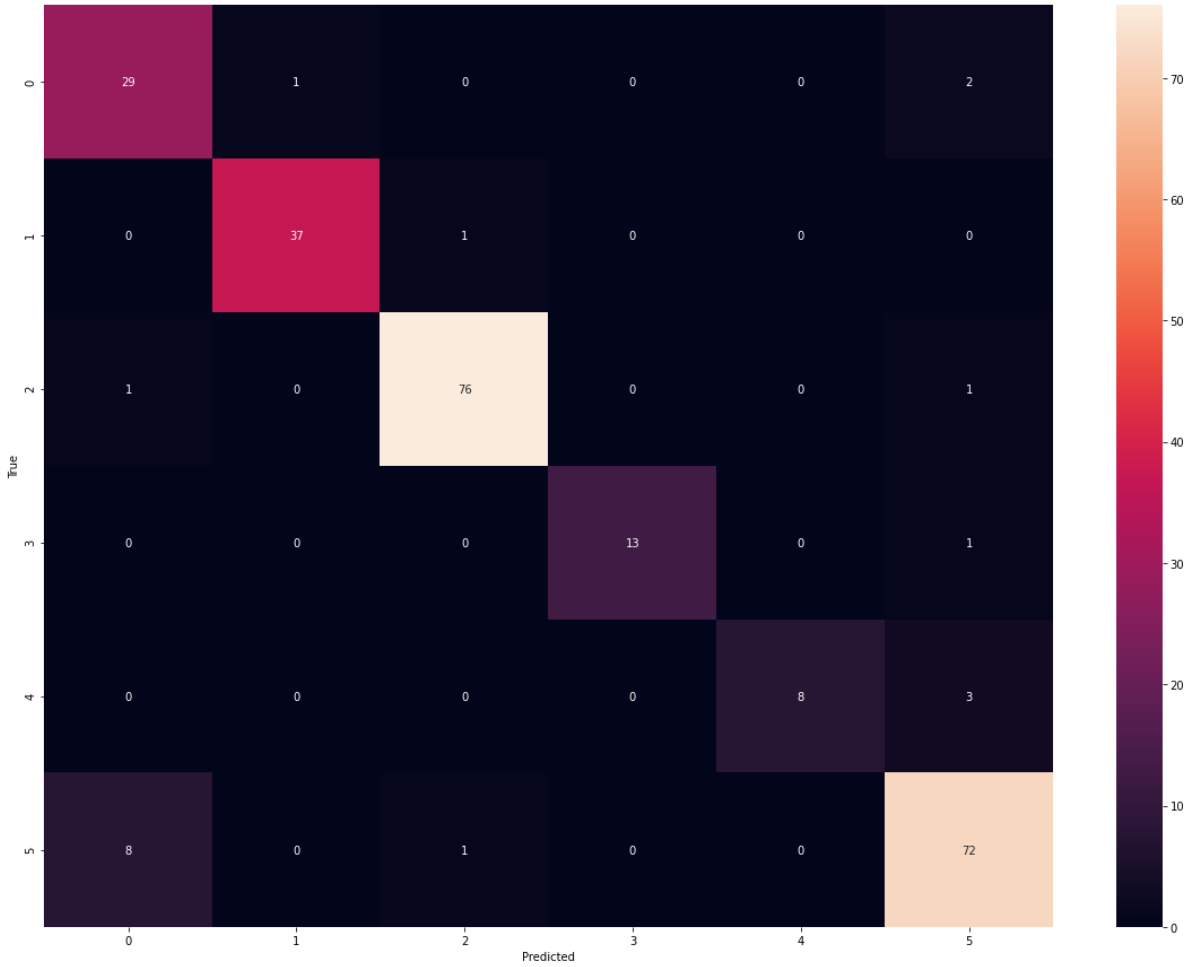
Πίνακας 17. GBT - parameters - combinatation and time elapsed

Multiclass Classification with GBT	Combinations and Time elapsed	Parameters
Pastry Z_Scratch K_Scatch Stains Dirtiness Bumps	Done 2160 out of 2160 | elap sed: 364.4min finished	{'criterion': 'mae', 'learning_rate': 0.15, 'loss': 'deviance', 'max_depth': 5, 'max_features': 'sqrt', 'min_samples_leaf': 12, 'min_samples_split': 12, 'n_estimators': 500, 'subsample': 1.0}



Εικόνα 47. Τύποι ελαττωμάτων χωρίς την Other Faults

Classification Report:				
	precision	recall	f1-score	support
1	0.76	0.91	0.83	32
2	0.97	0.97	0.97	38
3	0.97	0.97	0.97	78
4	1.00	0.93	0.96	14
5	1.00	0.73	0.84	11
6	0.91	0.89	0.90	81
accuracy			0.93	254
macro avg	0.94	0.90	0.91	254
weighted avg	0.93	0.93	0.93	254



Εικόνα 48. Confusion Matrix - GBT

Κεφάλαιο 8: ΣΥΜΠΕΡΑΣΜΑΤΑ

Στην παρούσα μεταπτυχιακή εργασία υλοποιήθηκαν 22 σενάρια μοντέλων μηχανικής μάθησης με σκοπό την αναγνώριση των τύπων ελαττωμάτων χαλύβδινων πλακών (faulty steel plates dataset). Επιπροσθέτως, εφαρμόστηκε ένα μεγάλο πλήθος τεχνικών προ – επεξεργασίας των δεδομένων καθώς και ανάλυσή τους. Τα αποτελέσματα που παράγονται χρησιμοποιούνται για τη σύγκριση των μοντέλων που δημιουργήθηκαν, καθώς και των διάφορων τεχνικών. Η δημιουργία μοντέλων με αρκετά μεγάλη συνολική ακρίβεια μας οδηγεί στο συμπέρασμα ότι η χρήση μηχανικής μάθησης στην Παραγωγή και πιο συγκεκριμένα στον τομέα της Ποιότητας Ελέγχου των εργοστασίων μπορεί να δημιουργήσει θετικά αποτελέσματα.

Όσον αφορά ποιοτικά συμπεράσματα για τα μοντέλα, η υλοποίηση του προσαρμοσμένου ταξινομητή που έγινε αποτελεί ένα αυτοματοποιημένο μοντέλο μηχανικής μάθησης που θα μπορούσε να βρει εφαρμογή και σε άλλες περιπτώσεις συνόλων δεδομένων που προέρχονται από τον χώρο της Βιομηχανικής Παραγωγής. Η ρύθμιση των παραμέτρων (tuning) που έγινε στα μοντέλα έδειξε αποτελέσματα με σημαντική βελτίωση σε σχέση με την απλή εφαρμογή τους. Η χρήση τεχνικών προ-επεξεργασίας των χαρακτηριστικών (feature engineering) με πρόσθεση των πολυωνύμων δευτέρου βαθμού στο αρχικό σετ δεδομένων είχε πολύ καλά αποτελέσματα. Επιπροσθέτως, η αντιμετώπιση της ανομοιογένειας των κλάσεων ως προς τις παρατηρήσεις τους (class imbalances) με την παραγωγή συνθετικών δεδομένων είναι κάτι που φαίνεται να δίνει λύσεις και μπορεί να εφαρμοστεί σε πλήθος προβλημάτων της Παραγωγής, ειδικά σε περιπτώσεις που τα δεδομένα περιέχουν λάθη ή ελλείψεις σχετικά με τον τρόπο καταγραφής τους. Ακόμα, παρατηρήθηκε ότι η σημαντικότητα των χαρακτηριστικών είναι κάτι που μεταβάλλεται ανάλογα με την κλάση, δηλαδή τον τύπο του ελαττώματος. Αυτό φαίνεται και από τα αποτελέσματα του mutual information στις δυϊκές κλάσεις του προσαρμοσμένου ταξινομητή. Κάτι τέτοιο, δείχνει σύνδεση χαρακτηριστικών με τύπους ελαττωμάτων χάλυβα και αποτελεί πληροφορία για τα δεδομένα (insight). Η αξιοποίηση μεθόδων ανάλυσης δεδομένων και οπτικοποίησής τους πριν την εφαρμογή μοντέλων μηχανικής μάθησης αποδυκνύεται πολύ χρήσιμή, καθώς εκεί στηρίχθηκε η επιλογή για την ένωση των κλάσεων που έγινε στο προσαρμοσμένο ταξινομητή με υψηλά αποτελέσματα ορθότητας. Στο δεύτερο και στο τρίτο στάδιο του προσαρμοσμένου ταξινομητή χρησιμοποιήθηκε και πολυωνυμική (n= 2) προ-επεξεργασία των δεδομένων αφού το RF με αυτόν τον τρόπο εμφάνισε τα καλύτερα αποτελέσματα. Τέλος, η ένωση των κύριων συνιστωσών (principal components) με τα 27 χαρακτηριστικά του σετ δεδομένων φαίνεται να λειτούργησε θετικά για κάποια μοντέλα. Το πλήθος των χαρακτηριστικών δεν ήταν σε κανένα βαθμό αρκετά μεγάλο ώστε να χρησιμοποιηθούν μόνο οι κύριες συνιστώσες.

Ποσοτικά συμπερασμάτα που προέκυψαν για την σύγκριση μοντέλων και τεχνικών παρουσιάζονται στην συνέχεια σε κουκκίδες (bullet points).

- Η απλή εφαρμογή του μοντέλου RF που υπήρξε και αυτό με τα καλύτερα αποτελέσματα δίνει ορθότητα 0.78 στο συγκεκριμένο σετ δεδομένων. Με την πολυωνυμική προ-επεξεργασία των δεδομένων που έγινε και την ρύθμιση των παραμέτρων του μοντέλου, τα αποτελέσματα έφτασαν στο 0.825.

- Η μεγαλύτερη αύξηση ως προς την ορθότητα ταξινόμησης έφτασε στο 0.910 και επιτεύχθηκε με την παραγωγή συνθετικών δεδομένων (ADASYN) και την απλή ρύθμιση της παράμετρου των δέντρων του RF στα 80.

- Ο προσαρμοσμένος ταξινομητής τριών σταδίων που κατασκευάστηκε έδωσε υψηλά αποτελέσματα που ξεπερνούν το 0.90 της ορθότητας ταξινόμησης σε κάθε στάδιο.

- Για τα υπόλοιπα μοντέλα υπήρξαν βελτιώσεις με τις διαφορετικές επεξεργασίες που έγιναν, καθώς η απλή εφαρμογή των μοντέλων SVM και k-NN δίνει αποτελέσματα ορθότητας 0.75 και 0.71 αντίστοιχα. Για το SVM και το k-NN πραγματοποιήθηκε βελτίωση της ορθότητας λίγο παραπάνω από το 0.77 και 0.73 αντίστοιχα κάνοντας PCA - ενώνωντας τις κύριες συνιστώσες με το αρχικό σετ δεδομένων και πραγματοποιώντας ρύθμιση παραμέτρων. Επίσης, το k-NN με την παραπάνω προ-επεξεργασία είχε παρόμοια απόδοση και με την απλή ρύθμιση των παραμέτρων του.

- Το μοντέλο LR που ήταν το πιο απλό που χρησιμοποιήθηκε έφτασε το 0.75 με την ρύθμιση των παραμέτρων και την προσθήκη πολυωνύμων δευτέρου βαθμού.

- Τα χαμηλότερα αποτελέσματα για όλα τα μοντέλα παράχθηκαν με την μέθοδο της αφαίρεσης χαρακτηριστικών λόγω σημαντικότητας που προέκυψε από το RF (model based feature importances).

- Με την αφαίρεση της κλάσης Other Faults το μοντέλο GBT αφού ρυθμίστηκε έδωσε αποτελέσματα ορθότητας 93%. Οι διακριτοί τύποι ελαττωμάτων διαχωρίζονται σε πολύ υψηλό ποσοστό.

Το προσδοκόμενο της παρούσας μεταπτυχιακής εργασίας και των μοντέλων που κατασκευάστηκαν είναι η εφαρμογή τους σε νέες μετρήσεις από την Παραγωγή. Με αυτόν τον τρόπο, θα μεγάλωνε και το σετ δεδομένων που υπάρχει ώστε να δοκιμαστούν και αλγόριθμοι βαθιάς μηχανικής μάθησης (deep learning). Επιπροσθέτως, θα μπορούσε σε πρώτο στάδιο να πραγματοποιηθεί έρευνα για την δημιουργία κατάλληλου λογισμικού επεξεργασίας εικόνας και την παραγωγή χαρακτηριστικών από φωτογραφίες ελασμάτων χάλυβα. Στην συνέχεια ο κώδικας για την επεξεργασία εικόνας θα ήταν δυνατόν να ενωθεί σειριακά με τον προσαρμοσμένο ταξινομητή ή και άλλων μοντέλων που έγιναν σε αυτή την εργασία ως ένα ενιαίο λογισμικό. Συγκρίσεις ως προς την ορθότητα ταξινόμησης και το χρόνο υπολογισμού θα μπορούσαν να γίνουν σε φωτογραφίες πλακών χάλυβα του ολοκληρωμένου συστήματος κώδικας επεξεργασίας εικόνας – προσαρμοσμένος ταξινομητής με νευρωνικά δίκτυα CNNs ή άλλα μοντέλα που χρησιμοποιούνται για vision και αναγνώριση ελαττωμάτων από εικόνες στην Παραγωγή. Τέλος, σημαντική μελλοντική δουλειά και επέκταση της συγκεκριμένης εργασίας θα ήταν και η ανάπτυξη λογισμικού για την αυτόματη επιλογή των ενώσεων που έγιναν στον προσαρμοσμένο ταξινομητή μέσω οπτικοποίησης και ανάλυσης των δεδομένων.

ΒΙΒΛΙΟΓΡΑΦΙΑ

[1] R. Schlaepfer, “Industry 4.0. Challenges and solutions for the digital transformation and use of exponential technologies,” *Deloitte*, pp. 1–30, 2015.

[2] S. Francois Cardarelli , Materials Handbook ,A Concise Desktop Reference , 3rd Edition (2017), “No Title.”

[3] P. G. Benardos and G. C. Vosniakos, “Predicting surface roughness in machining: A review,” *Int. J. Mach. Tools Manuf.*, vol. 43, no. 8, pp. 833–844, 2003, doi: 10.1016/S0890-6955(03)00059-2.

[4] A. M. Sam and C. Balaji, *Artificial Intelligence and Evolutionary Computations in Engineering Systems*, vol. 394. 2016.

[5] Y. Lecun, Y. Bengio, and G. Hinton, “Deep learning,” *Nature*, vol. 521, no. 7553, pp. 436–444, 2015, doi: 10.1038/nature14539.

[6] M. Li, X. Wang, and Z. Ma, “Steel defect detection with high-frequency camera images FA 19-20 CS 229 Project.”

[7] A. Sharma, E. Vans, D. Shigemizu, K. A. Boroevich, and T. Tsunoda, “DeepInsight: A methodology to transform a non-image data to an image for convolution neural network architecture,” *Sci. Rep.*, vol. 9, no. 1, pp. 1–7, 2019, doi: 10.1038/s41598-019-47765-6.

[8] S. of I. and C. S. I. Frank, A. and Asuncion, A. (2010) UCI Machine Learning Repository. University of California, “No Title.”

[9] https://www.kaggle.com/uciml/faulty-steel-plates, “No Title.” .

[10] V. Programmers, A. Vem, M. S. Nanduri, M. Vallamkonda, and N. Gunti, “Fault Diagnosis in Steel Plates using Machine Learning,” 2020.

[11] Https://www.openml.org/d/1504, “=.” .

[12] D. Simić, V. Svirčević, and S. Simić, “An approach of steel plates fault diagnosis in multiple classes decision making,” *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 8480 LNAI, pp. 86–97, 2014, doi: 10.1007/978-3-319-07617-1_8.

[13] A. K. Srivastava, “Comparison Analysis of Machine Learning algorithms for Steel Plate Fault Detection,” no. May, pp. 1231–1234, 2019.

[14] T. Nkonyana, Y. Sun, B. Twala, and E. Dogo, “Performance evaluation of data mining techniques in steel manufacturing industry,” *Procedia Manuf.*, vol. 35, pp. 623–628, 2019, doi: 10.1016/j.promfg.2019.06.004.

[15] P. Liashchynskyi and P. Liashchynskyi, “Grid Search, Random Search, Genetic Algorithm: A Big Comparison for NAS,” *arXiv*, Dec. 2019, Accessed: Apr. 29, 2021. [Online]. Available: http://arxiv.org/abs/1912.06059.

[16] M. Fakhr and A. M. Elsayad, “Steel plates faults diagnosis with data mining models,” *J. Comput. Sci.*, vol. 8, no. 4, pp. 506–514, 2012, doi: 10.3844/jcssp.2012.506.514.

[17] S. P. Sotiroudis, S. K. Goudos, and K. Siakavara, “Feature Importances: A Tool to Explain

Radio Propagation and Reduce Model Complexity," *Telecom*, vol. 1, no. 2, pp. 114–125, 2020, doi: 10.3390/telecom1020009.

[18] M. Saarela and S. Jauhiainen, "Comparison of feature importance measures as explanations for classification models," *SN Appl. Sci.*, vol. 3, no. 2, pp. 1–12, Feb. 2021, doi: 10.1007/s42452-021-04148-9.

[19] C. R. de Sá, "Variance-Based Feature Importance in Neural Networks," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, Oct. 2019, vol. 11828 LNAI, pp. 306–315, doi: 10.1007/978-3-030-33778-0_24.

[20] S. L. Salzberg, "Book Review: C4.5: Programs for Machine Learning," *Mach. Learn.*, vol. 16, no. 3, pp. 235–240, 1994, doi: 10.1023/A:1022645310020.

[21] "Data Mining: Practical Machine Learning Tools and Techniques - 3rd Edition." https://www.elsevier.com/books/data-mining-practical-machine-learning-tools-and-techniques/witten/978-0-12-374856-0 (accessed Apr. 29, 2021).

[22] "scikit-learn: machine learning in Python — scikit-learn 0.24.2 documentation." https://scikit-learn.org/stable/ (accessed Apr. 29, 2021).

[23] "1.10. Decision Trees — scikit-learn 0.24.2 documentation." https://scikit-learn.org/stable/modules/tree.html (accessed Apr. 29, 2021).

[24] M. Ali, A. Kazemi, S. Hajian, and N. Kiani, "Quality Control and Classification of Steel Plates Faults Using Data Mining," *Appl. Math. Inf. Sci. Lett. An Int. J.*, vol. 6, no. 2, p. 59, 2018, doi: 10.18576/amisl/060202.

[25] D. Elsner, P. A. Khosroshahi, B. Group, A. D. Maccormack, and R. Lagerström, *Multivariate Unsupervised Machine Learning for Anomaly Detection in Enterprise Applications*. .

[26] Y. Tian, M. Fu, and F. Wu, "Steel plates fault diagnosis on the basis of support vector machines," *Neurocomputing*, vol. 151, no. P1, pp. 296–303, 2015, doi: 10.1016/j.neucom.2014.09.036.

[27] H. Sanz, C. Valim, E. Vegas, J. M. Oller, and F. Reverter, "SVM-RFE: Selection and visualization of the most relevant features through non-linear kernels," *BMC Bioinformatics*, vol. 19, no. 1, p. 432, Nov. 2018, doi: 10.1186/s12859-018-2451-4.

[28] F. Busetti, "Genetic algorithms overview," *URL citeseer. ist. psu. edu/busetti01genetic. html*, pp. 1–13, 2007.

[29] T. Zeugmann *et al.*, "Particle Swarm Optimization," in *Encyclopedia of Machine Learning*, Boston, MA: Springer US, 2011, pp. 760–766.

[30] A. Kharal, "Explainable Artificial Intelligence Based Fault Diagnosis and Insight Harvesting for Steel Plates Manufacturing," *arXiv*, Aug. 2020, Accessed: Apr. 29, 2021. [Online]. Available: http://arxiv.org/abs/2008.04448.

[31] N. V Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "SMOTE: Synthetic Minority Over-sampling Technique," 2002.

[32] T. Pan, J. Zhao, W. Wu, and J. Yang, "Learning imbalanced datasets based on SMOTE and Gaussian distribution," *Inf. Sci. (Ny).*, vol. 512, pp. 1214–1233, Feb. 2020, doi:

10.1016/j.ins.2019.10.048.

[33] “artima - The Making of Python.” https://www.artima.com/articles/the-making-of-python (accessed Apr. 29, 2021).

[34] H. Shen, “Interactive notebooks: Sharing the code,” *Nature*, vol. 515, no. 7525. Nature Publishing Group, pp. 151–152, Nov. 06, 2014, doi: 10.1038/515151a.

[35] “Anaconda | The World’s Most Popular Data Science Platform.” https://www.anaconda.com/ (accessed Apr. 29, 2021).

[36] “NumPy.” https://numpy.org/ (accessed Apr. 29, 2021).

[37] “pandas - Python Data Analysis Library.” https://pandas.pydata.org/ (accessed Apr. 30, 2021).

[38] “1. Supervised learning — scikit-learn 0.24.2 documentation.” https://scikit-learn.org/stable/supervised_learning.html#supervised-learning (accessed Apr. 30, 2021).

[39] “TensorFlow.” https://www.tensorflow.org/ (accessed Apr. 30, 2021).

[40] “Keras: the Python deep learning API.” https://keras.io/ (accessed Apr. 30, 2021).

[41] “PyTorch.” https://pytorch.org/ (accessed Apr. 30, 2021).

[42] “pandas - Explore - Google Trends.” https://trends.google.com/trends/explore?date=today 5-y&geo=US&q=pandas (accessed Apr. 30, 2021).

[43] M. Maalouf, “Logistic regression in data analysis: An overview,” *International Journal of Data Analysis Techniques and Strategies*, vol. 3, no. 3. Inderscience Publishers, pp. 281–299, 2011, doi: 10.1504/IJDATS.2011.041335.

[44] “sklearn.linear_model.LogisticRegression — scikit-learn 0.24.2 documentation.” https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.LogisticRegression.html (accessed Apr. 30, 2021).

[45] A. L. Jousselme and P. Maupin, “Distances in evidence theory: Comprehensive survey and generalizations,” in *International Journal of Approximate Reasoning*, Feb. 2012, vol. 53, no. 2, pp. 118–145, doi: 10.1016/j.ijar.2011.07.006.

[46] G. Guo, H. Wang, D. Bell, Y. Bi, and K. Greer, “KNN Model-Based Approach in Classification.”

[47] “sklearn.neighbors.KNeighborsClassifier — scikit-learn 0.24.2 documentation.” https://scikit-learn.org/stable/modules/generated/sklearn.neighbors.KNeighborsClassifier.html (accessed Apr. 30, 2021).

[48] B. E. Boser, B. E. Boser, I. M. Guyon, and V. N. Vapnik, “A Training Algorithm for Optimal Margin Classifiers,” *Proc. 5TH Annu. ACM Work. Comput. Learn. THEORY*, pp. 144--152, 1992, Accessed: Apr. 30, 2021. [Online]. Available: https://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.21.3818.

[49] M. Onel, C. A. Kieslich, Y. A. Guzman, and E. N. Pistikopoulos, “Simultaneous Fault Detection and Identification in Continuous Processes via nonlinear Support Vector Machine

based Feature Selection," in *Computer Aided Chemical Engineering*, vol. 44, Elsevier B.V., 2018, pp. 2077–2082.

[50] "1.4. Support Vector Machines — scikit-learn 0.24.2 documentation." https://scikit-learn.org/stable/modules/svm.html (accessed Apr. 30, 2021).

[51] L. Breiman, "Random forests," *Mach. Learn.*, vol. 45, no. 1, pp. 5–32, Oct. 2001, doi: 10.1023/A:1010933404324.

[52] G. Biau and G. B. Fr, "Analysis of a Random Forests Model," 2012.

[53] "1.11. Ensemble methods — scikit-learn 0.24.2 documentation." https://scikit-learn.org/stable/modules/ensemble.html (accessed Apr. 30, 2021).

[54] J. H. Friedman, "Greedy function approximation: A gradient boosting machine.," *Ann. Stat.*, vol. 29, no. 5, pp. 1189–1232, Oct. 2001, doi: 10.1214/aos/1013203451.

[55] "sklearn.ensemble.GradientBoostingClassifier — scikit-learn 0.24.2 documentation." https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.GradientBoostingClassifier.html (accessed Apr. 30, 2021).

[56] "sklearn.model_selection.GridSearchCV — scikit-learn 0.24.2 documentation." https://scikit-learn.org/stable/modules/generated/sklearn.model_selection.GridSearchCV.html (accessed Apr. 30, 2021).

[57] J. Bergstra, J. B. Ca, and Y. B. Ca, "Random Search for Hyper-Parameter Optimization Yoshua Bengio," 2012. Accessed: Apr. 30, 2021. [Online]. Available: http://scikit-learn.sourceforge.net.

[58] H. C. Wen *et al.*, "One-Hot Encoding Functional verification FPGA vs . ASIC Designs Answers to Exercises Artificial intelligence methods for mu- sic generation : a review and future per- spectives," 2018.

[59] "Pearson's Correlation Coefficient," in *Encyclopedia of Public Health*, Dordrecht: Springer Netherlands, 2008, pp. 1090–1091.

[60] T. Höfer, H. Przyrembel, and S. Verleger, "New evidence for the Theory of the Stork," *Paediatr. Perinat. Epidemiol.*, vol. 18, no. 1, pp. 88–92, Jan. 2004, doi: 10.1111/j.1365-3016.2003.00534.x.

[61] B. H. Menze *et al.*, "A comparison of random forest and its Gini importance with standard chemometric methods for the feature selection and classification of spectral data," *BMC Bioinformatics*, vol. 10, Jul. 2009, doi: 10.1186/1471-2105-10-213.

[62] S. Loazia, "Gini Impurity Measure— An intuitive explanation using python," p. 605, 2020, [Online]. Available: https://towardsdatascience.com/gini-impurity-measure-dbd3878ead33.

[63] E. Tuv, A. Borisov, G. Runger, and K. Torkkola, "Feature selection with ensembles, artificial variables, and redundancy elimination," *J. Mach. Learn. Res.*, vol. 10, pp. 1341–1366, 2009, Accessed: Apr. 30, 2021. [Online]. Available: https://asu.pure.elsevier.com/en/publications/feature-selection-with-ensembles-artificial-variables-and-redunda.

[64] Y. Xu, G. Jones, J. Li, B. Wang, and C. Sun, "A Study on Mutual Information-based Feature

Selection for Text Categorization."

[65] E. Keogh and A. Mueen, "Curse of Dimensionality," in *Encyclopedia of Machine Learning and Data Mining*, Springer US, 2017, pp. 314–315.

[66] "Introduction," in *Principal Component Analysis*, Springer-Verlag, 2006, pp. 1–9.

[67] L. Van Der Maaten and G. Hinton, "Visualizing Data using t-SNE," 2008.

[68] "sklearn.manifold.TSNE — scikit-learn 0.24.2 documentation." https://scikit-learn.org/stable/modules/generated/sklearn.manifold.TSNE.html (accessed Apr. 30, 2021).

[69] "Feature Engineering and Selection: A Practical Approach for Predictive Models." http://www.feat.engineering/ (accessed Apr. 30, 2021).

[70] K. Thurnhofer-Hemsi, E. López-Rubio, M. A. Molina-Cabello, and K. Najarian, "Radial basis function kernel optimization for Support Vector Machine classifiers," Jul. 2020, Accessed: Apr. 30, 2021. [Online]. Available: http://arxiv.org/abs/2007.08233.

[71] "sklearn.metrics.pairwise.rbf_kernel — scikit-learn 0.24.2 documentation." https://scikit-learn.org/stable/modules/generated/sklearn.metrics.pairwise.rbf_kernel.html (accessed Apr. 30, 2021).

[72] R. Blagus and L. Lusa, "SMOTE for high-dimensional class-imbalanced data," *BMC Bioinformatics*, vol. 14, no. 1, p. 106, Mar. 2013, doi: 10.1186/1471-2105-14-106.

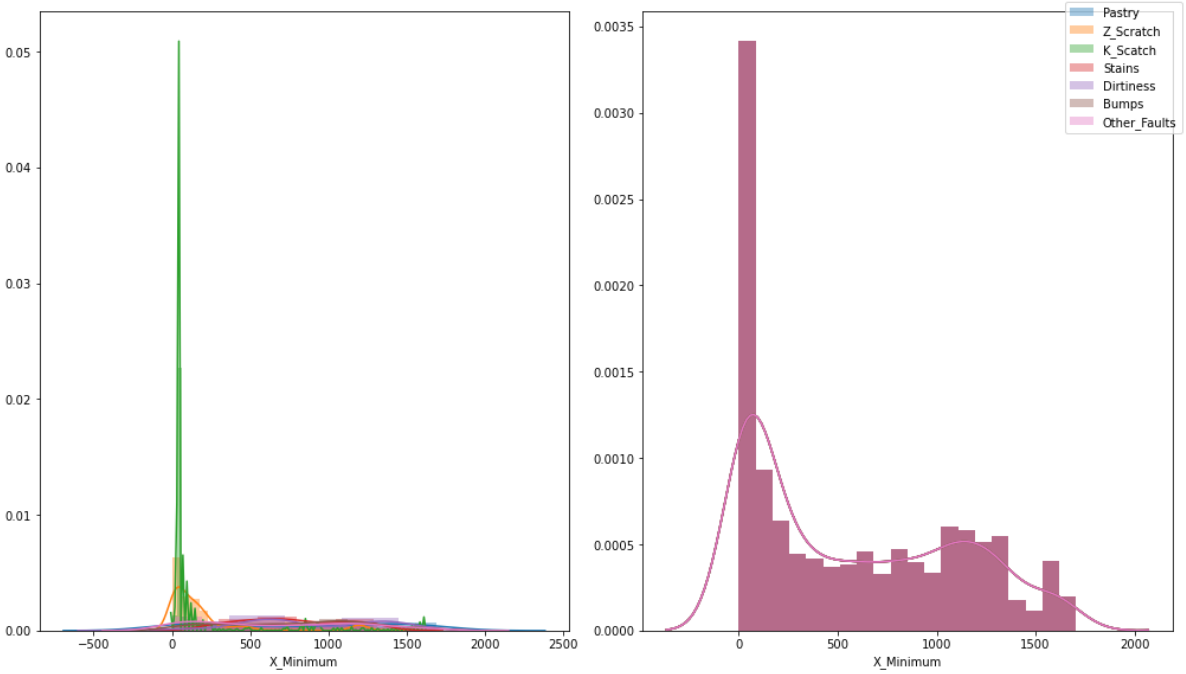
[73] H. He, Y. Bai, E. A. Garcia, and S. Li, "ADASYN: Adaptive synthetic sampling approach for imbalanced learning," in *Proceedings of the International Joint Conference on Neural Networks*, 2008, pp. 1322–1328, doi: 10.1109/IJCNN.2008.4633969.

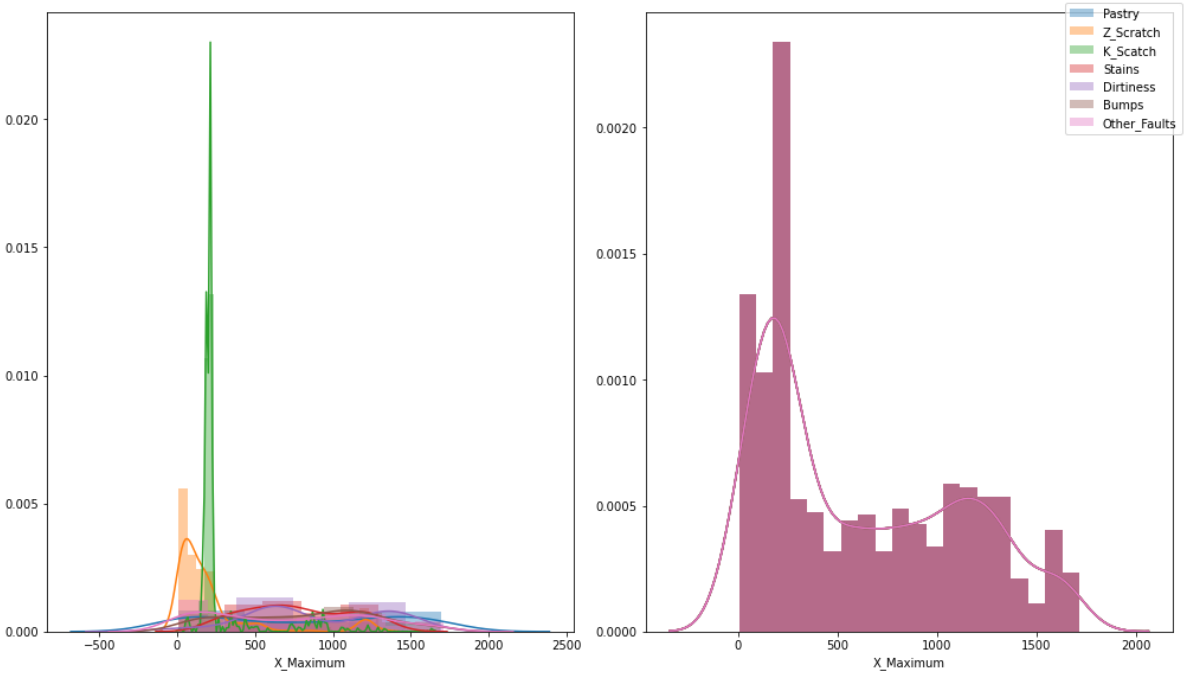
[74] "imblearn.over_sampling.ADASYN — imbalanced-learn 0.3.0.dev0 documentation." http://glemaitre.github.io/imbalanced-learn/generated/imblearn.over_sampling.ADASYN.html (accessed Apr. 30, 2021).

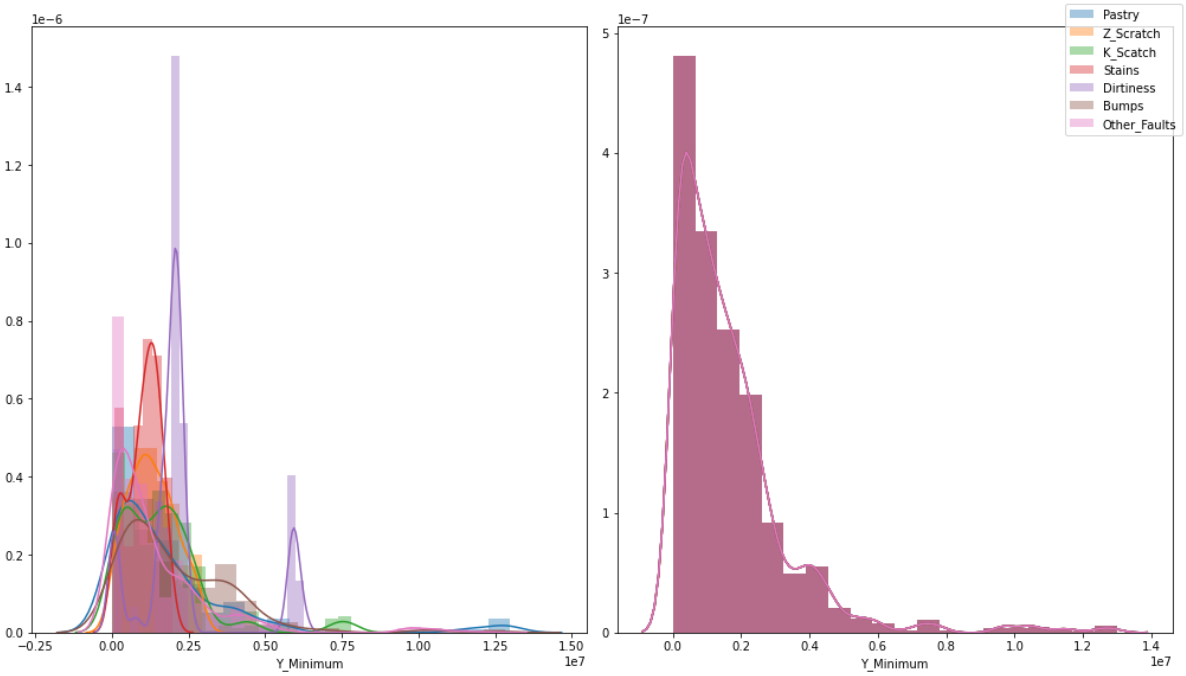
ΠΑΡΑΡΤΗΜΑ 1. ΓΡΑΜΜΙΚΗ ΣΥΣΧΕΤΙΣΗ ΧΑΡΑΚΤΗΡΙΣΤΙΚΩΝ

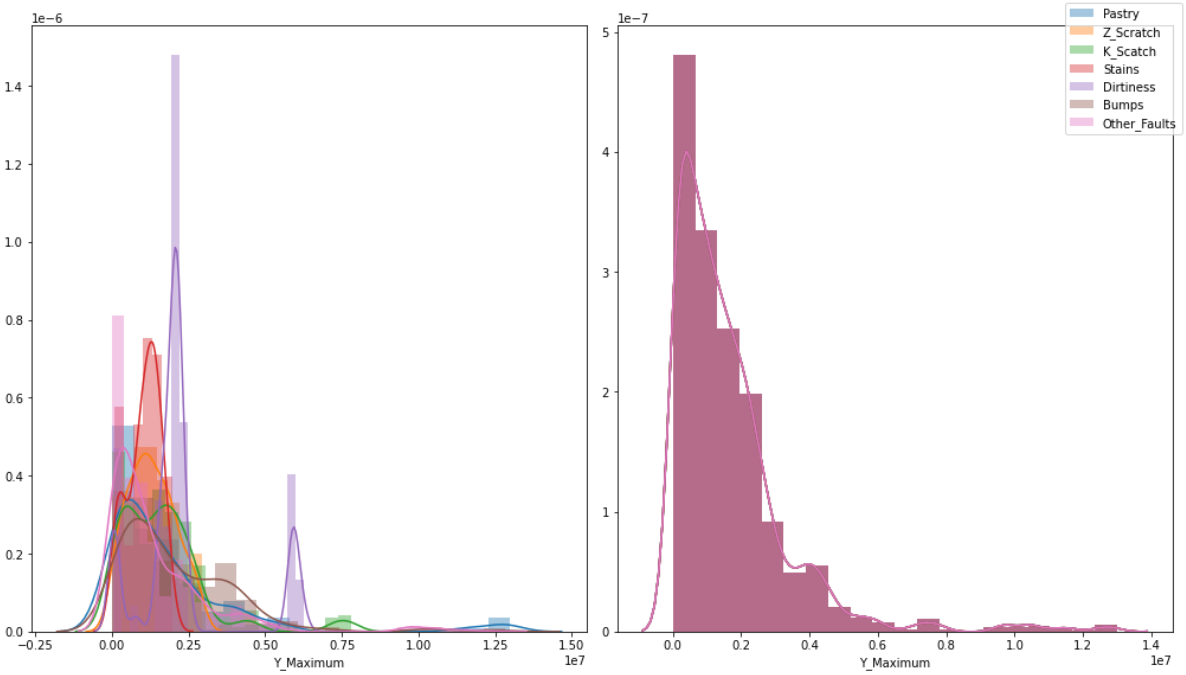
	X_Minimum	X_Maximum	Y_Minimum	Y_Maximum	Pixels_Areas	X_Perimeter	Y_Perimeter	Sum_of_Luminosity	Minimum_of_Luminosity	Maximum_of_Luminosity	Length_of_Conveyer	TypeOfSteel_A300	TypeOfSteel_A400	Steel_Plate_Thickness	Edges_Index	Empty_Index	Square_Index	Outside_X_Index	Edges_X_Index	Edges_Y_Index	Outside_Global_Index	LogOfAreas	Log_X_Index	Log_Y_Index	Orientation_Index	Luminosity_Index	SigmoidOfAreas
X_Minimum	1	0.9883135	0.0418213	0.0418065	-0.30732176	-0.2589374	-0.118757368	-0.339045394	0.237637154	-0.075554349	0.316662417	0.144319482	-0.144319482	0.136625071	0.2780749	-0.1984608	0.0636579	-0.361159508	0.154778153	0.367907412	0.147281761	-0.4285529	-0.4379439	-0.3268505	0.178585372	-0.031577766	-0.355251295
X_Maximum	0.9883135	1	0.0521473	0.0521352	-0.22539928	-0.1863261	-0.090137677	-0.247052068	0.168649497	-0.062391556	0.299390273	0.112008867	-0.112008867	0.106118853	0.2428463	-0.1526802	0.048575	-0.214930402	0.149258695	0.2719148	0.099252892	-0.3321691	-0.3240118	-0.2659901	0.115019392	-0.038996153	-0.286735946
Y_Minimum	0.0418213	0.0521473	1	1	0.017669766	0.0238432	0.024150084	0.007362208	-0.065702585	-0.067785101	-0.049210774	0.07516363	-0.07516363	-0.207640422	0.0213142	-0.0431171	-0.0061349	0.054164575	0.06608496	-0.036543194	-0.062911339	0.0449518	0.0704063	-0.008442	-0.086496905	-0.090653533	0.025257091
Y_Maximum	0.0418065	0.0521352	1	1	0.017840349	0.0240381	0.0243804	0.00749885	-0.065733089	-0.067775823	-0.04921918	0.075150658	-0.075150658	-0.207644246	0.0212996	-0.0430853	-0.0061519	0.054184891	0.066051086	-0.036549142	-0.062901483	0.0449936	0.0704318	-0.0083818	-0.08647971	-0.090665823	0.025283891
Pixels_Areas	-0.3073218	-0.2253993	0.0176698	0.0178403	1	0.9666441	0.827199212	0.978951617	-0.497204417	0.110062541	-0.155852584	-0.235591307	0.235591307	-0.183735086	-0.2752885	0.2728082	0.0178655	0.588605799	-0.294673005	-0.463571447	-0.109654571	0.6502339	0.6030719	0.5783425	-0.137603658	-0.043448768	0.422947493
X_Perimeter	-0.2589374	-0.1863261	0.0238432	0.0240381	0.96664407	1	0.912436459	0.912955837	-0.400427368	0.111363038	-0.134239989	-0.18925007	0.18925007	-0.147711687	-0.2275899	0.3063477	0.0045069	0.51709842	-0.293038692	-0.412099511	-0.079105547	0.5630359	0.5247155	0.5234715	-0.101731103	-0.032616544	0.380604947
Y_Perimeter	-0.1187574	-0.0901377	0.0241501	0.0243804	0.827199212	0.9124365	1	0.704875735	-0.213758225	0.061808674	-0.063824645	-0.09515373	0.09515373	-0.058888803	-0.1112399	0.1888247	-0.0475114	0.209160155	-0.195162304	-0.136722659	0.013438174	0.2940398	0.2284848	0.3443777	0.031381341	-0.047777525	0.191771984
Sum_of_Luminosity	-0.3390454	-0.2470521	0.0073622	0.0074989	0.978951617	0.9129558	0.704875735	1	-0.540565938	0.136515251	-0.169330746	-0.263631925	0.263631925	-0.204811585	-0.3014525	0.293691	0.049607	0.658338913	-0.327728351	-0.529745402	-0.121089808	0.7121281	0.6677365	0.6187951	-0.158483002	-0.014066927	0.464248328
Minimum_of_Luminosity	0.2376372	0.1686495	-0.0657026	-0.0657331	-0.49720442	-0.4004274	-0.213758225	-0.540565938	1	0.429604577	-0.023579318	0.04204828	-0.04204828	0.103392815	0.3589154	-0.0441113	0.0667477	-0.487573835	0.252255803	0.316609765	0.035461953	-0.6787617	-0.5676545	-0.5882075	0.057122531	0.669533902	-0.514796538
Maximum_of_Luminosity	-0.0755543	-0.0623916	-0.0677851	-0.0677758	0.110062541	0.111363	0.061808674	0.136515251	0.429604577	1	-0.098008853	-0.216338719	0.216338719	-0.128396509	0.1496747	0.0314253	0.0655175	0.099299864	0.093522019	-0.167440803	-0.124038557	0.007672	0.0928228	-0.0695223	-0.169747359	0.870160232	-0.039651285
Length_of_Conveyer	0.3166624	0.2993903	-0.0492108	-0.0492192	-0.15585258	-0.13424	-0.063824645	-0.169330746	-0.023579318	-0.098008853	1	0.378542494	-0.378542494	0.214768876	0.1351522	-0.2306007	0.0736941	-0.21741688	0.123585186	0.235731966	0.128663426	-0.1932472	-0.2199731	-0.1570571	0.120715474	-0.149768537	-0.197542564
TypeOfSteel_A300	0.1443195	0.1120089	0.0751636	0.0751507	-0.23559131	-0.1892501	-0.09515373	-0.263631925	0.04204828	-0.216338719	0.378542494	1	-1	0.12564974	0.1121398	-0.0919535	0.1641562	-0.244765482	0.17383618	0.240633665	0.022142259	-0.3296143	-0.2669549	-0.3117962	0.010629992	-0.252818417	-0.308909632
TypeOfSteel_A400	-0.1443195	-0.1120089	-0.0751636	-0.0751507	0.235591307	0.1892501	0.09515373	0.263631925	-0.04204828	0.216338719	-0.378542494	-1	1	-0.12564974	-0.1121398	0.0919535	-0.1641562	0.244765482	-0.17383618	-0.240633665	-0.022142259	0.3296143	0.2669549	0.3117962	-0.010629992	0.252818417	0.308909632
Steel_Plate_Thickness	0.1366251	0.1061189	-0.2076404	-0.2076442	-0.18373509	-0.1477117	-0.058888803	-0.204811585	0.103392815	-0.128396509	0.214768876	0.125649074	-0.125649074	1	0.0634488	0.0125257	-0.1243819	-0.228351623	-0.077408379	0.251984915	0.221243648	-0.1766391	-0.2528219	-0.0372875	0.274096866	-0.116499088	-0.085159159
Edges_Index	0.2780749	0.2428463	0.0213142	0.0212996	-0.2752885	-0.2275899	-0.111239929	-0.30145247	0.358915386	0.149674689	0.135152169	0.112139815	-0.112139815	0.063448805	1	-0.1807393	0.1494981	-0.296509641	0.250178456	0.285302472	0.008282369	-0.4086189	-0.3558526	-0.3719889	0.020547582	0.207515945	-0.330005805
Empty_Index	-0.1984608	-0.1526802	-0.0431171	-0.0430853	0.272808218	0.3063477	0.188824662	0.293690984	-0.044111314	0.031425274	-0.230600694	-0.091953523	0.091953523	0.012525676	-0.1807393	1	-0.0764389	0.334995803	-0.389341933	-0.459800102	-0.165292724	0.356685	0.4488643	0.3972888	-0.139419641	0.061608417	0.481737839
Square_Index	0.0636579	0.048575	-0.0061349	-0.0061519	0.017865472	0.0045069	-0.04751137	0.049606974	0.066747749	0.065517495	0.073694071	0.164156182	-0.164156182	-0.124381925	0.1494981	-0.0764389	1	-0.113626958	0.242779472	0.081488486	-0.069912564	-0.1893405	-0.0828464	-0.257661	-0.162034129	0.111976621	-0.292251454
Outside_X_Index	-0.3611595	-0.2149304	0.0541646	0.0541849	0.588605799	0.5170984	0.209160155	0.658338913	-0.487573835	0.099299864	-0.21741688	-0.244765482	0.244765482	-0.228351623	-0.2965096	0.3349958	-0.113627	1	-0.076663052	-0.68986746	-0.33717317	0.710837	0.8202233	0.4648598	-0.440357747	-0.035721241	0.518909685
Edges_X_Index	0.1547782	0.1492587	0.066085	0.0660511	-0.29467301	-0.2930387	-0.195162304	-0.327728351	0.252255803	0.093522019	0.123585186	0.17383618	-0.17383618	-0.077408379	0.2501785	-0.3893419	0.2427795	-0.076663052	1	0.108143852	-0.419383343	-0.4962062	-0.189262	-0.748892	-0.550301556	0.126460157	-0.558426195
Edges_Y_Index	0.3679074	0.2719148	-0.0365432	-0.0365491	-0.46357145	-0.4120995	-0.136722659	-0.529745402	0.316609765	-0.167440803	0.235731966	0.240633665	-0.240633665	0.251984915	0.2853025	-0.4598001	0.0814885	-0.68986746	0.108143852	1	0.537565154	-0.6429913	-0.8554137	-0.3218921	0.658049389	-0.094367668	-0.545393159
Outside_Global_Index	0.1472818	0.0992529	-0.0629113	-0.0629015	-0.10965457	-0.0791055	0.013438174	-0.121089808	0.035461953	-0.124038557	0.128663426	0.022142259	-0.022142259	0.221243648	0.0082824	-0.1652927	-0.0699126	-0.33717317	-0.419383343	0.537565154	1	-0.0977616	-0.4280599	0.2418981	0.8626697	-0.122321303	-0.053770347
LogOfAreas	-0.4285529	-0.3321691	0.0449518	0.0449936	0.650233889	0.5630359	0.294039794	0.712128115	-0.678761689	0.007671961	-0.19324724	-0.329614299	0.329614299	-0.176639075	-0.4086189	0.356685	-0.1893405	0.710836981	-0.496206249	-0.642991336	-0.097761619	1	0.8889194	0.8829741	-0.123898299	-0.175878707	0.877768308
Log_X_Index	-0.4379439	-0.3240118	0.0704063	0.0704318	0.603071866	0.5247155	0.228484845	0.667736457	-0.567654515	0.092822767	-0.219973111	-0.266954876	0.266954876	-0.252821872	-0.3558526	0.4488643	-0.0828464	0.820223254	-0.189261969	-0.855413748	-0.428059867	0.8889194	1	0.5986516	-0.536628718	-0.064923128	0.757342648
Log_Y_Index	-0.3268505	-0.2659901	-0.008442	-0.0083818	0.578342453	0.5234715	0.344377686	0.618795075	-0.58820752	-0.069522316	-0.157057147	-0.311796195	0.311796195	-0.037287459	-0.3719889	0.3972888	-0.257661	0.464859766	-0.748892043	-0.321892091	0.241898064	0.8829741	0.5986516	1	0.316792479	-0.21910951	0.838187844
Orientation_Index	0.1785854	0.1150194	-0.0864969	-0.0864797	-0.13760366	-0.1017311	0.031381341	-0.158483002	0.057122531	-0.169747359	0.120715474	0.010629992	-0.010629992	0.274096866	0.0205476	-0.1394196	-0.1620341	-0.440357747	-0.550301556	0.658049389	0.8626697	-0.1238983	-0.5366287	0.3167925	1	-0.153464188	-0.023978025
Luminosity_Index	-0.0315778	-0.0389962	-0.0906535	-0.0906658	-0.04344877	-0.0326165	-0.047777525	-0.014066927	0.669533902	0.870160232	-0.149768537	-0.252818417	0.252818417	-0.116499088	0.2075159	0.0616084	0.1119766	-0.035721241	0.126460157	-0.094367668	-0.122321303	-0.1758787	-0.0649231	-0.2191095	-0.153464188	1	-0.184840346
SigmoidOfAreas	-0.3552513	-0.2867359	0.0252571	0.0252839	0.422947493	0.3806049	0.191771984	0.464248328	-0.514796538	-0.039651285	-0.197542564	-0.308909632	0.308909632	-0.085159159	-0.3300058	0.4817378	-0.2922515	0.518909685	-0.558426195	-0.545393159	-0.053770347	0.8777683	0.7573426	0.8381878	-0.023978025	-0.184840346	1
Pastry	0.1349557	0.1198139	0.0364881	0.0364884	-0.07675241	-0.075418	-0.01761618	-0.084306591	-0.0746971	-0.058742303	0.196209466	-0.054791584	0.054791584	0.076579117	-0.0299152	-0.1849935	-0.2429225	-0.132080632	-0.12242779	0.229431699	0.244320463	-0.0425257	-0.1982502	0.0911302	0.329384814	-0.108018446	-0.017422109
Z_Scratch	-0.2289595	-0.2581781	-0.0633268	-0.0633287	-0.08844021	-0.0605818	-0.025721009	-0.099592485	0.049905415	-0.189441057	-0.230855819	0.339487754	-0.339487754	-0.02487562	-0.1527301	0.1296719	-0.0055427	-0.109354443	-0.141291301	0.088045396	0.084885463	-0.0755645	-0.0944257	-0.0031697	0.11750133	-0.131622884	-0.019396798
K_Scatch	-0.4192635	-0.3360841	-0.0004204	-0.000397	0.556846091	0.4550026	0.203062982	0.616950055	-0.461000081	0.18589743	-0.261070765	-0.407730287	0.407730287	-0.351654138	-0.3377012	0.1567114	0.0591748	0.63136963	-0.121540903	-0.61434095	-0.279992327	0.7045312	0.7358602	0.4818525	-0.384695592	0.098458996	0.488878426
Stains	0.0737397	0.0614715	-0.0666011	-0.0666061	-0.07118248	-0.0675466	-0.035743253	-0.078111362	0.183326679	0.090455953	-0.136838669	-0.154796091	0.154796091	-0.099945083	0.1657323	-0.0102433	0.0973096	-0.093722275	0.229406486	0.096336426	-0.177802172	-0.3126898	-0.2123747	-0.321185	-0.136712909	0.154318959	-0.252488398
Dirtiness	0.103924	0.0965233	0.0642623	0.0642621	-0.05057812	-0.0378203	-0.01005756	-0.055271571	0.092764544	0.022156766	-0.052603369	-0.08248924	0.08248924	0.017726888	0.1035169	-0.0125142	-0.2143686	-0.074104521	-0.180587896	0.096861965	0.105173108	-0.0343447	-0.1205606	0.0623155	0.174628448	0.045294847	-0.000462261
Bumps	0.2212955	0.2017035	0.1261212	0.1261104	-0.16373854	-0.1401969	-0.07098861	-0.179830793	0.078690254	-0.047549466	0.225504422	0.306384545	-0.306384545	-0.016773442	0.2319998	-0.1721467	0.2613853	-0.185729266	0.185481255	0.22923069	0.044963889	-0.22649	-0.1986897	-0.2218126	0.013614095	-0.064744482	-0.220090525
Other_Faults	0.1648036	0.1457835	-0.0844147	-0.0844221	-0.18463179	-0.1429031	-0.066800915	-0.205889612	0.228111848	-0.007784208	0.132090596	-0.007529666	0.007529666	0.31567113	0.097698	0.0482671	-0.093296	-0.166709488	0.074921407	0.063717579	0.038231256	-0.1931207	-0.1517199	-0.1617686	0.043389043	0.039327607	-0.102046163

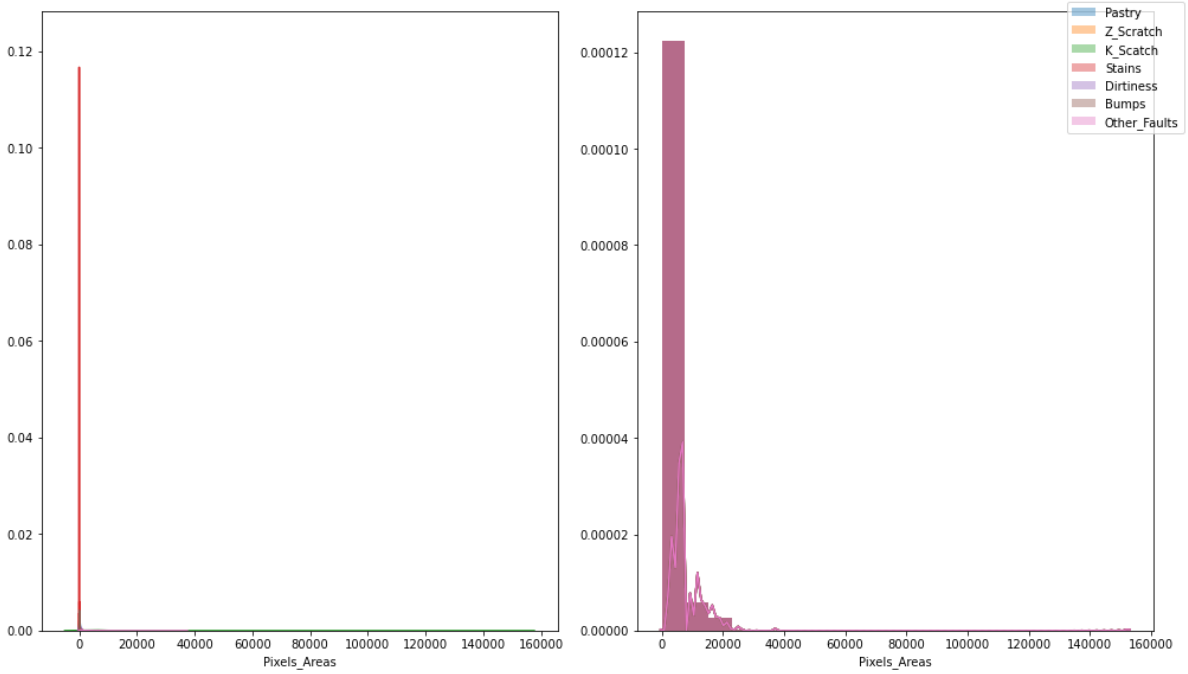
ΠΑΡΑΡΤΗΜΑ 2. ΚΑΤΑΝΟΜΕΣ ΔΕΔΟΜΕΝΩΝ

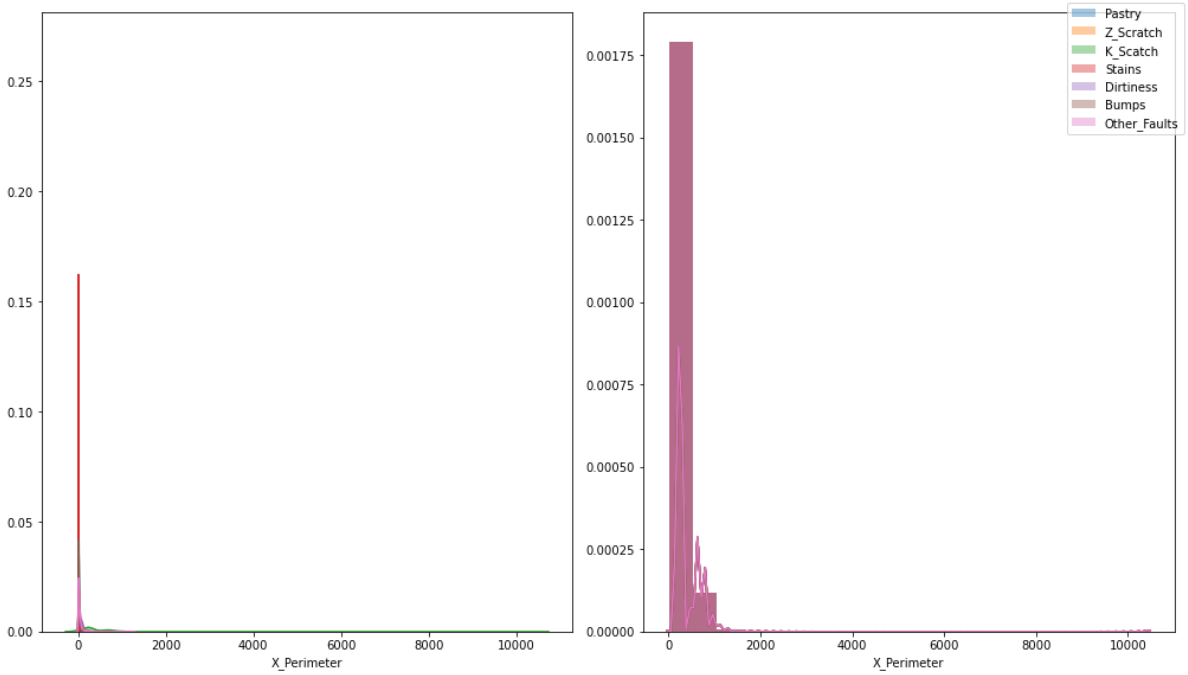


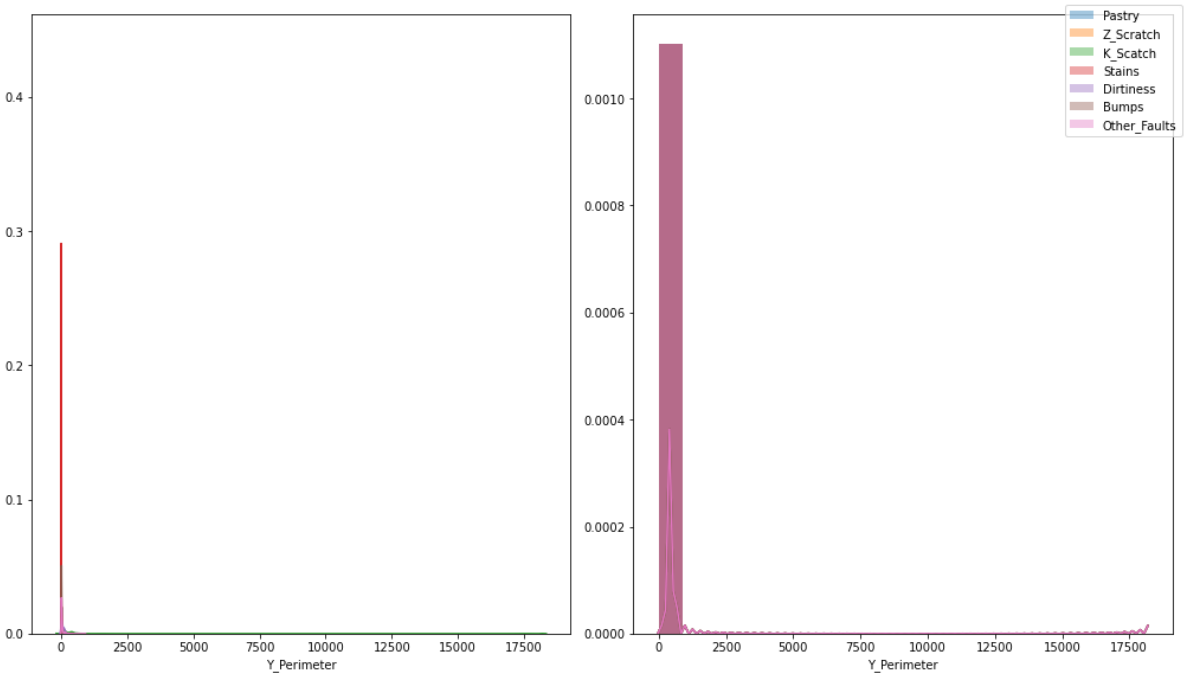


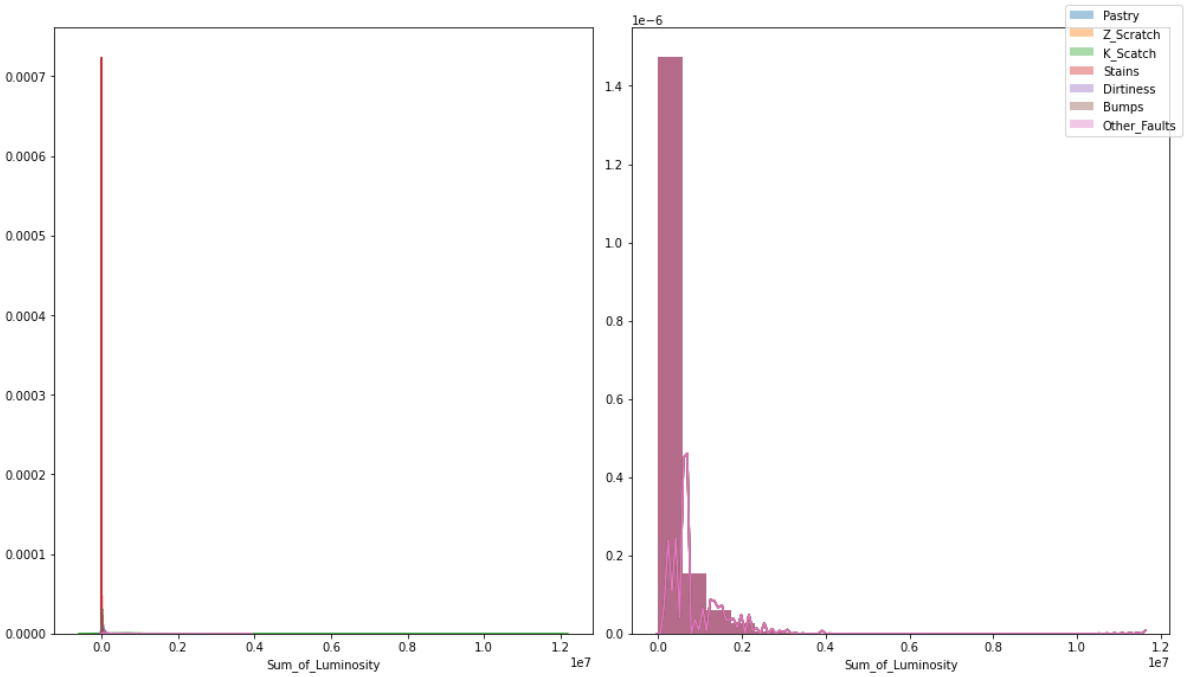

1e−6
1e−7
Pastry
Z_Scratch
K_Scatch
Stains
Dirtiness
Bumps
Other_Faults
Y_Minimum
1e7

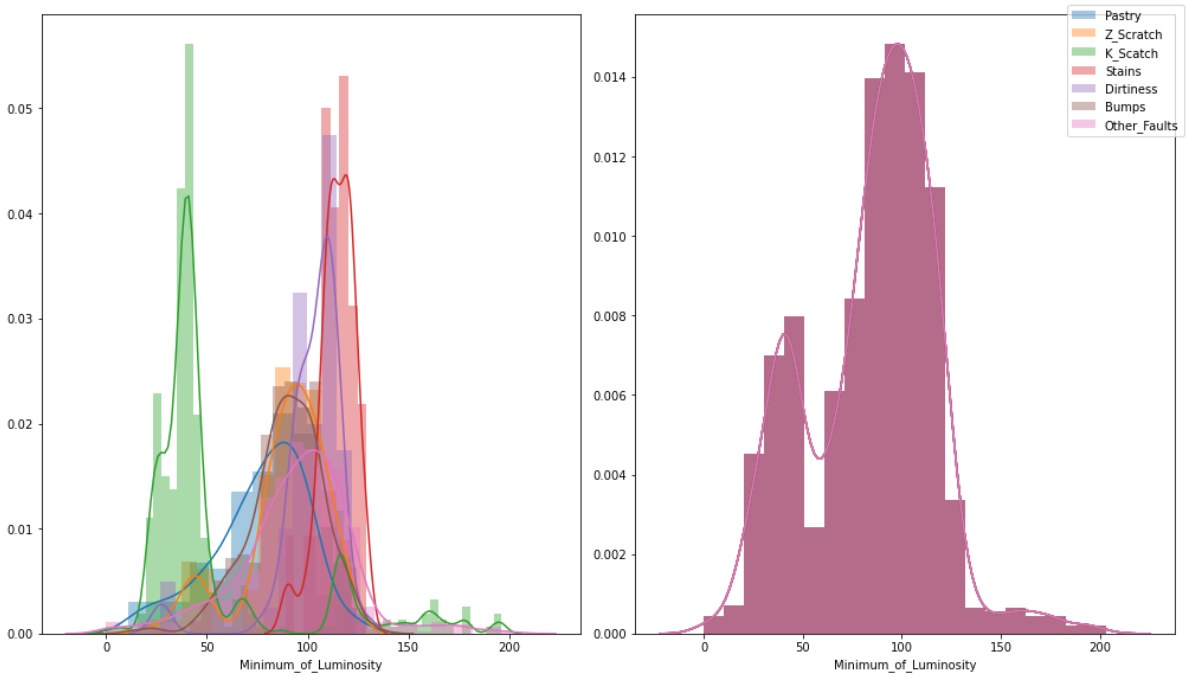

1e−6
1e−7
Pastry
Z_Scratch
K_Scatch
Stains
Dirtiness
Bumps
Other_Faults
Y_Maximum
1e7

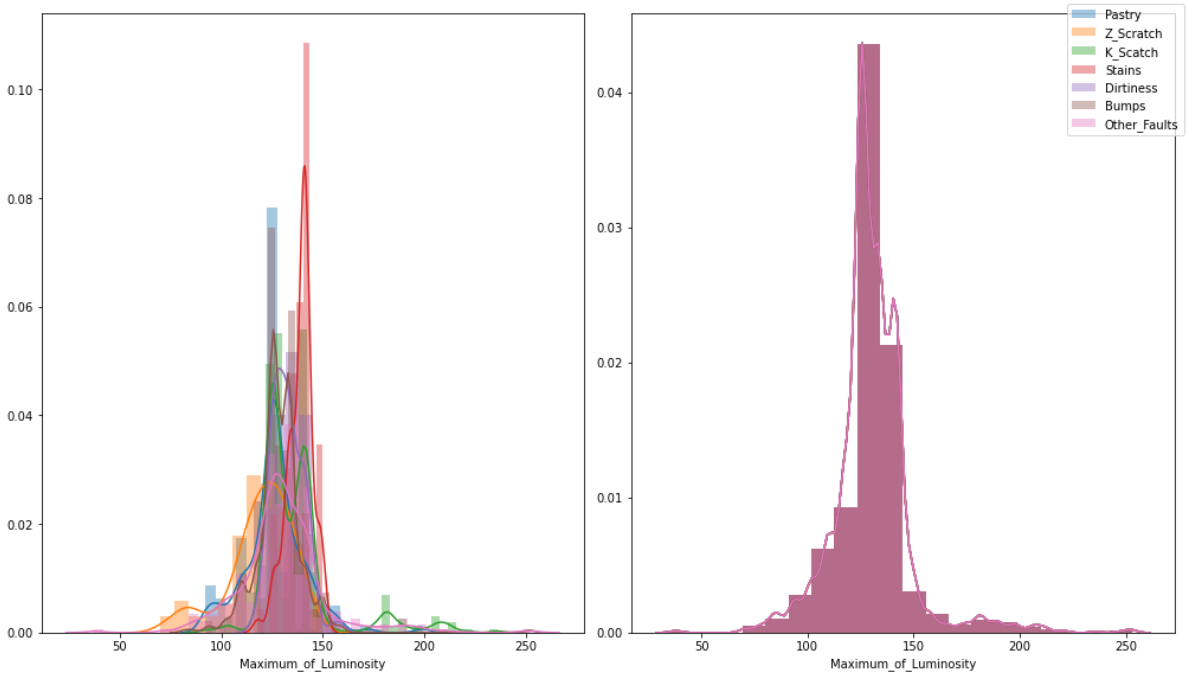

Pastry
Z_Scratch
K_Scatch
Stains
Dirtiness
Bumps
Other_Faults
Pixels_Areas
Pixels_Areas

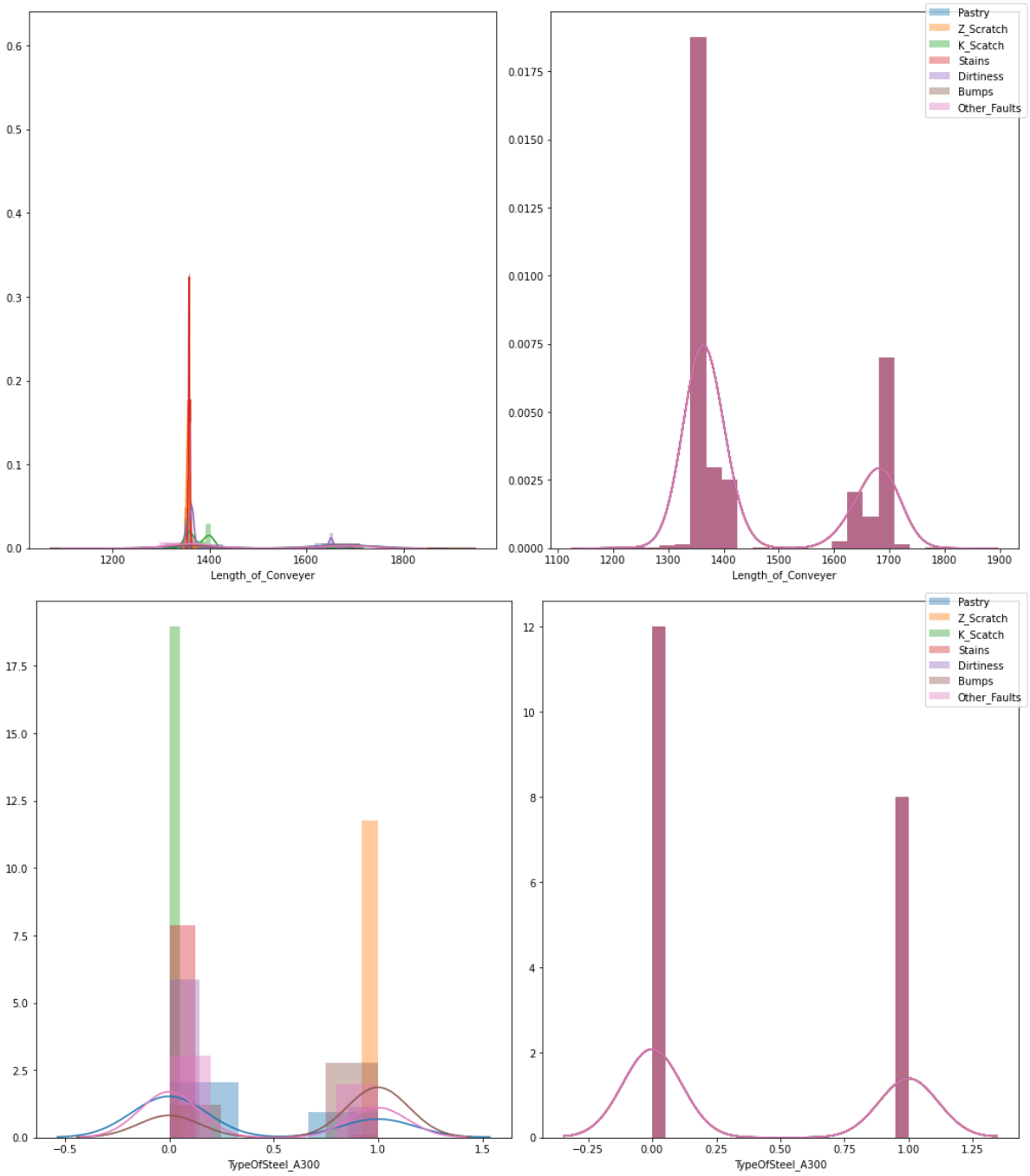

Pastry
Z_Scratch
K_Scatch
Stains
Dirtiness
Bumps
Other_Faults
X_Perimeter
X_Perimeter

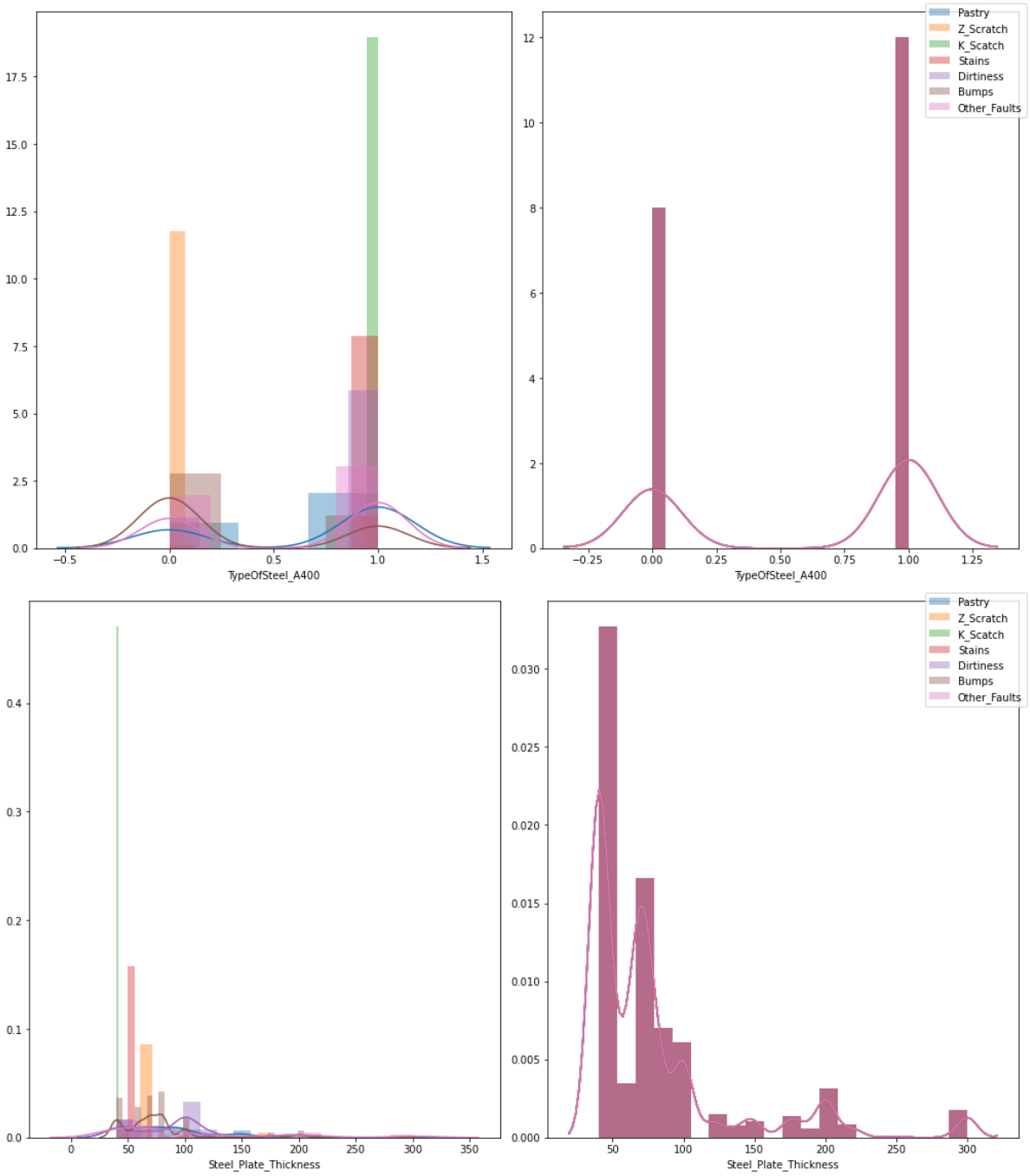

Pastry
Z_Scratch
K_Scatch
Stains
Dirtiness
Bumps
Other_Faults
Y_Perimeter
Y_Perimeter

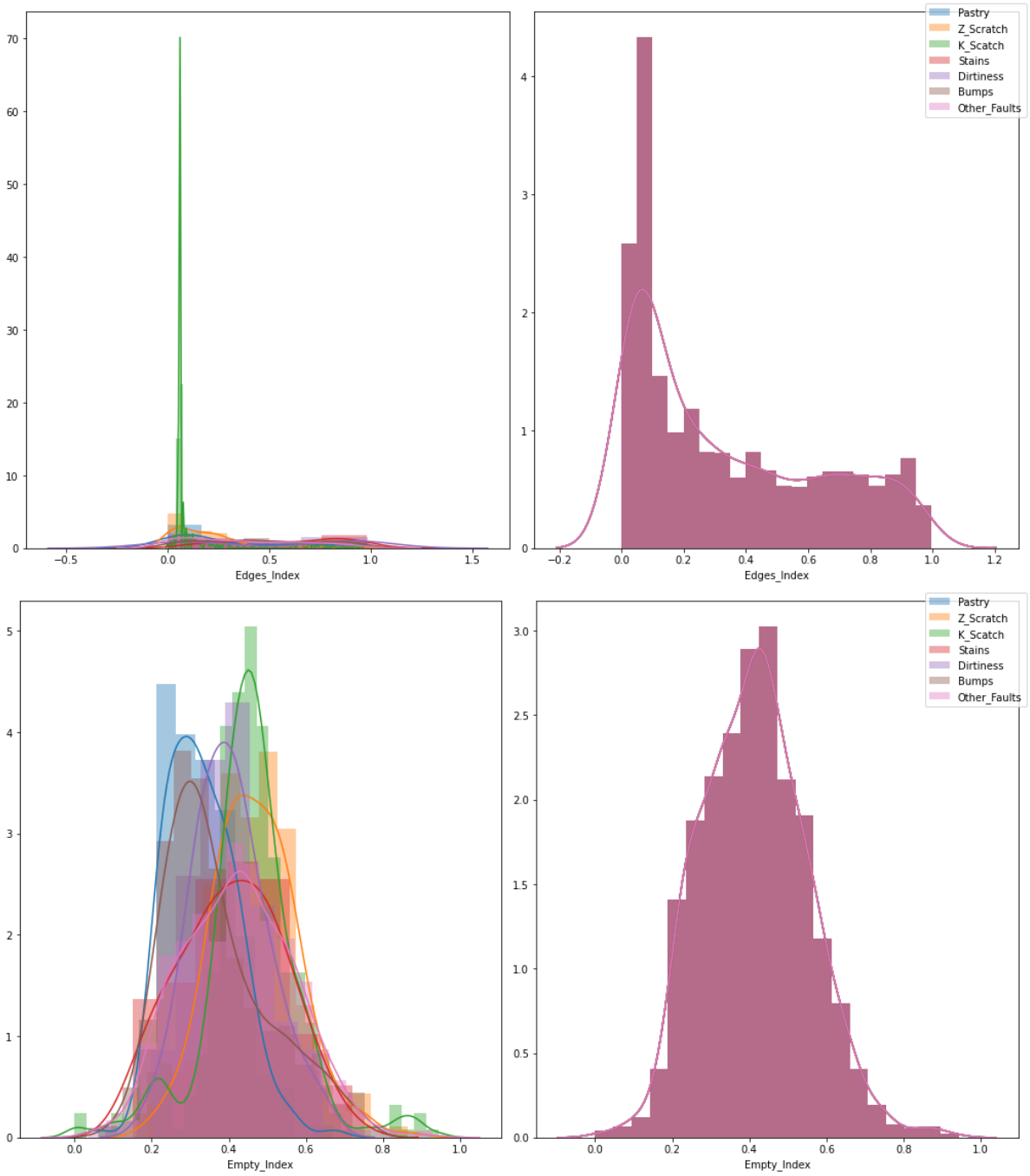

Pastry
Z_Scratch
K_Scatch
Stains
Dirtiness
Bumps
Other_Faults
1e−6
Sum_of_Luminosity
1e7
Sum_of_Luminosity
1e7

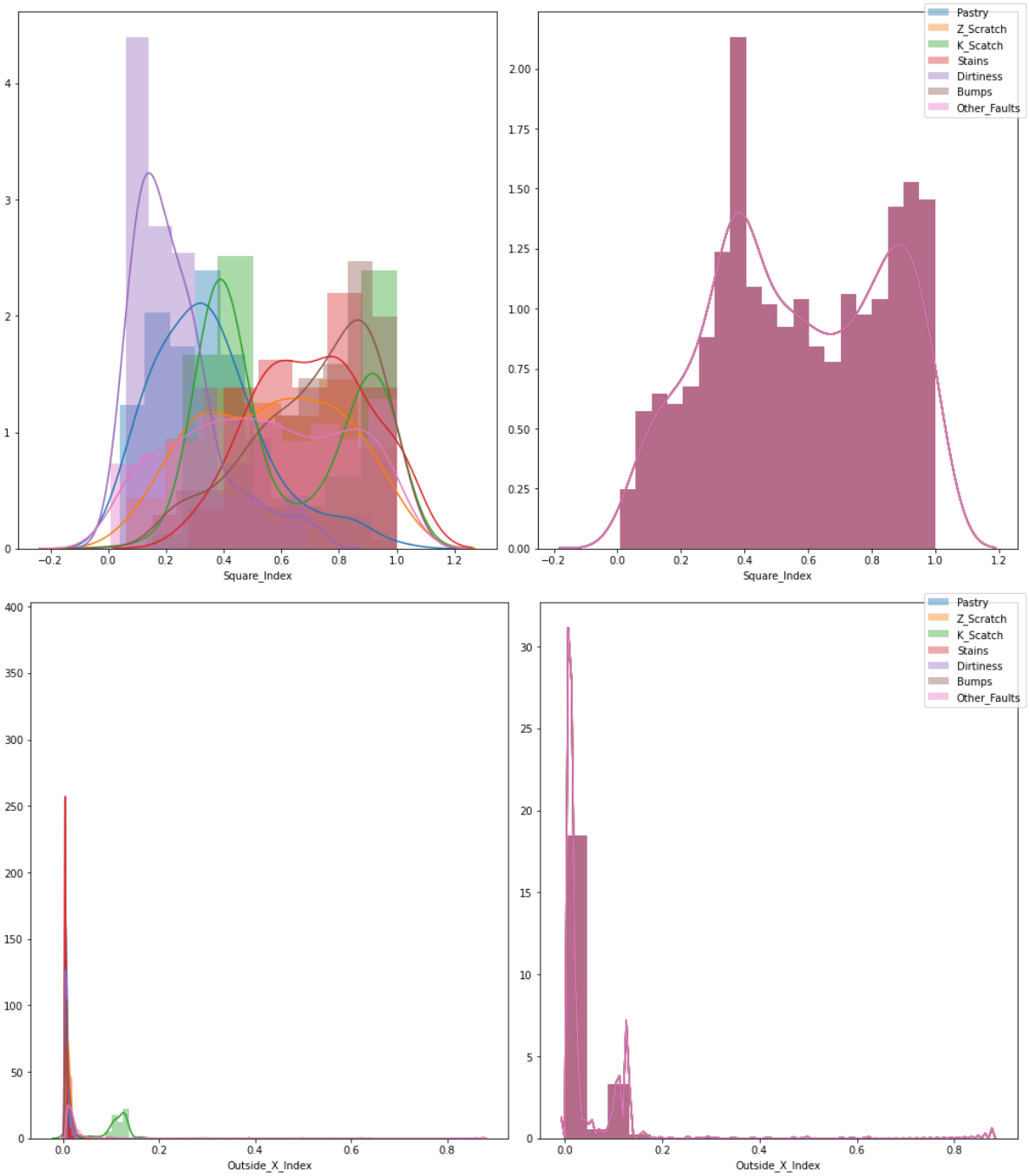

Pastry
Z_Scratch
K_Scatch
Stains
Dirtiness
Bumps
Other_Faults
Minimum_of_Luminosity
Minimum_of_Luminosity

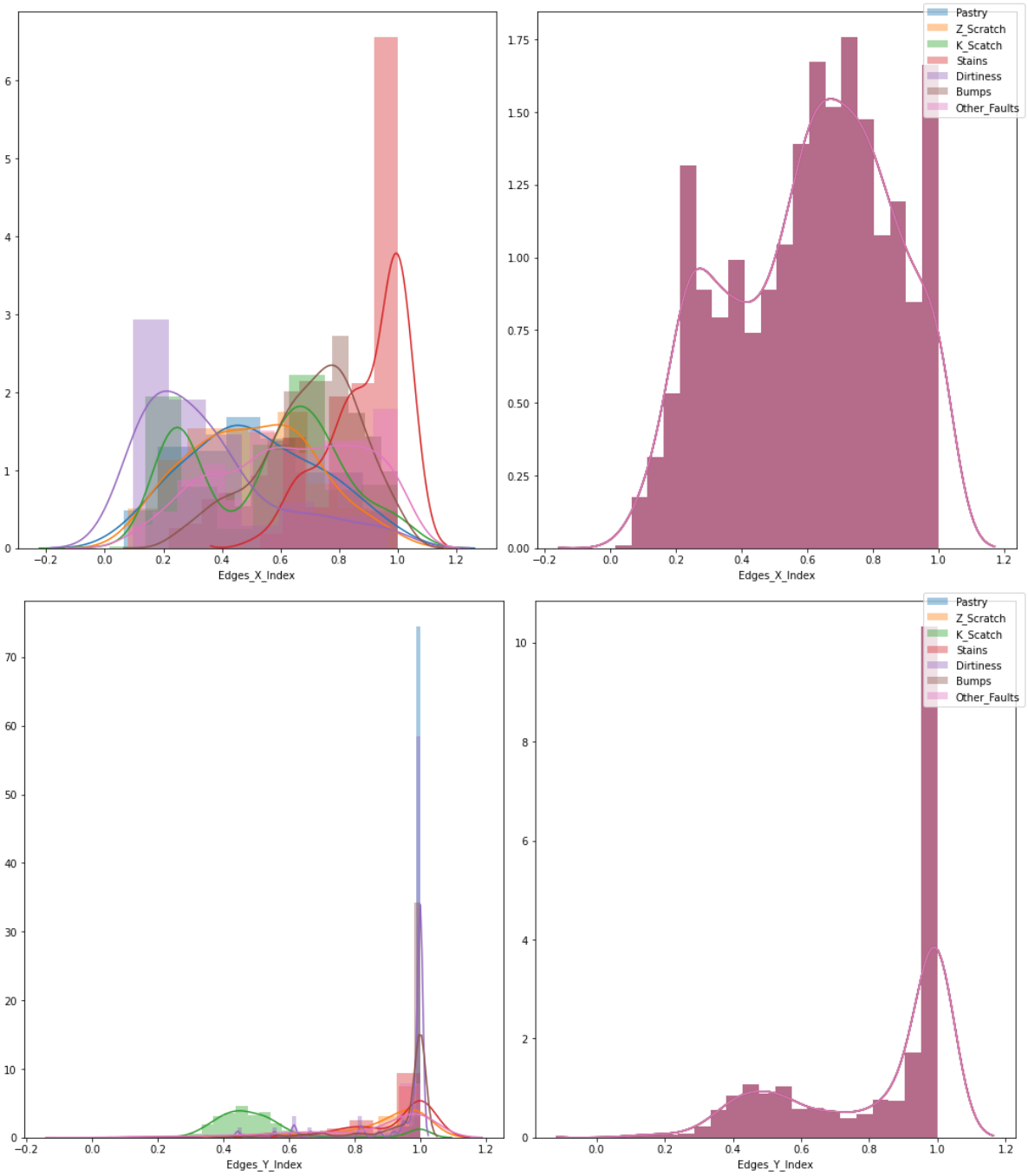

Pastry
Z_Scratch
K_Scatch
Stains
Dirtiness
Bumps
Other_Faults
Maximum_of_Luminosity
Maximum_of_Luminosity

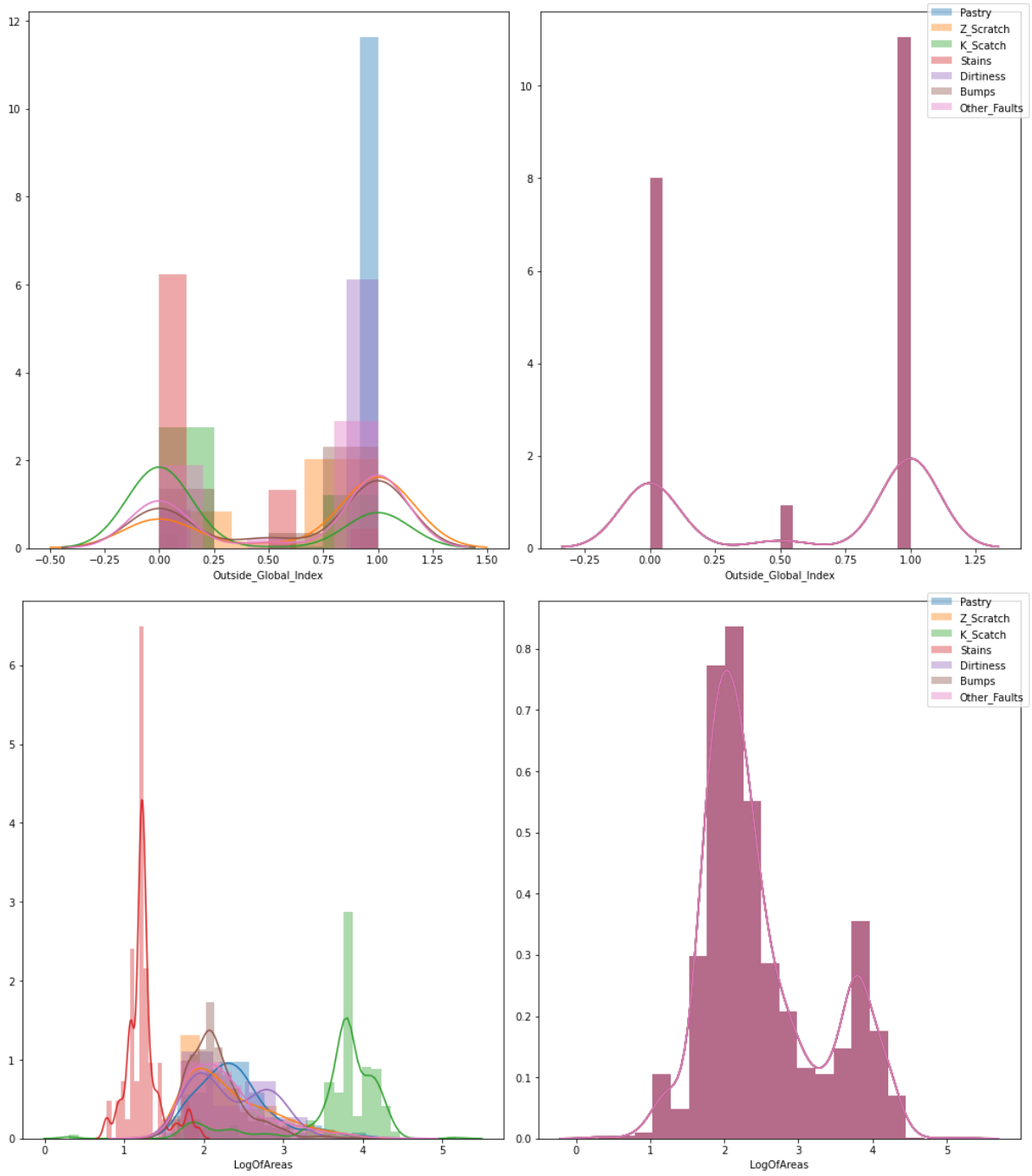

Pastry
Z_Scratch
K_Scatch
Stains
Dirtiness
Bumps
Other_Faults
Length_of_Conveyer
Length_of_Conveyer
TypeOfSteel_A300
TypeOfSteel_A300

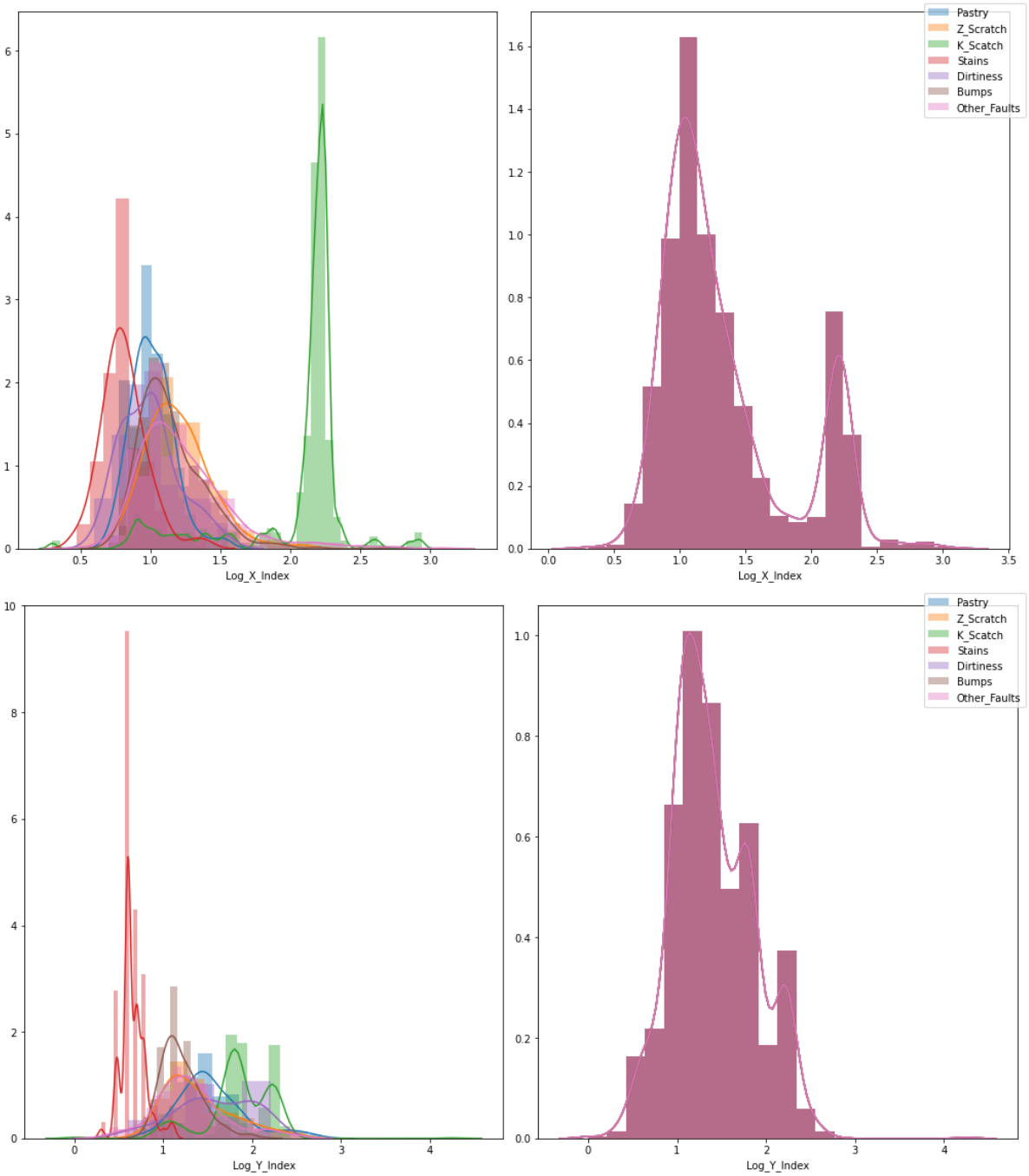

Pastry
Z_Scratch
K_Scatch
Stains
Dirtiness
Bumps
Other_Faults
TypeOfSteel_A400
TypeOfSteel_A400
Steel_Plate_Thickness
Steel_Plate_Thickness

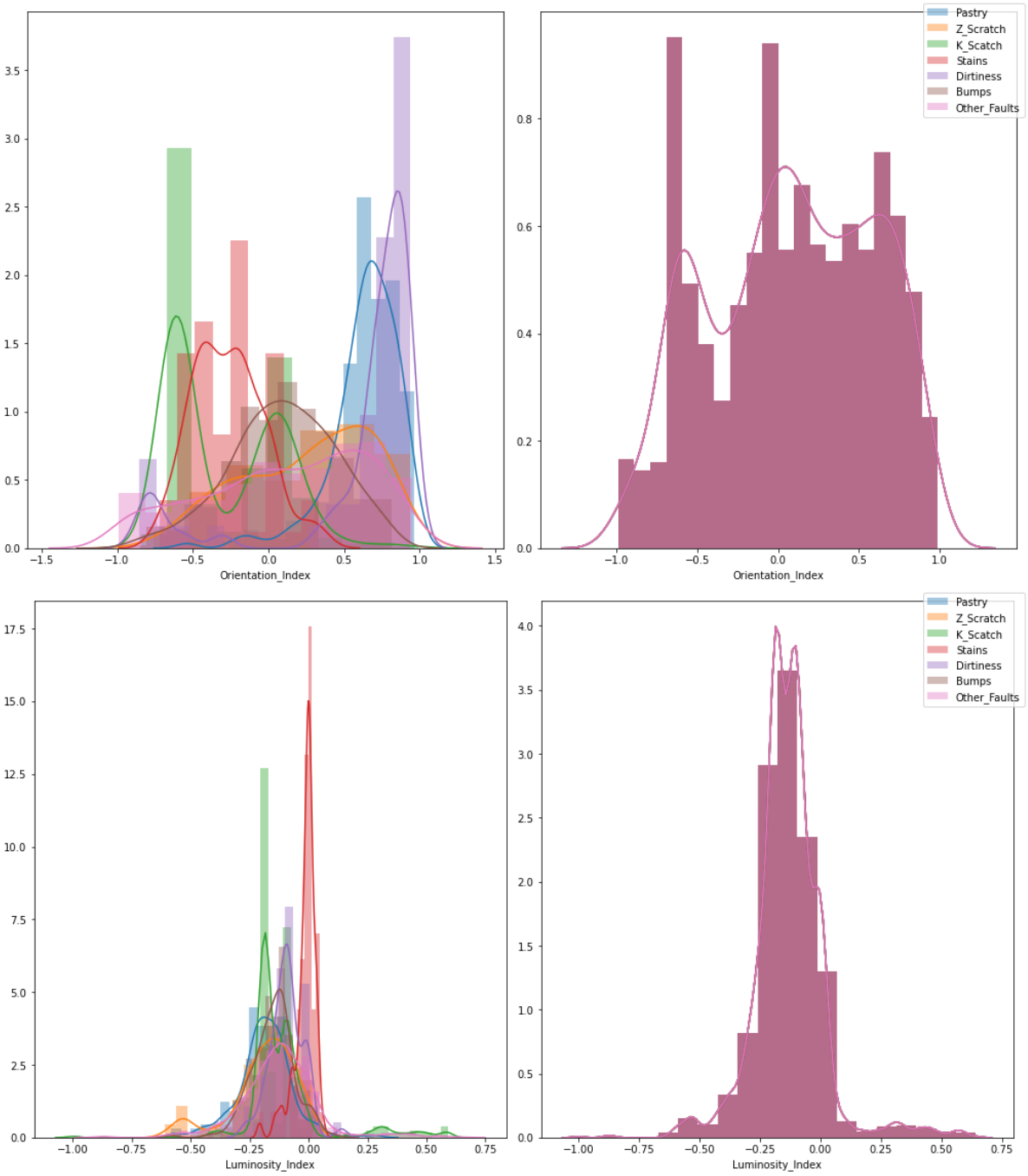

Pastry
Z_Scratch
K_Scatch
Stains
Dirtiness
Bumps
Other_Faults
Edges_Index
Edges_Index
Pastry
Z_Scratch
K_Scatch
Stains
Dirtiness
Bumps
Other_Faults
Empty_Index
Empty_Index

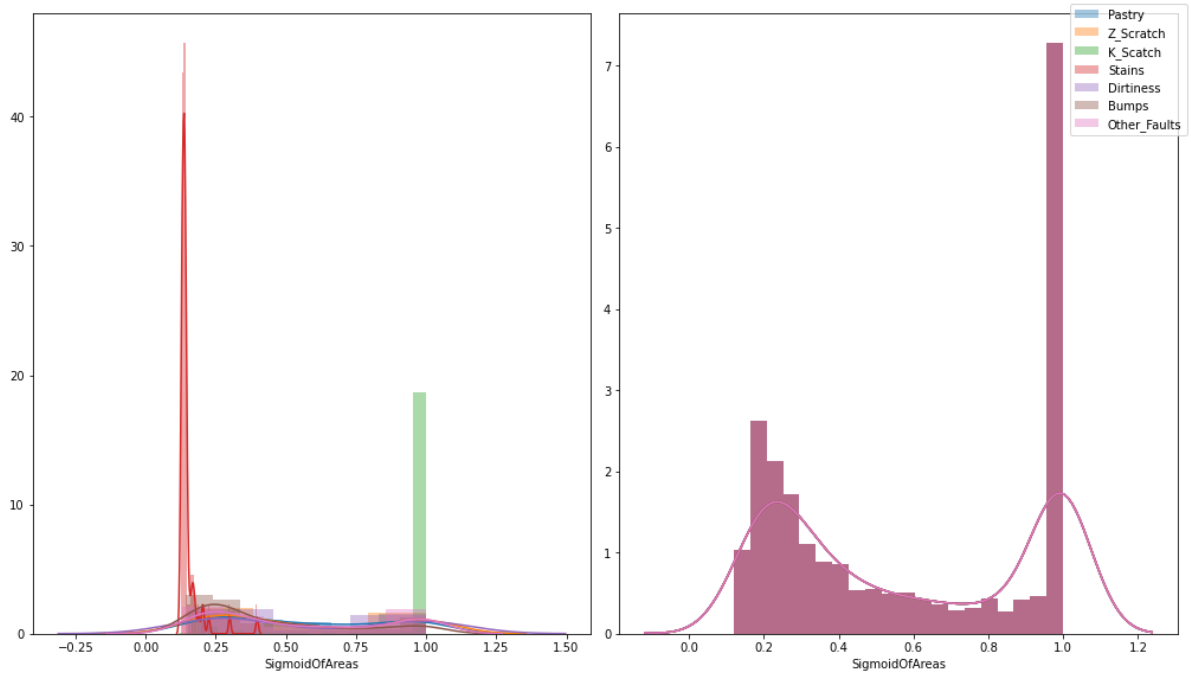

Pastry
Z_Scratch
K_Scatch
Stains
Dirtiness
Bumps
Other_Faults
Square_Index
Square_Index
Pastry
Z_Scratch
K_Scatch
Stains
Dirtiness
Bumps
Other_Faults
Outside_X_Index
Outside_X_Index


Pastry
Z_Scratch
K_Scatch
Stains
Dirtiness
Bumps
Other_Faults
Edges_X_Index
Edges_X_Index
Edges_Y_Index
Edges_Y_Index

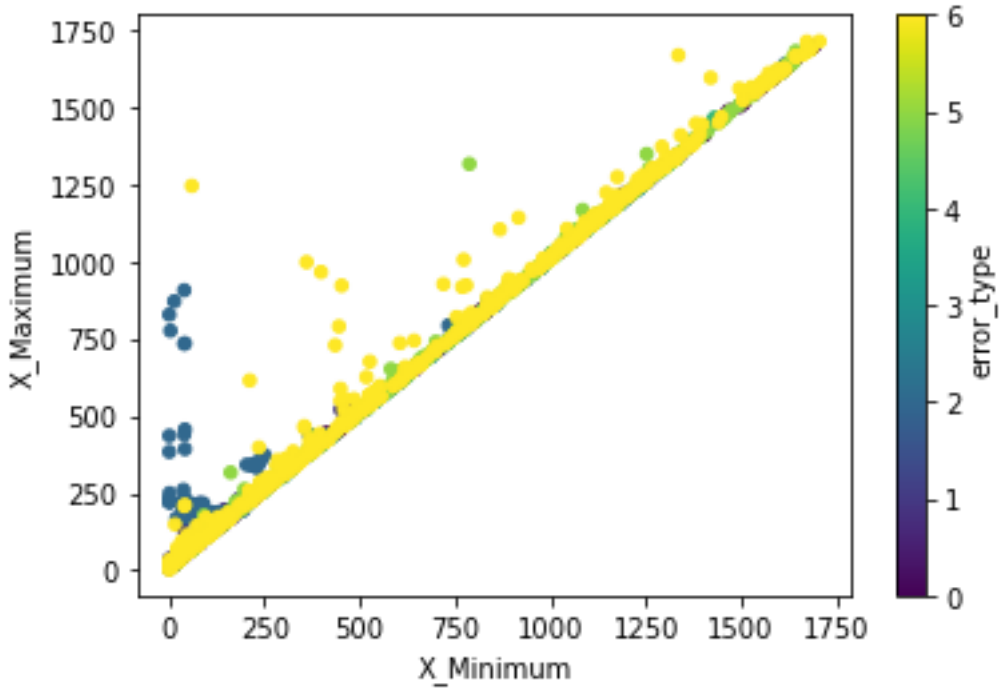

Pastry
Z_Scratch
K_Scatch
Stains
Dirtiness
Bumps
Other_Faults
Outside_Global_Index
Outside_Global_Index
Pastry
Z_Scratch
K_Scatch
Stains
Dirtiness
Bumps
Other_Faults
LogOfAreas
LogOfAreas

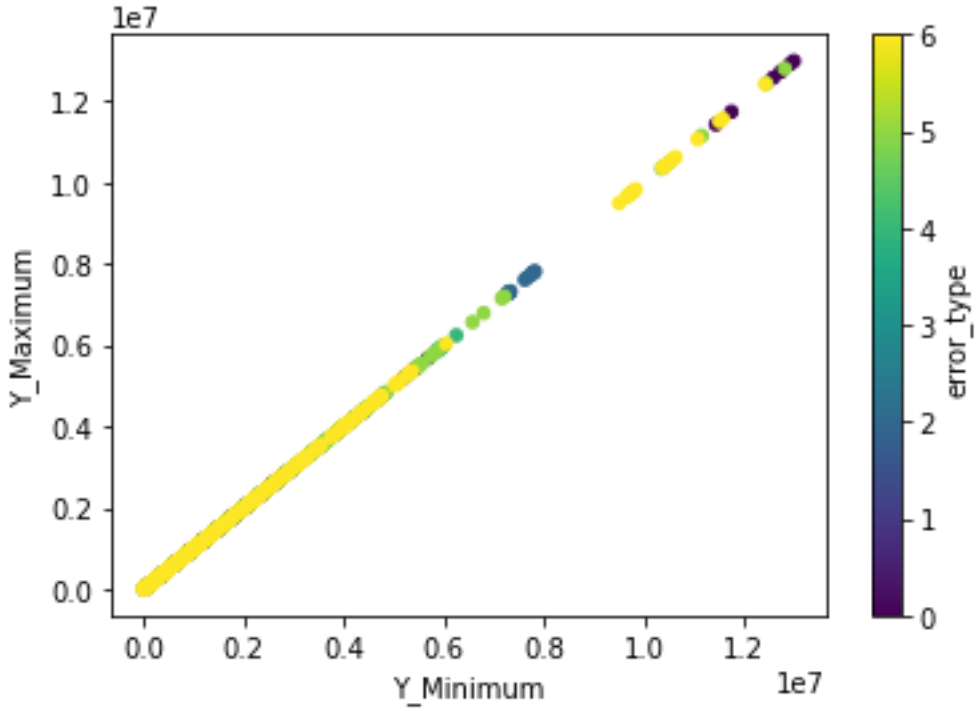

Pastry
Z_Scratch
K_Scatch
Stains
Dirtiness
Bumps
Other_Faults
Log_X_Index
Log_X_Index
Log_Y_Index
Log_Y_Index

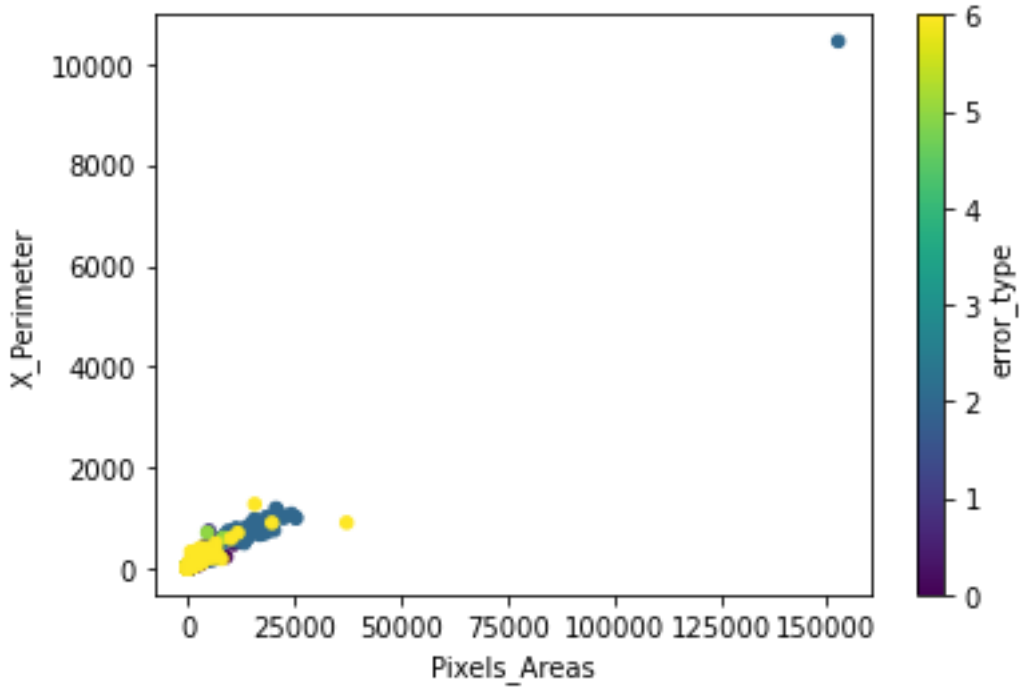

Pastry
Z_Scratch
K_Scatch
Stains
Dirtiness
Bumps
Other_Faults
Orientation_Index
Orientation_Index
Pastry
Z_Scratch
K_Scatch
Stains
Dirtiness
Bumps
Other_Faults
Luminosity_Index
Luminosity_Index

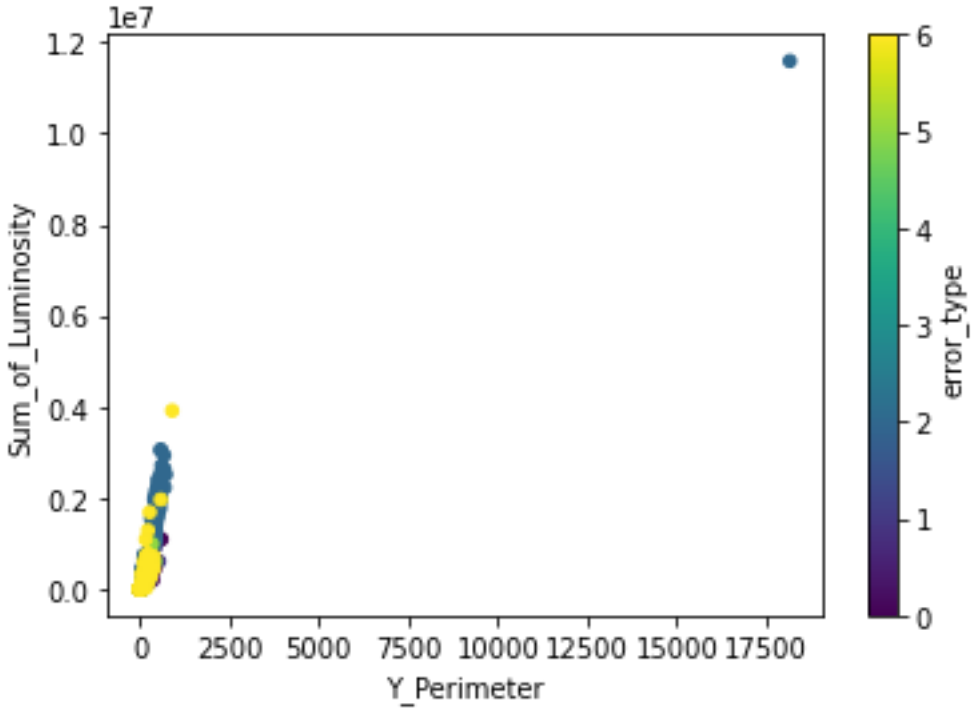


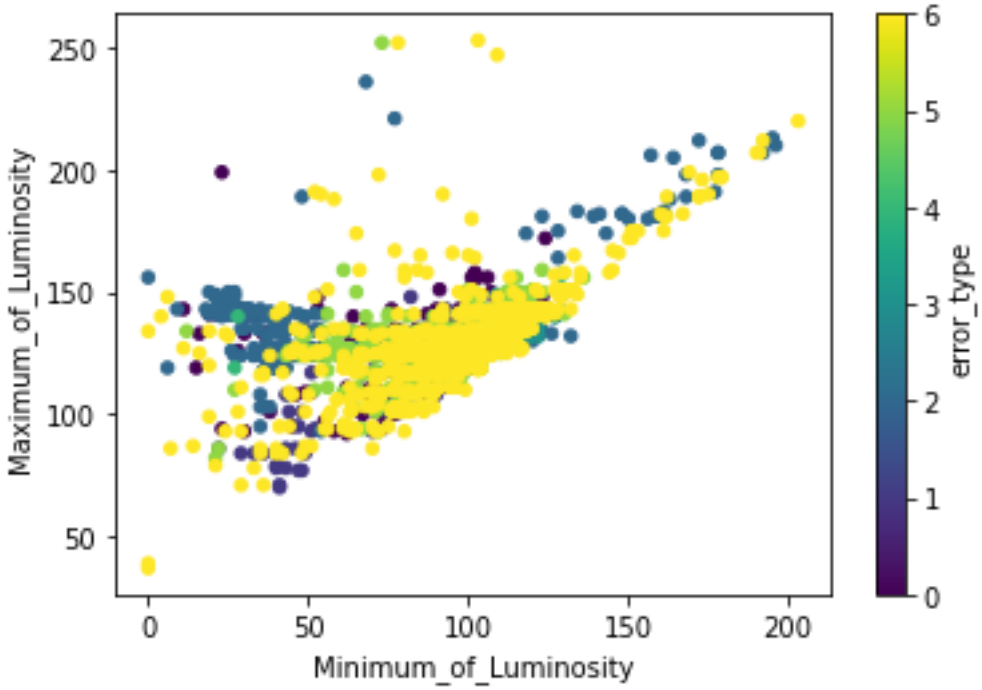
ΠΑΡΑΡΤΗΜΑ 3. ΔΙΑΓΡΑΜΜΑΤΑ ΧΑΡΑΚΤΗΡΙΣΤΙΚΩΝ

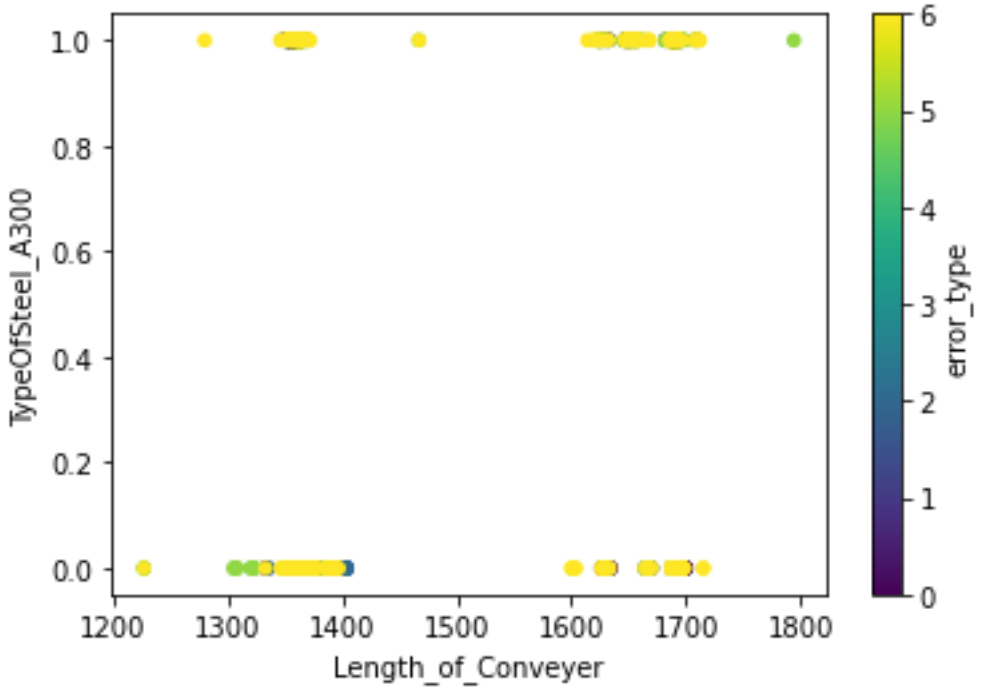


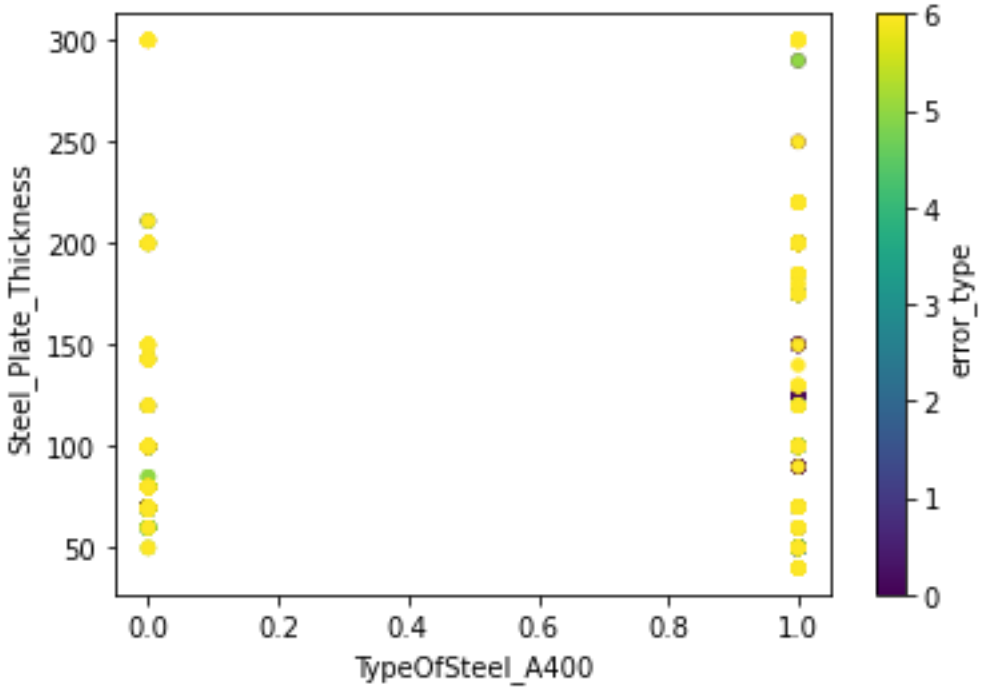

1e7
1.2
1.0
0.8
0.6
0.4
0.2
0.0
Y_Maximum
0.0
0.2
0.4
0.6
0.8
1.0
1.2
1e7
Y_Minimum
6
5
4
3
2
1
0
error_type

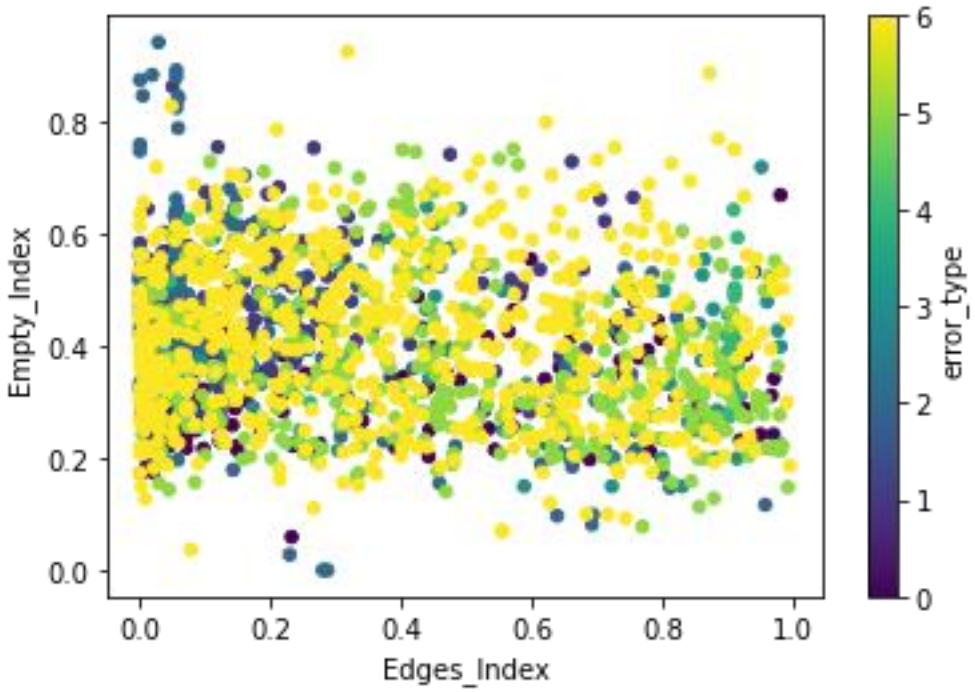

10000
8000
6000
4000
2000
0
X_Perimeter
0
25000
50000
75000
100000
125000
150000
Pixels_Areas
6
5
4
3
2
1
0
error_type

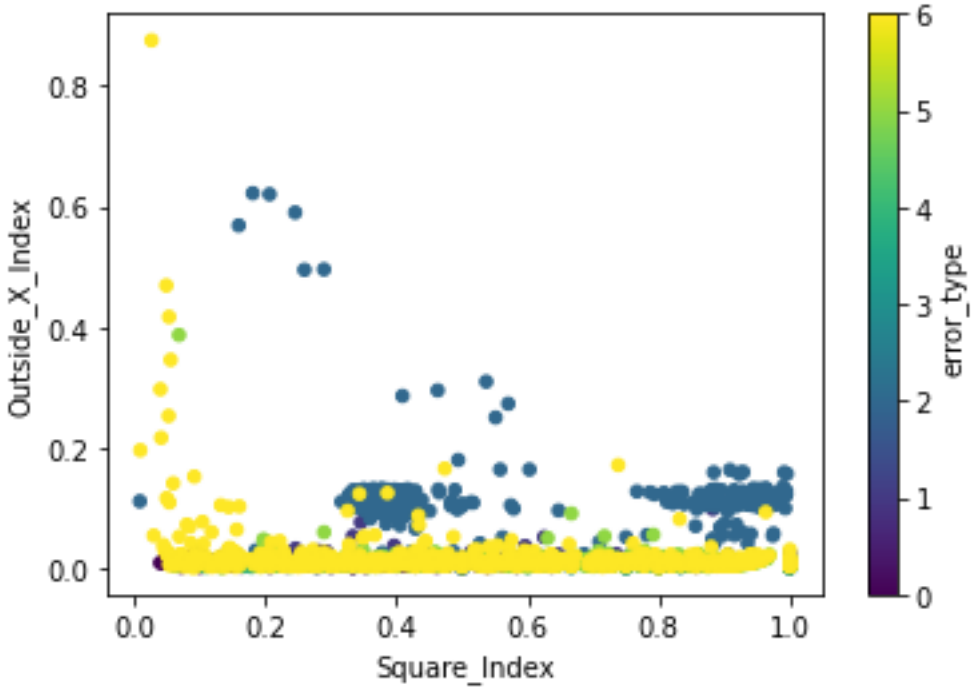

1e7
1.2
1.0
0.8
0.6
0.4
0.2
0.0
Sum_of_Luminosity
0
2500
5000
7500
10000
12500
15000
17500
Y_Perimeter
6
5
4
3
2
1
0
error_type

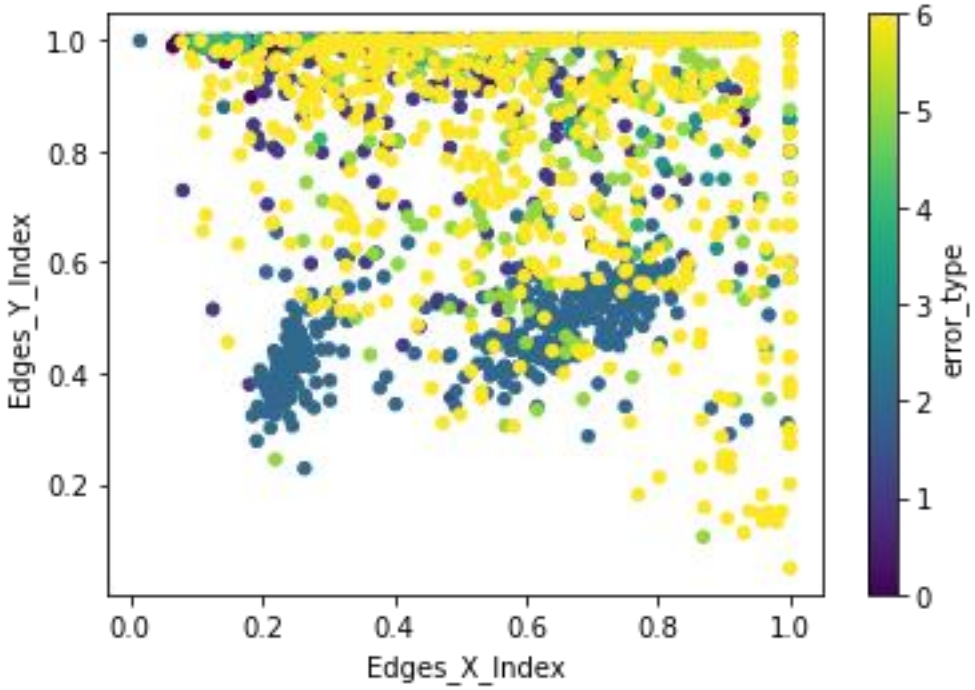

250
200
150
100
50
Maximum_of_Luminosity
0
50
100
150
200
Minimum_of_Luminosity
6
5
4
3
2
1
0
error_type

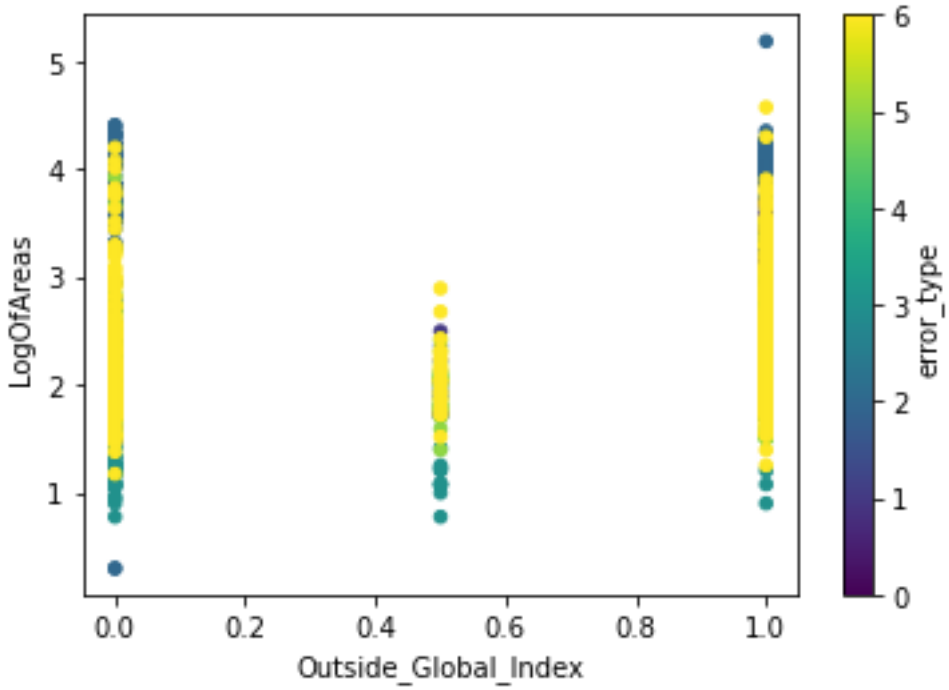

TypeOfSteel_A300
Length_of_Conveyer
error_type
1.0
0.8
0.6
0.4
0.2
0.0
1200
1300
1400
1500
1600
1700
1800
6
5
4
3
2
1
0

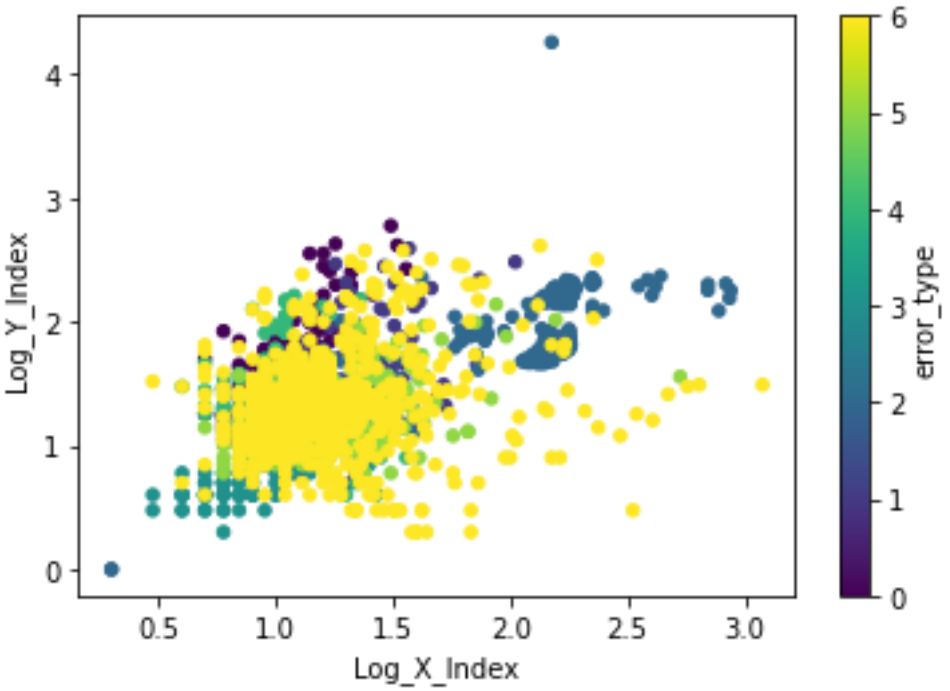

Steel_Plate_Thickness
TypeOfSteel_A400
error_type
300
250
200
150
100
50
0.0
0.2
0.4
0.6
0.8
1.0
6
5
4
3
2
1
0

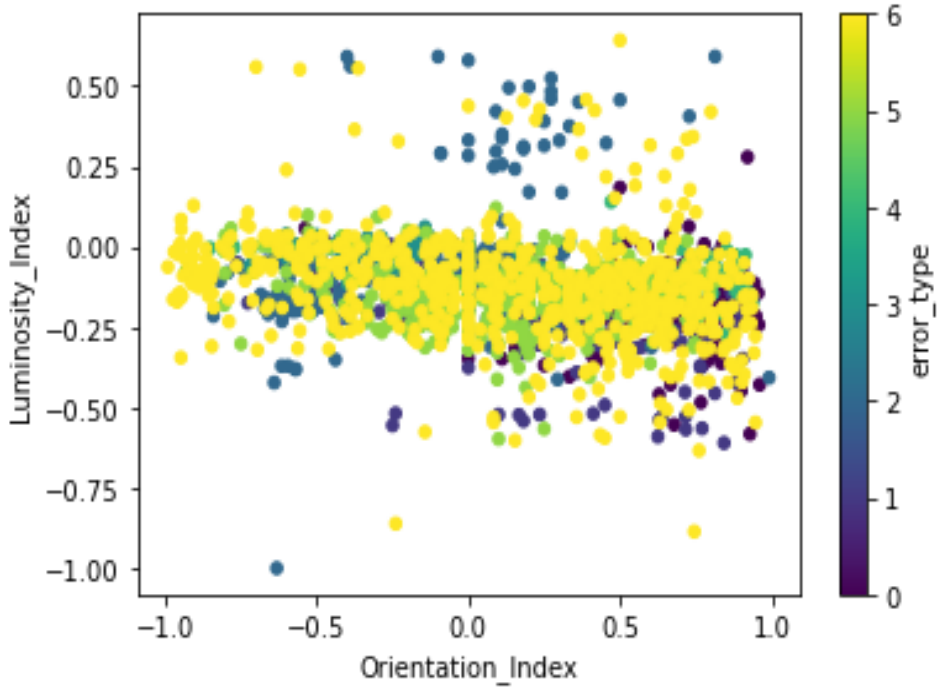

Empty_Index
Edges_Index
error_type
0.0
0.2
0.4
0.6
0.8
1.0
0
1
2
3
4
5
6


Outside_X_Index
Square_Index
error_type
0.0
0.2
0.4
0.6
0.8
1.0
0
1
2
3
4
5
6


Edges_Y_Index
Edges_X_Index
error_type
1.0
0.8
0.6
0.4
0.2
0.0
0.2
0.4
0.6
0.8
1.0
0
1
2
3
4
5
6


LogOfAreas
Outside_Global_Index
error_type
5
4
3
2
1
0.0
0.2
0.4
0.6
0.8
1.0
0
1
2
3
4
5
6


Log_Y_Index
Log_X_Index
error_type
0
1
2
3
4
0.5
1.0
1.5
2.0
2.5
3.0
5
6

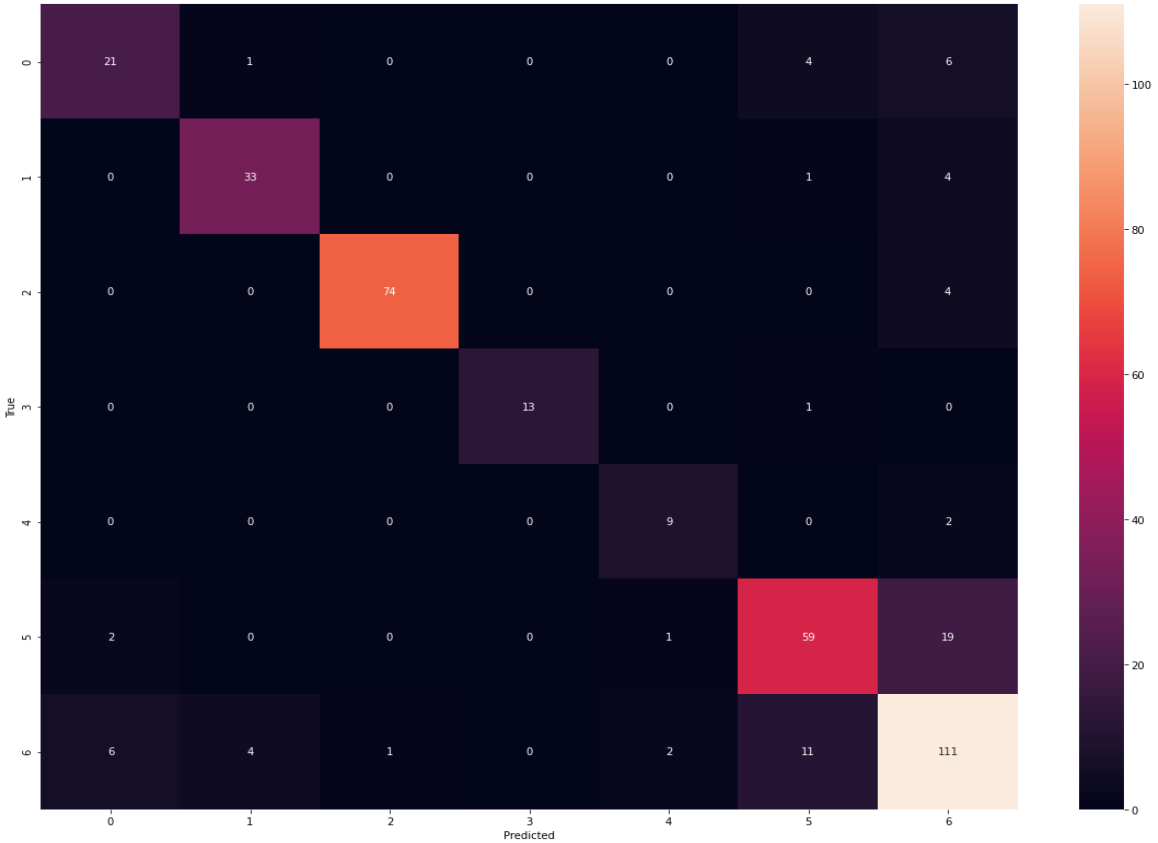

Luminosity_Index
Orientation_Index
error_type
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00
−1.0
−0.5
0.0
0.5
1.0
0
1
2
3
4
5
6

ΠΑΡΑΡΤΗΜΑ 4. ΠΙΝΑΚΕΣ ΣΥΓΧΥΣΗΣ – ΚΕΦΑΛΑΙΟ 4

Αποτελέσματα μοντέλων με ρύθμιση παραμέτρων

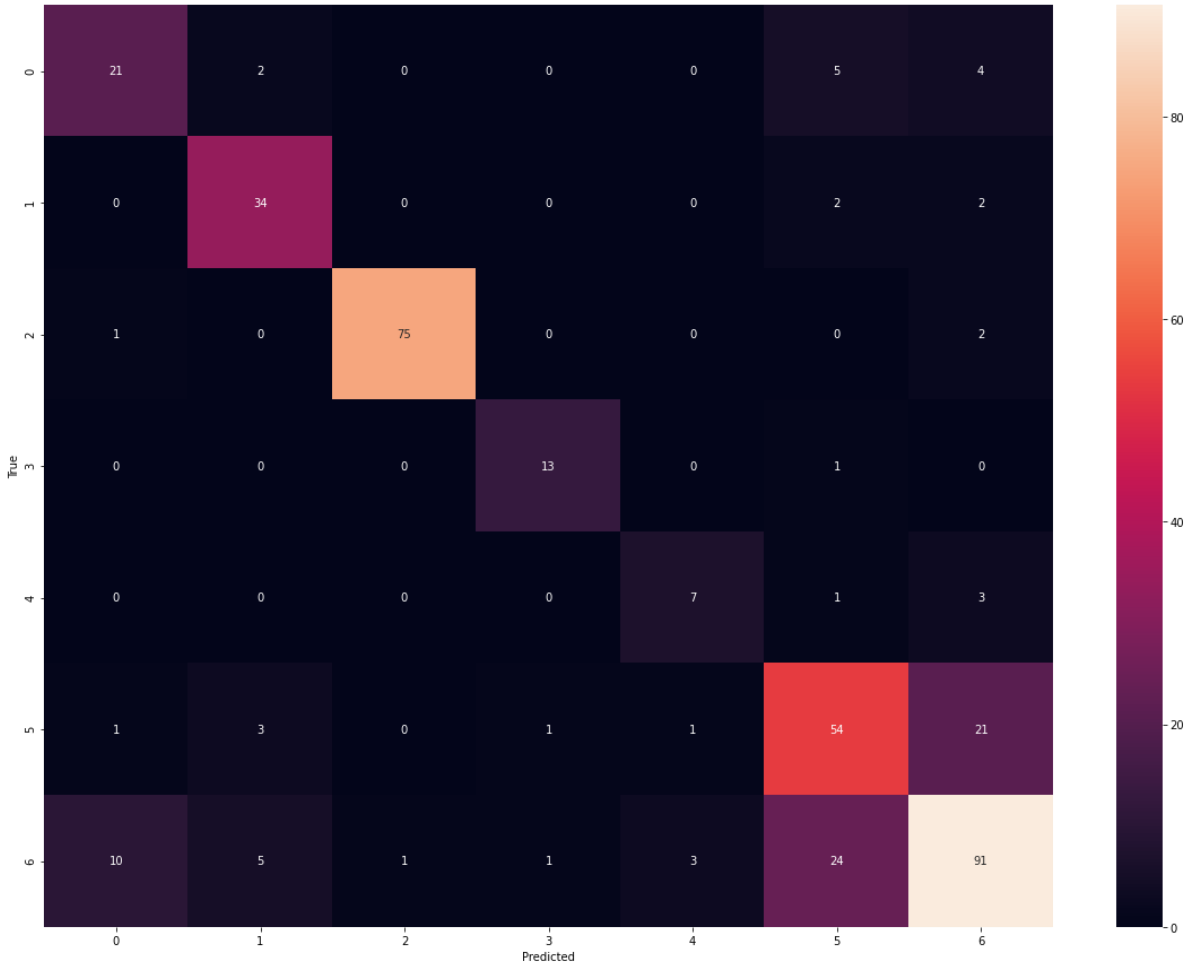
Random Forest Confusion Matrix (test set)

Classification Report:				
	precision	recall	f1-score	support
1	0.72	0.66	0.69	32
2	0.87	0.87	0.87	38
3	0.99	0.95	0.97	78
4	1.00	0.93	0.96	14
5	0.75	0.82	0.78	11
6	0.78	0.73	0.75	81
7	0.76	0.82	0.79	135
accuracy			0.82	389
macro avg	0.84	0.82	0.83	389
weighted avg	0.82	0.82	0.82	389



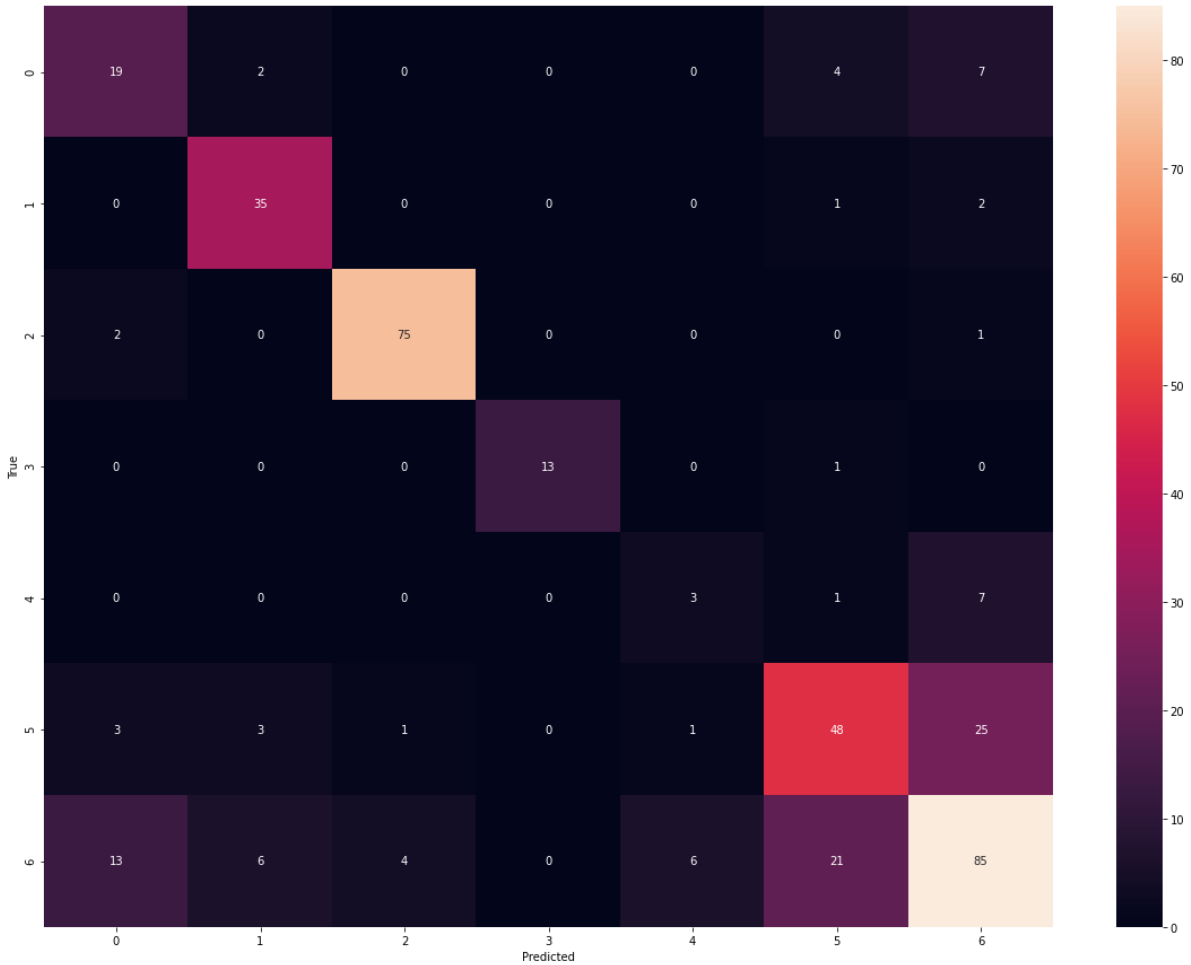
Support Vector Classifier - Confusion Matrix(test set)

Classification Report:				
	precision	recall	f1-score	support
1	0.64	0.66	0.65	32
2	0.77	0.89	0.83	38
3	0.99	0.96	0.97	78
4	0.87	0.93	0.90	14
5	0.64	0.64	0.64	11
6	0.62	0.67	0.64	81
7	0.74	0.67	0.71	135
accuracy			0.76	389
macro avg	0.75	0.77	0.76	389
weighted avg	0.76	0.76	0.76	389



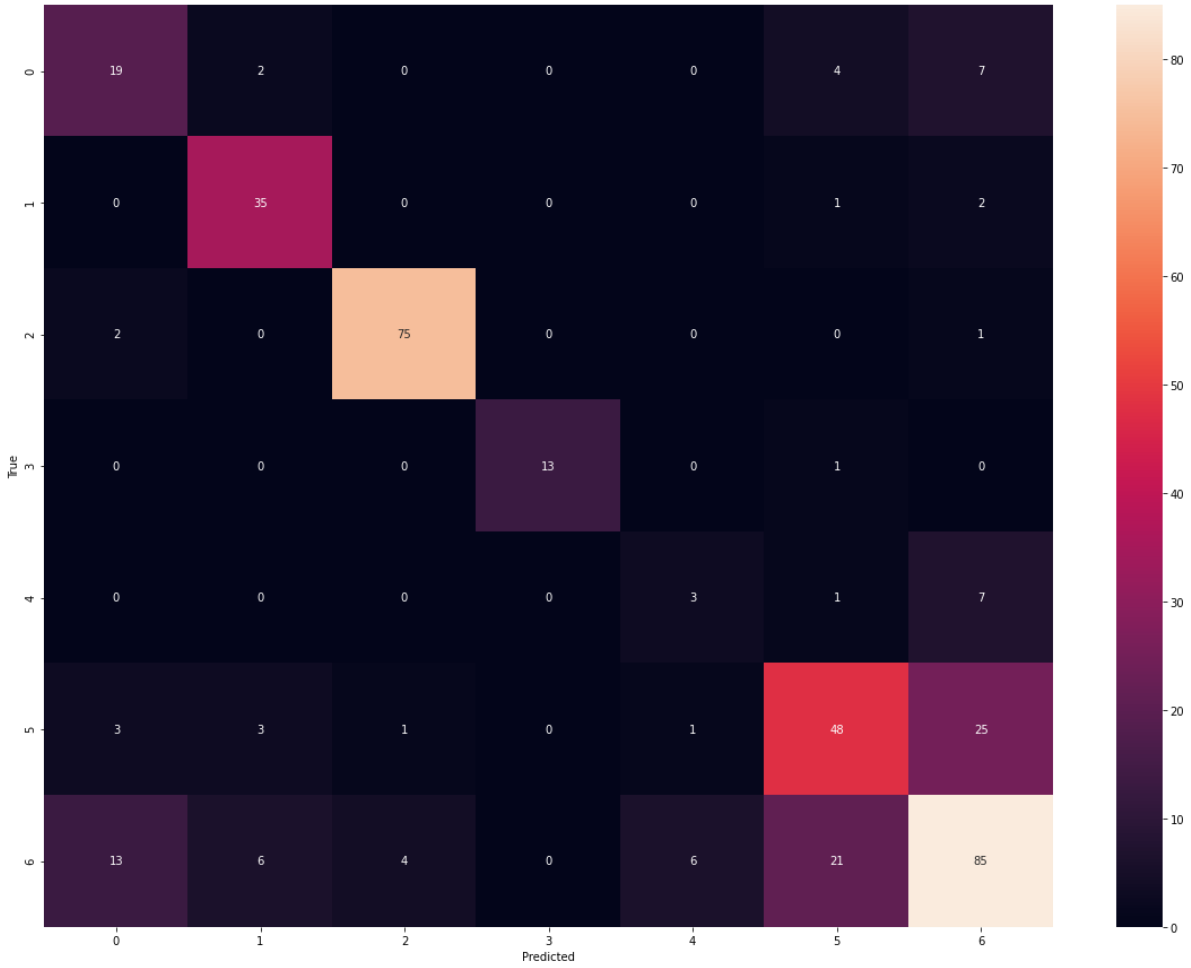
Logistic Regression - Confusion Matrix (test set)

Classification Report:				
	precision	recall	f1-score	support
1	0.51	0.59	0.55	32
2	0.76	0.92	0.83	38
3	0.94	0.96	0.95	78
4	1.00	0.93	0.96	14
5	0.30	0.27	0.29	11
6	0.63	0.59	0.61	81
7	0.67	0.63	0.65	135
accuracy			0.71	389
macro avg	0.69	0.70	0.69	389
weighted avg	0.71	0.71	0.71	389



k-Nearest Neighbor - Confusion Matrix (test set)

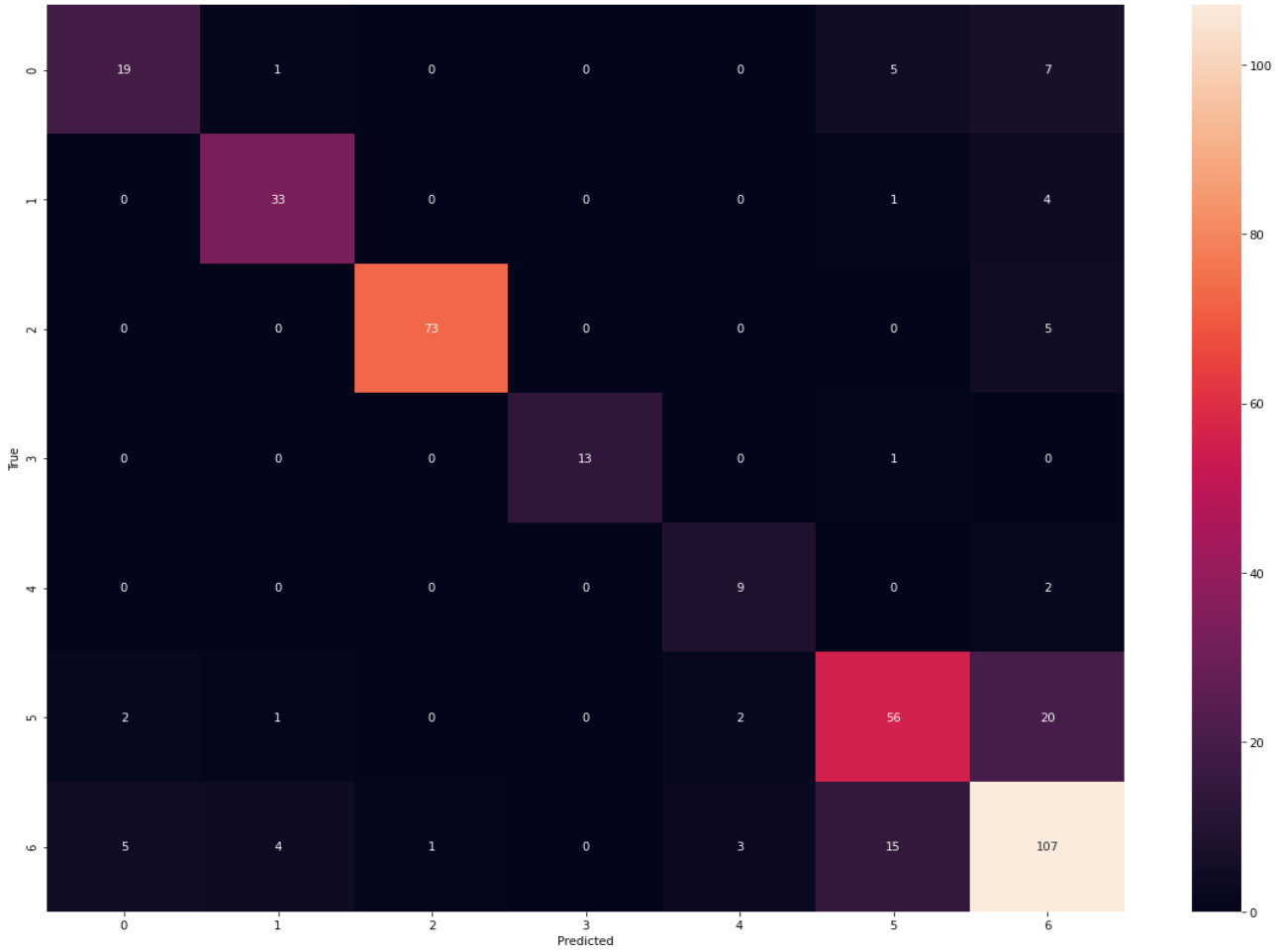
	precision	recall	f1-score	support
1	0.68	0.53	0.60	32
2	0.71	0.95	0.81	38
3	0.99	0.96	0.97	78
4	0.71	0.86	0.77	14
5	0.57	0.73	0.64	11
6	0.59	0.57	0.58	81
7	0.68	0.64	0.66	135
accuracy			0.72	389
macro avg	0.70	0.75	0.72	389
weighted avg	0.72	0.72	0.72	389



Αποτελέσματα μοντέλων με μείωση διάστασης και προσθήκη των components στο αρχικό dataset

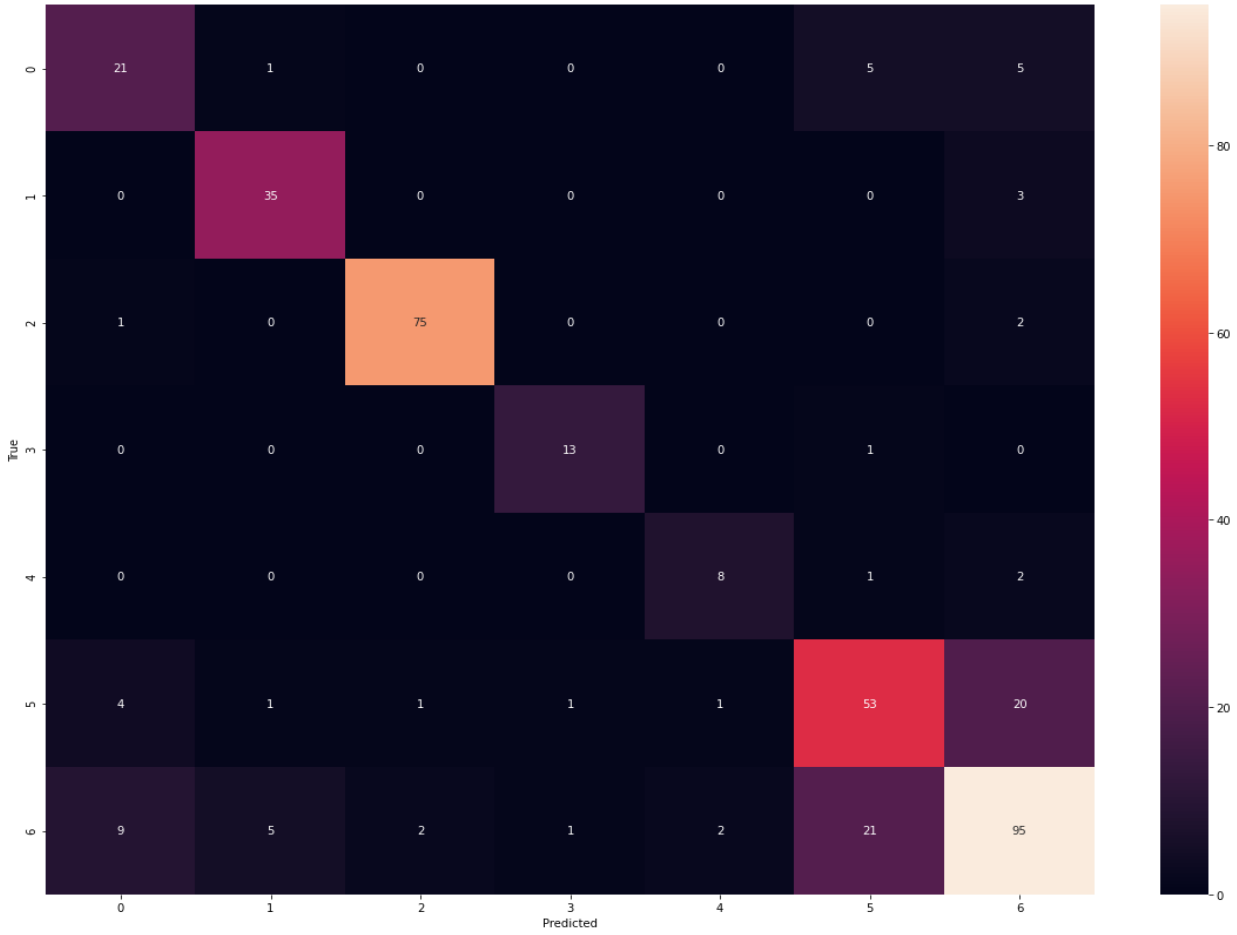
Random Forest Confusion Matrix (test set)

Classification Report:				
	precision	recall	f1-score	support
1	0.73	0.59	0.66	32
2	0.85	0.87	0.86	38
3	0.99	0.94	0.96	78
4	1.00	0.93	0.96	14
5	0.64	0.82	0.72	11
6	0.72	0.69	0.70	81
7	0.74	0.79	0.76	135
accuracy			0.80	389
macro avg	0.81	0.80	0.80	389
weighted avg	0.80	0.80	0.80	389



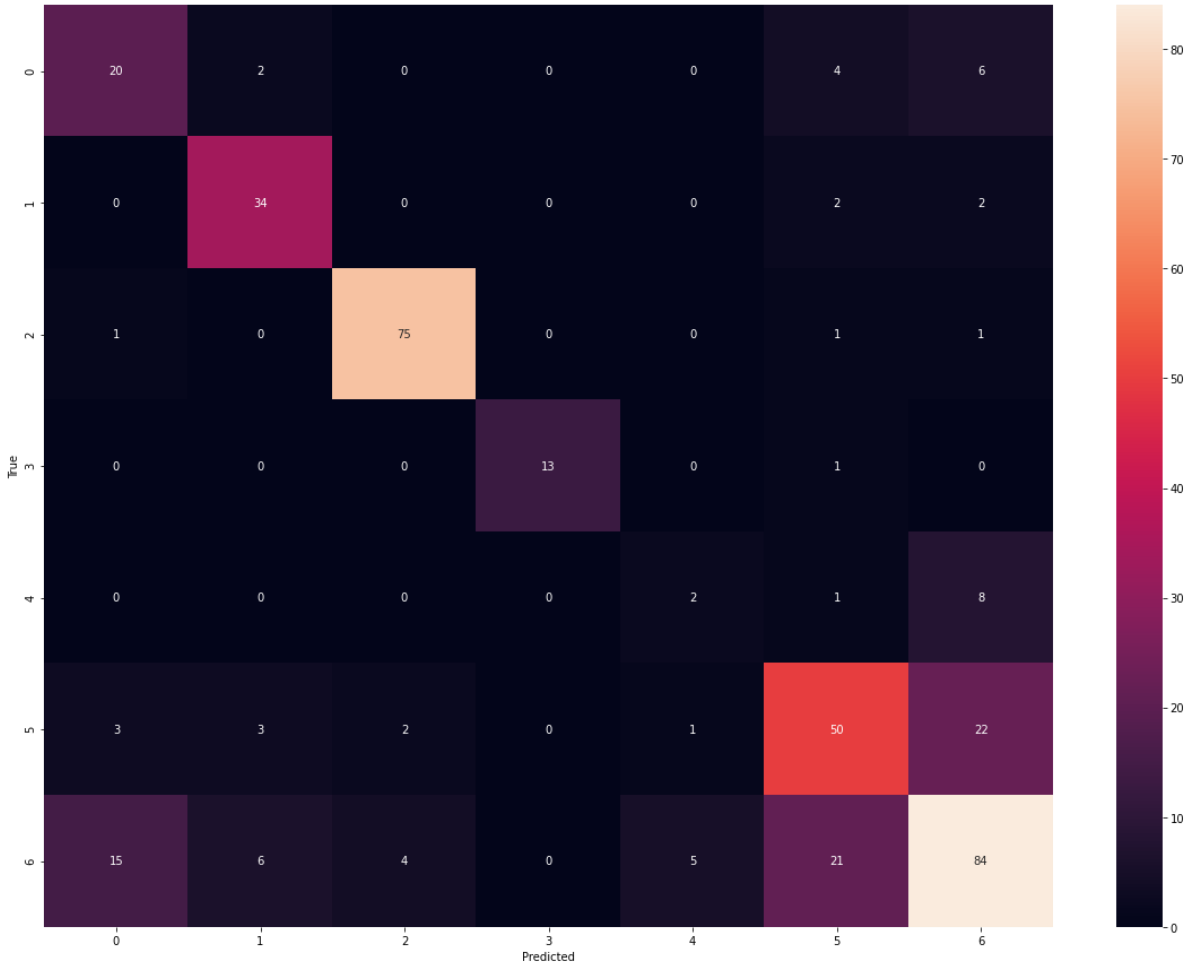
Support Vector Classifier - Confusion Matrix (test set)

Classification Report:				
	precision	recall	f1-score	support
1	0.60	0.66	0.63	32
2	0.83	0.92	0.88	38
3	0.96	0.96	0.96	78
4	0.87	0.93	0.90	14
5	0.73	0.73	0.73	11
6	0.65	0.65	0.65	81
7	0.75	0.70	0.73	135
accuracy			0.77	389
macro avg	0.77	0.79	0.78	389
weighted avg	0.77	0.77	0.77	389



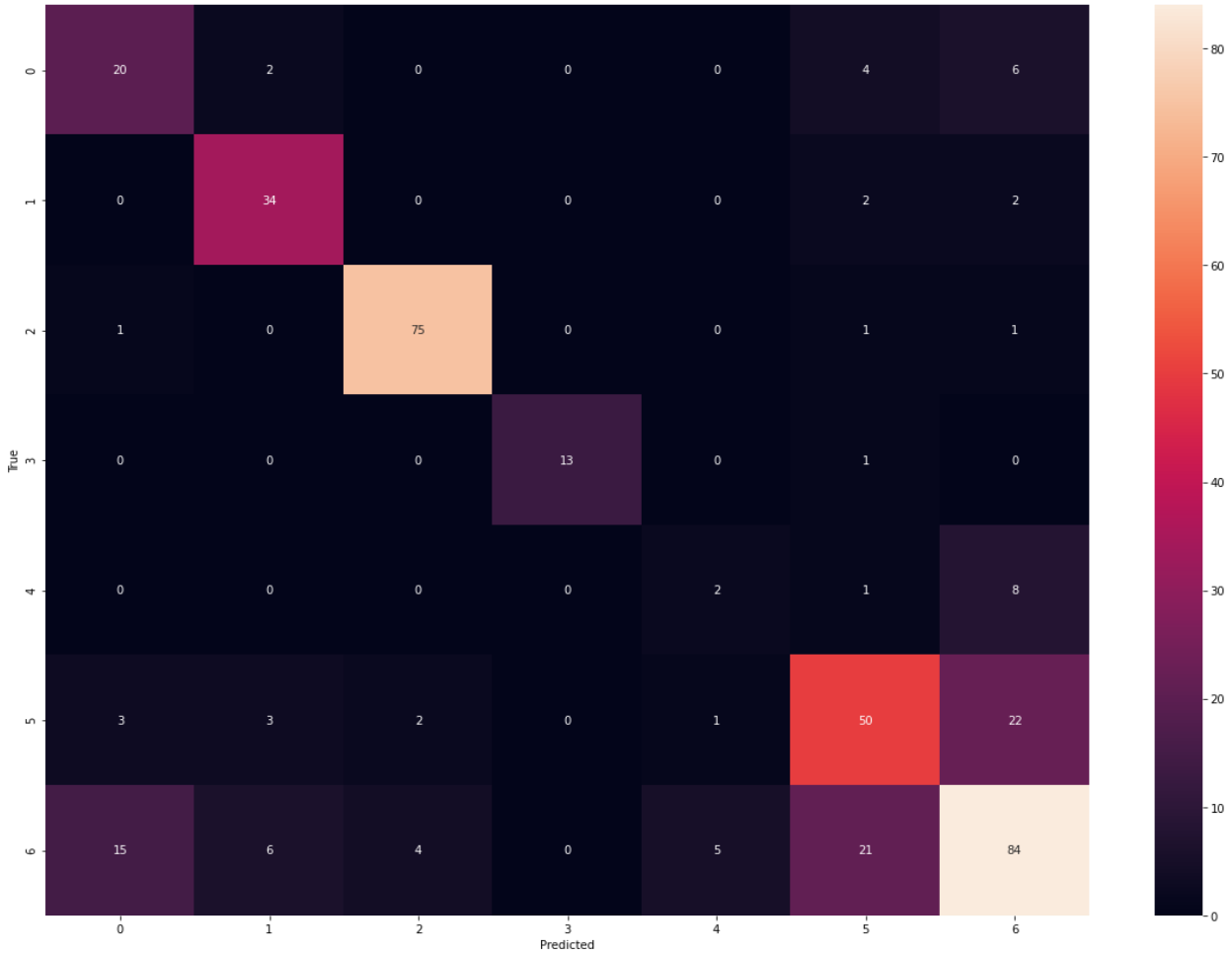
Logistic Regression - Confusion Matrix (test set)

Classification Report:				
	precision	recall	f1-score	support
1	0.51	0.62	0.56	32
2	0.76	0.89	0.82	38
3	0.93	0.96	0.94	78
4	1.00	0.93	0.96	14
5	0.25	0.18	0.21	11
6	0.62	0.62	0.62	81
7	0.68	0.62	0.65	135
accuracy			0.71	389
macro avg	0.68	0.69	0.68	389
weighted avg	0.71	0.71	0.71	389



k-N-N - Confusion Matrix (test set)

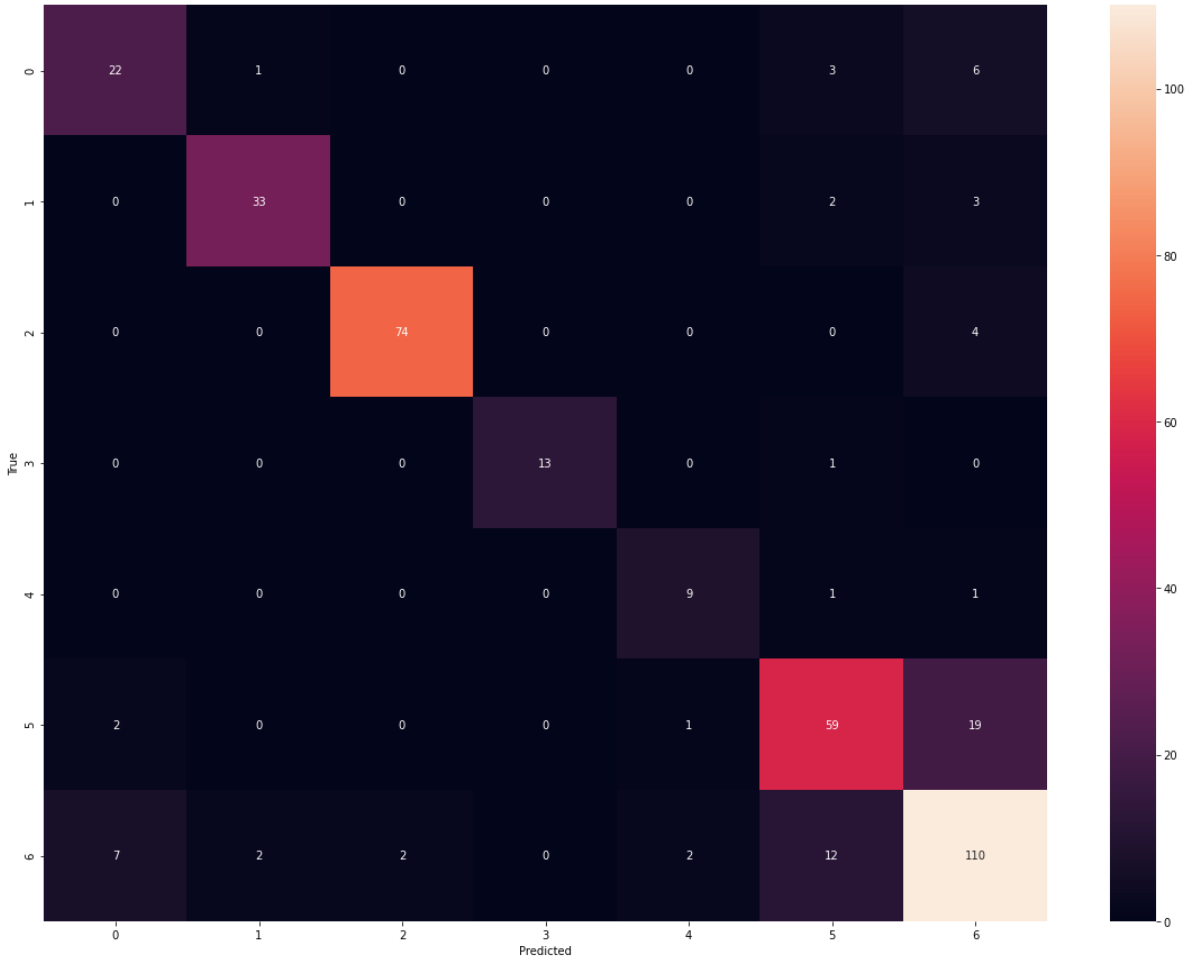
Classification Report:				
	precision	recall	f1-score	support
1	0.61	0.59	0.60	32
2	0.74	0.89	0.81	38
3	0.97	0.97	0.97	78
4	0.80	0.86	0.83	14
5	0.57	0.73	0.64	11
6	0.59	0.62	0.60	81
7	0.70	0.62	0.66	135
accuracy			0.73	389
macro avg	0.71	0.76	0.73	389
weighted avg	0.73	0.73	0.73	389



Αποτελέσματα μοντέλων με προσθήκη πολυωνύμων δεύτερης τάξης

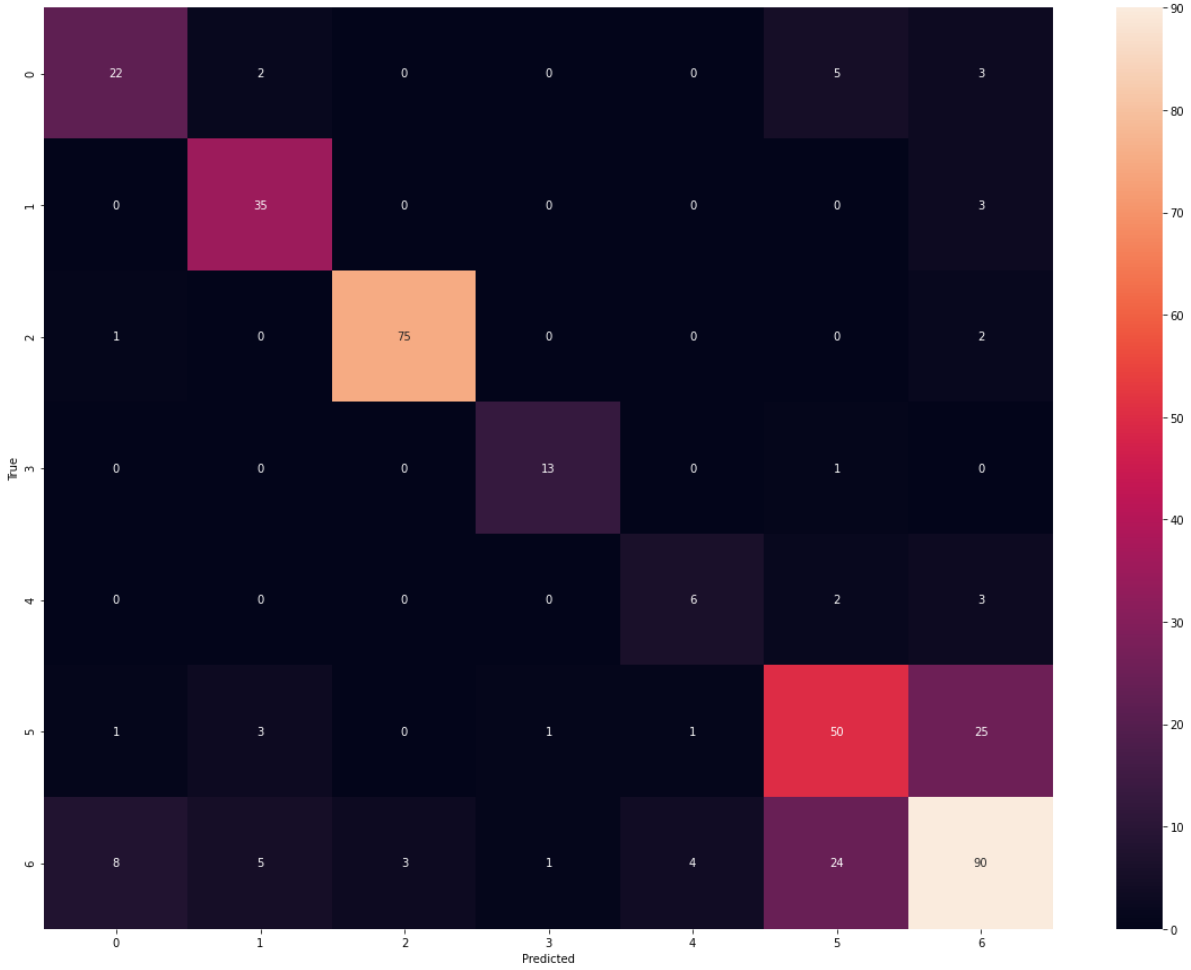
RF – Polynomial Feature Engineering - Confusion Matrix (test set)

Classification Report:				
	precision	recall	f1-score	support
1	0.71	0.69	0.70	32
2	0.92	0.87	0.89	38
3	0.97	0.95	0.96	78
4	1.00	0.93	0.96	14
5	0.75	0.82	0.78	11
6	0.76	0.73	0.74	81
7	0.77	0.81	0.79	135
accuracy			0.82	389
macro avg	0.84	0.83	0.83	389
weighted avg	0.82	0.82	0.82	389



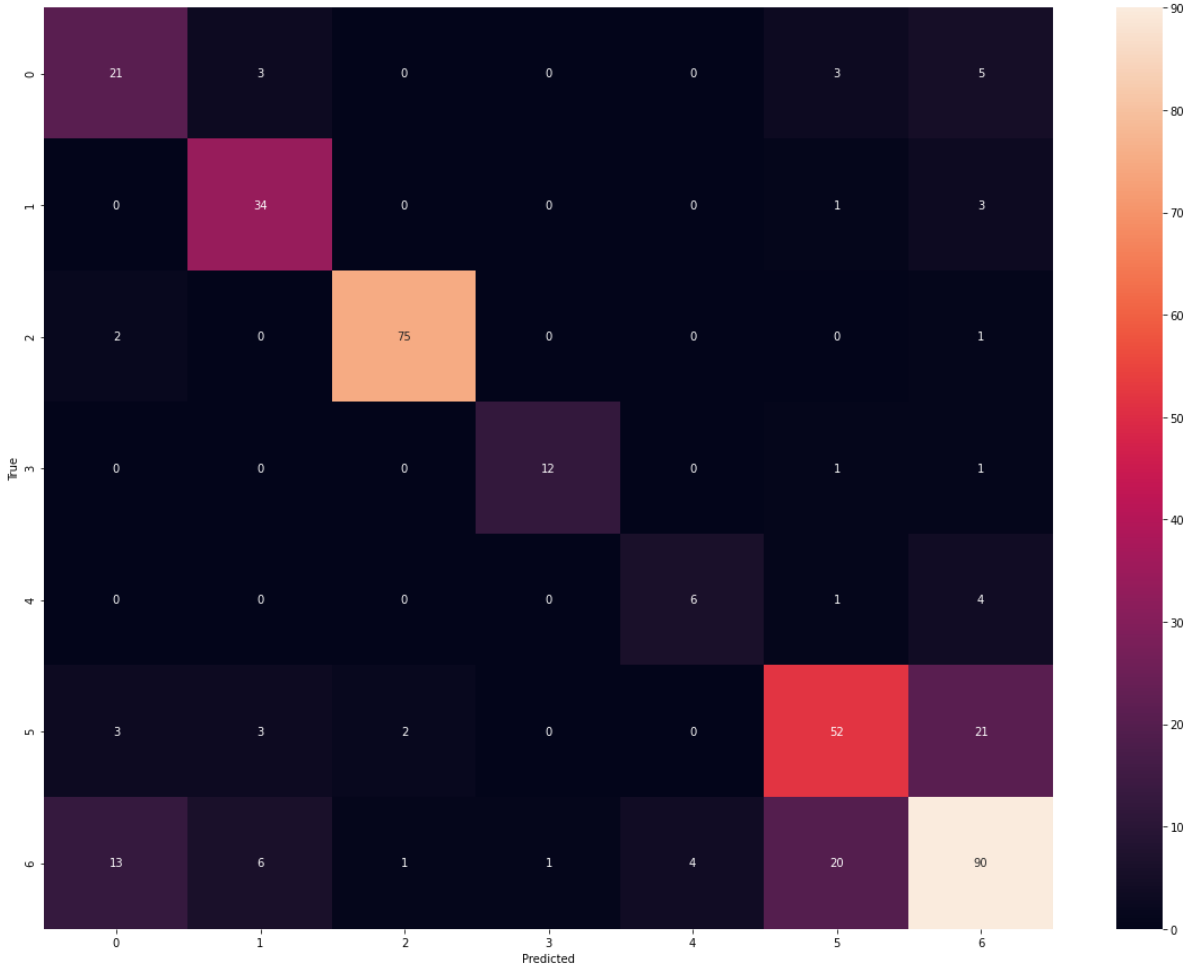
S|VC – Polynomial Feature Engineering - Confusion Matrix (test set)

Classification Report:				
	precision	recall	f1-score	support
1	0.69	0.69	0.69	32
2	0.78	0.92	0.84	38
3	0.96	0.96	0.96	78
4	0.87	0.93	0.90	14
5	0.55	0.55	0.55	11
6	0.61	0.62	0.61	81
7	0.71	0.67	0.69	135
accuracy			0.75	389
macro avg	0.74	0.76	0.75	389
weighted avg	0.75	0.75	0.75	389



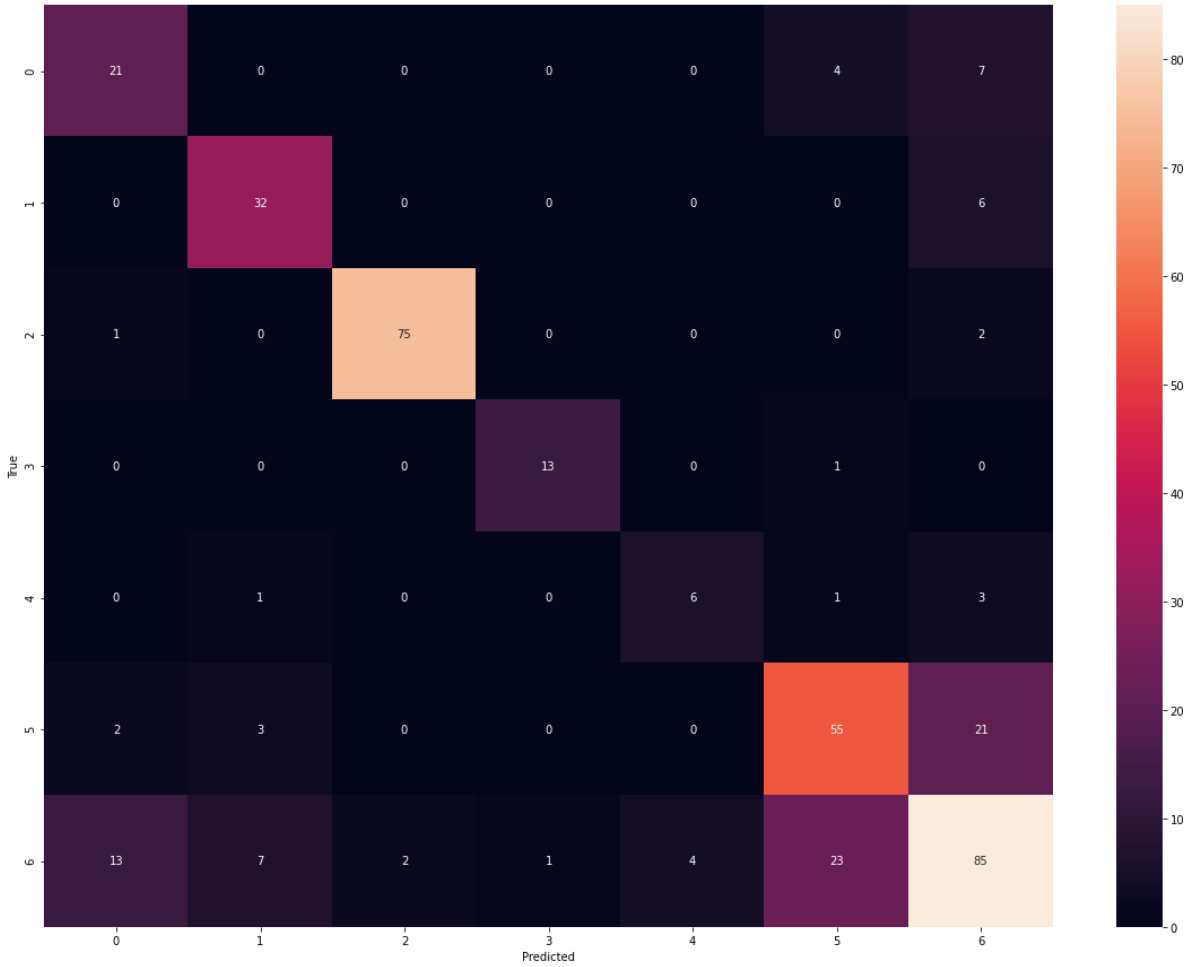
LR – Polynomial Feature Engineering – Confusion Matrix (test set)

lassification Report:				
	precision	recall	f1-score	support
1	0.54	0.66	0.59	32
2	0.74	0.89	0.81	38
3	0.96	0.96	0.96	78
4	0.92	0.86	0.89	14
5	0.60	0.55	0.57	11
6	0.67	0.64	0.65	81
7	0.72	0.67	0.69	135
accuracy			0.75	389
macro avg	0.74	0.75	0.74	389
weighted avg	0.75	0.75	0.75	389



k-N-N – Polynomial Feature Engineering – Confusion Matrix (test set)

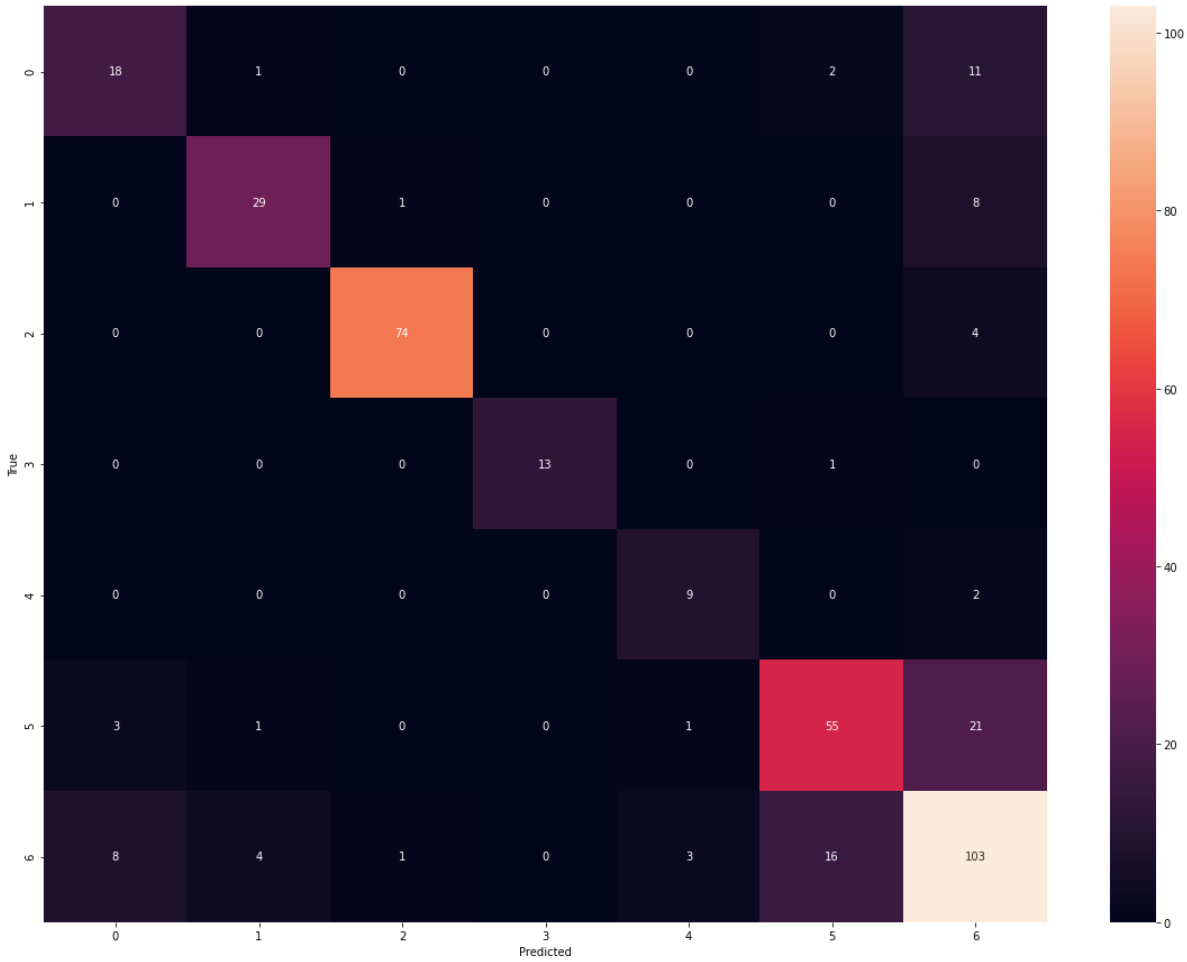
Classification Report:				
	precision	recall	f1-score	support
1	0.58	0.59	0.58	32
2	0.64	0.79	0.71	38
3	0.95	0.94	0.94	78
4	0.93	0.93	0.93	14
5	0.62	0.73	0.67	11
6	0.58	0.59	0.59	81
7	0.71	0.64	0.68	135
accuracy			0.71	389
macro avg	0.71	0.74	0.73	389
weighted avg	0.72	0.71	0.72	389



Αποτελέσματα μοντέλων με επιλογή καλύτερων χαρακτηριστικών

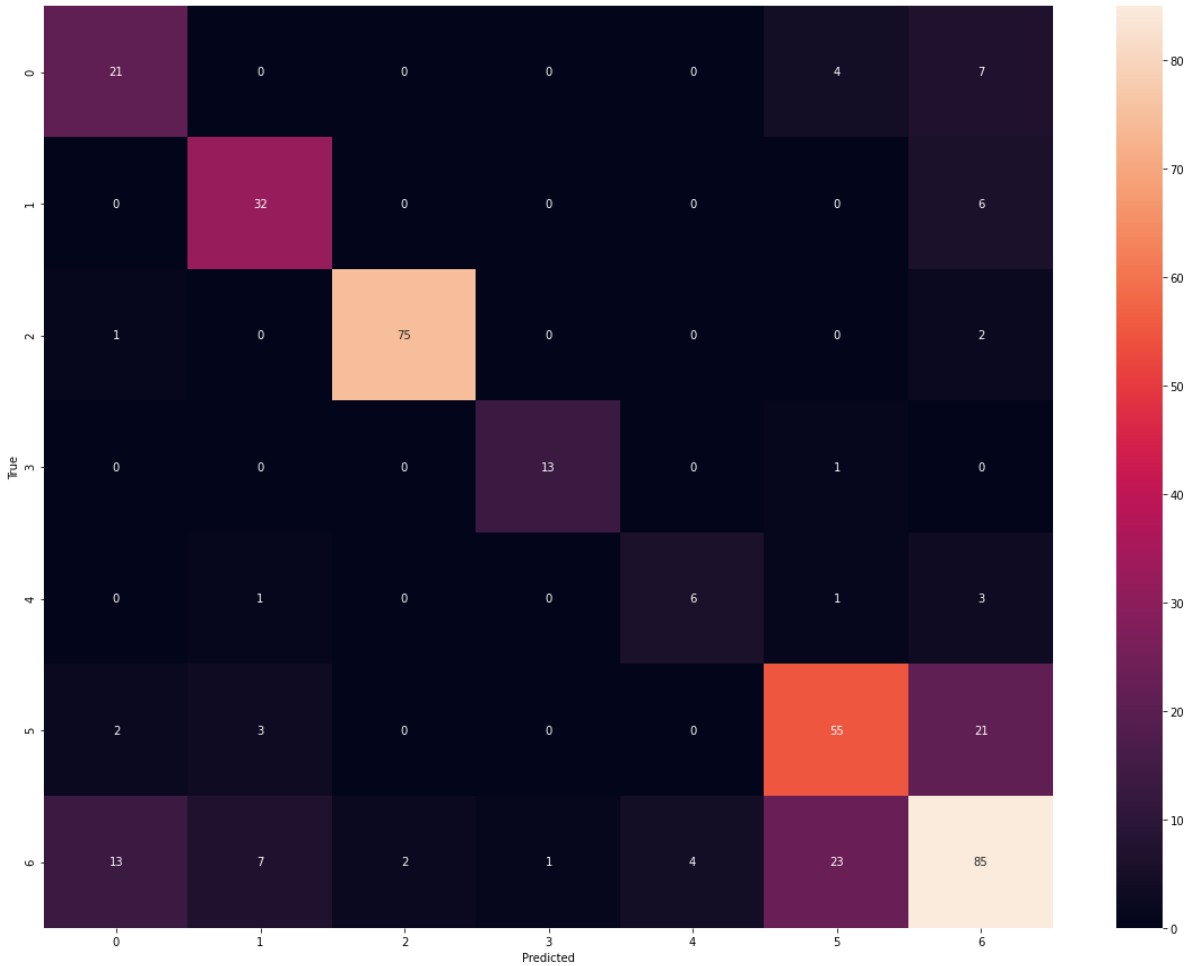
Random Forest – Feature Importances – Confusion Matrix (test set)

Classification Report:				
	precision	recall	f1-score	support
1	0.62	0.56	0.59	32
2	0.83	0.76	0.79	38
3	0.97	0.95	0.96	78
4	1.00	0.93	0.96	14
5	0.69	0.82	0.75	11
6	0.74	0.68	0.71	81
7	0.69	0.76	0.73	135
accuracy			0.77	389
macro avg	0.79	0.78	0.78	389
weighted avg	0.78	0.77	0.77	389



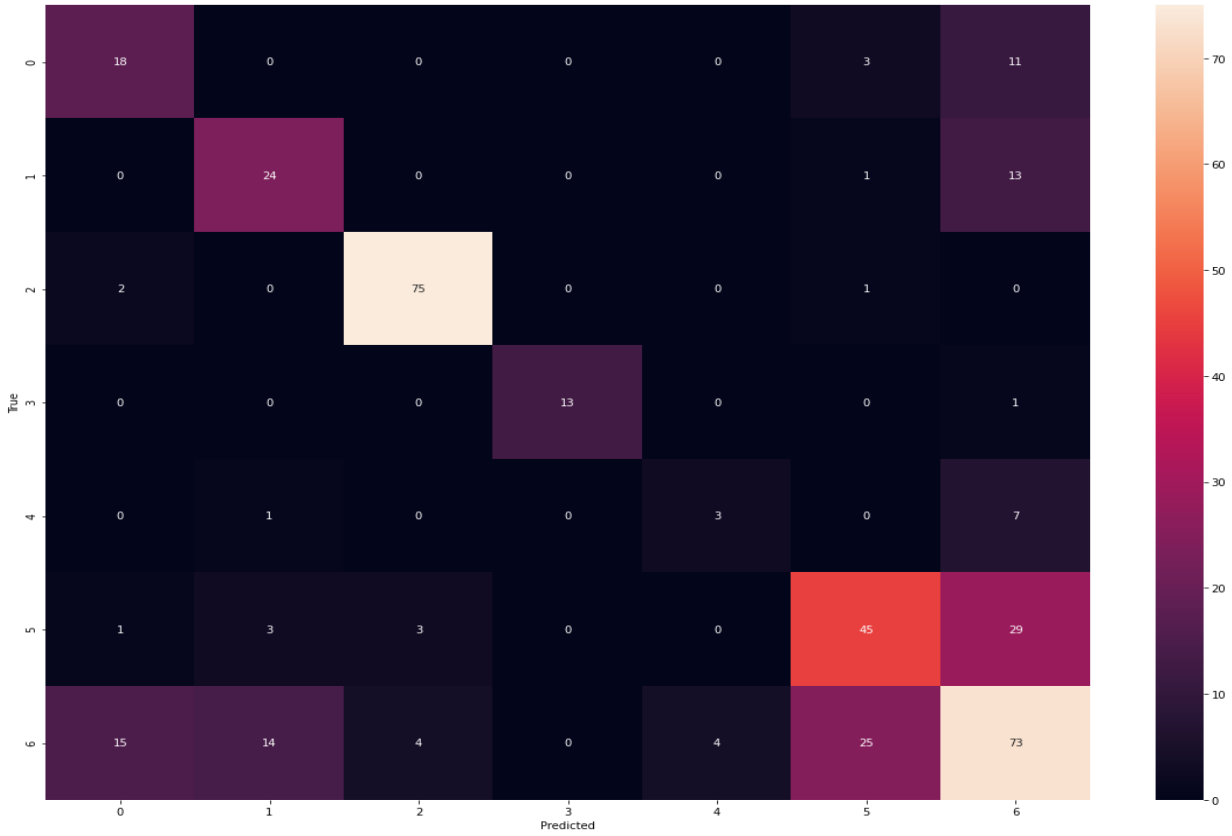
Support Vector Classifier – Feature Importances – Confusion Matrix (test set)

Classification Report:				
	precision	recall	f1-score	support
1	0.57	0.66	0.61	32
2	0.74	0.84	0.79	38
3	0.97	0.96	0.97	78
4	0.93	0.93	0.93	14
5	0.60	0.55	0.57	11
6	0.65	0.68	0.67	81
7	0.69	0.63	0.66	135
accuracy			0.74	389
macro avg	0.74	0.75	0.74	389
weighted avg	0.74	0.74	0.74	389



Logistic Regression – Feature Importances – Confusion Matrix (test set)

Classification Report:				
	precision	recall	f1-score	support
1	0.50	0.56	0.53	32
2	0.57	0.63	0.60	38
3	0.91	0.96	0.94	78
4	1.00	0.93	0.96	14
5	0.43	0.27	0.33	11
6	0.60	0.56	0.58	81
7	0.54	0.54	0.54	135
accuracy			0.65	389
macro avg	0.65	0.64	0.64	389
weighted avg	0.64	0.65	0.64	389



k-N-N – Feature Importances – Confusion Matrix (test set)

Classification Report:				
	precision	recall	f1-score	support
1	0.58	0.59	0.58	32
2	0.64	0.79	0.71	38
3	0.95	0.94	0.94	78
4	0.93	0.93	0.93	14
5	0.62	0.73	0.67	11
6	0.58	0.59	0.59	81
7	0.71	0.64	0.68	135
accuracy			0.71	389
macro avg	0.71	0.74	0.73	389
weighted avg	0.72	0.71	0.72	389