Εθνικο Μετσοβιο Πολυτεχνειο

Σχολη Εφαρμοσμενων Μαθηματικων και Φυσικων Επιστημων
Τομεας Μαθηματικων

Ταξινόμηση Αστρονομικών Παρατηρήσεων με τη βοήθεια μοντέλων Μηχανικής Μάθησης

Διπλωματικη Εργασια

Γεωργια Χριστοδουλου

Επιβλέπων: Δρ. Κωνσταντίνος Κουσουρής,
Επίκουρος Καθηγητής Ε.Μ.Π.

Αθήνα, Σεπτέμβριος 2019

Εθνικο Μετσοβιο Πολυτεχνειο

Σχολη Εφαρμοσμενων Μαθηματικων και Φυσικων Επιστημων
Τομεας Μαθηματικων

Ταξινόμηση Αστρονομικών Παρατηρήσεων με τη βοήθεια μοντέλων Μηχανικής Μάθησης

Διπλωματικη Εργασια

Γεωργια Χριστοδουλου

Επιβλέπων: Δρ. Κωνσταντίνος Κουσουρής,
Επίκουρος Καθηγητής Ε.Μ.Π.

Εγκρίθηκε από την τριμελή εξεταστική επιτροπή την 27η Σεπτεμβρίου 2019.

(Υπογραφή)	*(Υπογραφή)*	*(Υπογραφή)*
.....	.....	.....
Κωνσταντίνος Κουσουρής, Επίκουρος Καθηγητής Ε.Μ.Π.	Αικατερίνη Τζαμαριουδάκη, Ερευνήτρια	Γεώργιος Τσιπολίτης, Καθηγητής Ε.Μ.Π.

Αθήνα, Σεπτέμβριος 2019

(Υπογραφή)

.....

Γεωργια Χριστοδουλου
Διπλωματούχος Σχολής Εφαρμοσμένων Μαθηματικών και Φυσικών Επιστημών Ε.Μ.Π.
© 2019 - All rights reserved

Εθνικο Μετσοβιο Πολυτεχνειο

Σχολη Εφαρμοσμενων Μαθηματικων και Φυσικων Επιστημων
Τομεας Μαθηματικων

Copyright © - All rights reserved Γεωργία Χριστοδούλου, 2019
Με επιφύλαξη παντός δικαιώματος.

Απαγορεύεται η αντιγραφή, αποθήκευση και διανομή της παρούσας εργασίας, εξ ολοκλήρου ή τμήματος αυτής, για εμπορικό σκοπό. Επιτρέπεται η ανατύπωση, αποθήκευση και διανομή για σκοπό μη κερδοσκοπικό, εκπαιδευτικής ή ερευνητικής φύσης, υπό την προϋπόθεση να αναφέρεται η πηγή προέλευσης και να διατηρείται το παρόν μήνυμα. Ερωτήματα που αφορούν τη χρήση της εργασίας για κερδοσκοπικό σκοπό πρέπει να απευθύνονται προς τον συγγραφέα.

Ευχαριστίες

Θα ήθελα να εκφράσω ολόθερμες ευχαριστίες στον επιβλέποντα καθηγητή μου, Δρ. Κωνσταντίνο Κουσουρή, για την καθοδήγηση, την υποστήριξη και την εμπιστοσύνη που μου έδειξε κατά την εκπόνηση της διπλωματικής εργασίας. Η βοήθεια που μου προσέφερε ήταν απαραίτητη για την ολοκλήρωση της εργασίας.

Επίσης, θα ήθελα να ευχαριστήσω τον κάθε έναν που αποτέλεσε μέρος της φοιτητικής μου ζωής, και κυρίως την Κυβέλη και τη Ρωξάνη που δεν έφυγαν στιγμή από το πλευρό μου. Θερμές ευχαριστίες σε αυτούς που μου έμαθαν να στοχεύω για τ΄ αστέρια, διότι κανένα όνειρο δεν είναι ακατόρθωτο, φτάνει να υπάρχει τρέλα, θέληση, επιμονή και υπομονή. Γιατί το μέλλον ανήκει στους τολμηρούς.

Η εργασία είναι αφιερωμένη στην οικογένεια μου για την αμέριστη υποστήριξη που μου προσφέρει για κάθε μικρό και μεγάλο στόχο που θέτω.

Περίληψη

Η εξέλιξη της επιστήμης των δεδομένων (Data Science) και η διαρκώς αυξανόμενη συλλογή δεδομένων, δημιούργησε την επιτακτική ανάγκη της ανέλιξης του κλάδου της Τεχνητής Νοημοσύνης. Τα μαζικά δεδομένα (Big Data) θα παρέμεναν ανεκμετάλλευτα εάν ήταν αδύνατη η αξιοποίησή τους για την άντληση πληροφοριών και την εξαγωγή συμπερασμάτων.

Η Τεχνητή Νοημοσύνη αποτελεί έναν από τους πιο ραγδαία αναπτυσσόμενους τομείς της Πληροφορικής. Η πληθώρα των εφαρμογών της βοηθάει τόσο τον άνθρωπο στην καθημερινότητά του όσο και την επιστημονική κοινότητα να εξελίσσεται. Οι δυνατότητες φαίνεται ότι είναι απεριόριστες και η ανθρωπότητα ασταμάτητη. Σύμφωνα με τον Pedro Domingos, *"Machine learning is remaking science, technology, business, politics and war. Satellites, DNA sequencers, and particle accelerators probe nature in ever-finer detail, and learning algorithms turn the torrents of data into new scientific knowledge."* [1].

Ένα παράδειγμα αξιοποίησης τεράστιου όγκου δεδομένου για επιστημονική μελέτη και τεκμηρίωση αποτελεί η Sloan Digital Sky Survey (SDSS). Τα δεδομένα που συνέλεξε η SDSS περιέχουν πληροφορίες για μερικά εκατομμύρια αστρονομικά αντικείμενα. Επειδή η ένα-προς-ένα μελέτη των ουράνιων αντικειμένων ήταν αδύνατη, οι επιστήμονες διεξήγαγαν το project SKICAT (SKy Image Cataloging and Analysis Tool), όπου ανέπτυξαν και εκπαίδευσαν αλγόριθμους Μηχανικής Μάθησης με τη βοήθεια εικόνων που απεικόνιζαν αστρονομικά αντικείμενα, των οποίων η φύση τους ήταν γνωστή. Οι αλγόριθμοι μετά την εκπαίδευση ήταν σε θέση να εντοπίζουν και να ταξινομούν αστρονομικά αντικείμενα που απεικονίζονταν σε εικόνες άγνωστου περιεχομένου, με μεγάλη επιτυχία. Έτσι, οι επιστήμονες μπορούσαν να αφοσιωθούν σε πιο σημαντικές δραστηριότητες όπως η μελέτη των ουράνιων σωμάτων, υποβοηθούμενοι και πάλι από εργαλεία της Τεχνητής Νοημοσύνης. Η SDSS προσέφερε στους αστροφυσικούς νέες δυνατότητες για την μελέτη του σύμπαντος, χωρίς περιορισμούς.

Ωστόσο, το project SKICAT ασχολήθηκε με την ταξινόμηση εικόνων. Στην παρούσα εργασία ασχοληθήκαμε με τα αριθμητικά δεδομένα της SDSS. Δηλαδή αναπτύξαμε και εκπαιδεύσαμε μοντέλα Μηχανικής Μάθησης που ταξινομούν αριθμητικά διανύσματα σε τρεις κατηγορίες: Αστέρες, Γαλαξίες και Κβάζαρς. Στην εργασία μελετάμε τι μπορούμε να κάνουμε με 5 μεταβλητές απόκρισης φίλτρων συν τη μεταβλητή που περιγράφει τη μετατόπιση προς το ερυθρό (redshift). Δηλαδή με το σκεπτικό ότι τα διάφορα αστρονομικά αντικείμενα έχουν διαφορετικό αποτύπωμα στο Ηλεκτρομαγνητικό φάσμα, μελετάμε το πόσο ισχυροί μπορούν να είναι οι

ταξινομητές που εκπαιδεύουμε.

Αρχικά, εφαρμόσαμε δύο διαφορετικές μεθόδους Linear Discriminant Analysis στα δεδομένα μας για να μελετήσουμε την απόδοση των απλούστερων μεθόδων ταξινόμησης πριν προχωρήσουμε στην εκπαίδευση άλλων μη γραμμικών μεθόδων, και άρα πιο σύνθετων. Συγκεκριμένα, ασχοληθήκαμε με την κατασκευή και την εκπαίδευση Νευρωνικών Δικτύων και Ενδυναμωμένων Δέντρων Απόφασης. Καταγράψαμε και μελετήσαμε την απόδοσή τους για να συγκρίνουμε τα αποτελέσματα και να καταλήξουμε στο μοντέλο που επιλύει το πρόβλημα της ταξινόμησης με τα λιγότερα σφάλματα.

Εικόνα από τη βάση δεδομένων του SDSS

Λέξεις κλειδιά

Μηχανική Μάθηση, Τεχνητή Νοημοσύνη, Sloan Digital Sky Survey, αστρονομικά αντικείμενα, ταξινόμηση, γραμμικοί ταξινομητές, μη γραμμικοί ταξινομητές, Linear Discriminant Analysis, Νευρωνικά Δίκτυα, Ενδυναμωμένα Δέντρα Απόφασης

Περιεχόμενα

Ευχαριστίες **4**

Περίληψη **6**

Περιεχόμενα **10**

1 Περιγραφή του προβλήματος και ανάλυση των δεδομένων **12**
1.1 Sloan Digital Sky Survey 12
1.2 Σύνολο δεδομένων 13
1.3 Περιγραφή του προβλήματος 14
1.4 Μελέτη των μεταβλητών 15
1.5 Επεξήγηση των μεταβλητών 16
1.6 Διαχωριστική ικανότητα μεταβλητών 17
1.7 Συσχέτιση μεταβλητών 21
1.8 Τυποποίηση δεδομένων 27

2 Εισαγωγή στη Μηχανική Μάθηση **28**
2.1 Ορισμός 28
2.2 Είδη Μηχανικής Μάθησης 28
2.3 Εφαρμογές Μηχανικής Μάθησης 29
2.4 Μαζικά Δεδομένα (Big Data) 30

3 Linear Discriminant Analysis **31**
3.1 Curse of dimensionality 31
3.2 Εισαγωγή στην Linear Discriminant Analysis 32
3.3 Θεωρία LDA 33
3.4 Μέτρα αξιολόγησης της LDA 35
3.5 Μέθοδος 1:Μέθοδος Fisher 37
3.5.1 Αλγόριθμος Fisher 41
3.5.2 Υλοποίηση 42
3.5.3 Εκτέλεση και Αποτελέσματα 43

3.5.4 Εναλλακτικός τρόπος ταξινόμησης 48
3.5.5 Αξιολόγηση των αποτελεσμάτων της μεθόδου Fisher 51
3.6 Μέθοδος 2: LDA+SVD 55
3.6.1 Θεωρία LDA+SVD 55
3.6.2 Υλοποίηση 56
3.6.3 Εκτέλεση και Αποτελέσματα 57
3.6.4 Εναλλακτικός τρόπος ταξινόμησης 64
3.6.5 Αξιολόγηση των αποτελεσμάτων της μεθόδου LDA+SVD 66
3.7 Σύγκριση αποτελεσμάτων μεθόδων LDA 71

4 Νευρωνικά Δίκτυα 72
4.1 Αλγόριθμος 73
4.2 Υλοποίηση 74
4.2.1 Αρχιτεκτονική 76
4.2.2 Υπερεκπαίδευση 76
4.3 Εκτέλεση και Αποτελέσματα 78
4.3.1 Ένα Νευρωνικό Δίκτυο με ένα κρυφό στρώμα 6 κόμβων 78
4.3.2 Δύο Νευρωνικά Δίκτυα με ένα κρυφό στρώμα 6 κόμβων το καθένα . . 83
4.3.3 Συνδυασμός ενός Νευρωνικού δικτύου με ένα κρυφό στρώμα 6 κόμβων και συνθήκης που βασίζεται στην τιμή του redshift 89
4.4 Σύγκριση αποτελεσμάτων αλγορίθμων 92

5 Ενδυναμωμένα Δέντρα Απόφασης 93
5.1 Αλγόριθμος 94
5.2 Υλοποίηση 99
5.3 Εκτέλεση και Αποτελέσματα 101
5.4 Αξιολόγηση Αλγορίθμου 110

6 Σύγκριση αποτελεσμάτων 112

Συντομογραφίες 116

Αναφορές 118

Πηγές 119

Κεφάλαιο 1

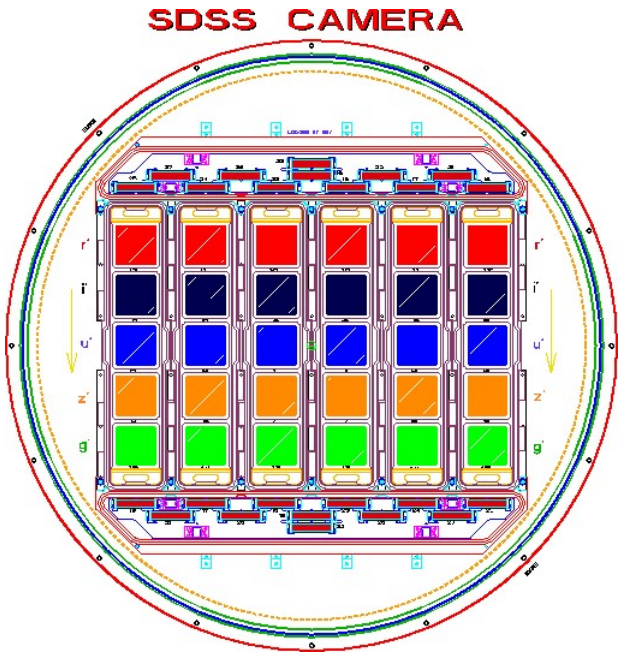
Περιγραφή του προβλήματος και ανάλυση των δεδομένων

1.1 Sloan Digital Sky Survey

Η πηγή των δεδομένων που χρησιμοποιήσαμε είναι η Sloan Digital Sky Survey (SDSS, `www.sdss.org`), η οποία αποτελεί την μεγαλύτερη έρευνα παρατήρησης και μελέτης του ουρανού. Ξεκίνησε τη συλλογή δεδομένων με τη βοήθεια οπτικού τηλεσκοπίου το 1996 και η δραστηριότητά της θα συνεχίσει για αρκετά χρόνια ακόμη. Ανάμεσα σε άλλα επιτεύγματα, κατάφερε να δημιουργήσει λεπτομερείς τρισδιάστατους χάρτες του σύμπαντος με πολύχρωμες εικόνες που καλύπτουν το 1/3 του ουρανού.

Το τηλεσκόπιο της SDSS αποτελείται από δύο κύρια όργανα: Μια φωτογραφική μηχανή απεικόνισης (imaging camera) η οποία μπορεί να φωτογραφήσει μεγάλο μέρος του ουρανού σε μια λήψη, και ένα φασματογράφο (spectrograph) ο οποίος μετρά το φάσμα των ουράνιων αντικειμένων, οπότε μπορούν να προσδιοριστούν οι αποστάσεις μεταξύ τους.

Η φωτογραφική μηχανή της SDSS συλλέγει φωτομετρικά δεδομένα χρησιμοποιώντας 6 στήλες από 5 ανιχνευτές CCDs (Charge Coupled Devices) στην κάθε μια. Κάθε στήλη αποτελείται από τα φίλτρα u, g, r, i, z, όπου το καθένα συλλαμβάνει ακτινοβολία διαφορετικού μήκους κύματος. Συνδυάζοντας τα αποτελέσματα των φίλτρων μπορεί να καθοριστεί το πραγματικό χρώμα των ουράνιων αντικειμένων. Πιο κάτω απεικονίζεται μια σχηματική αναπαράσταση της φωτογραφικής μηχανής:



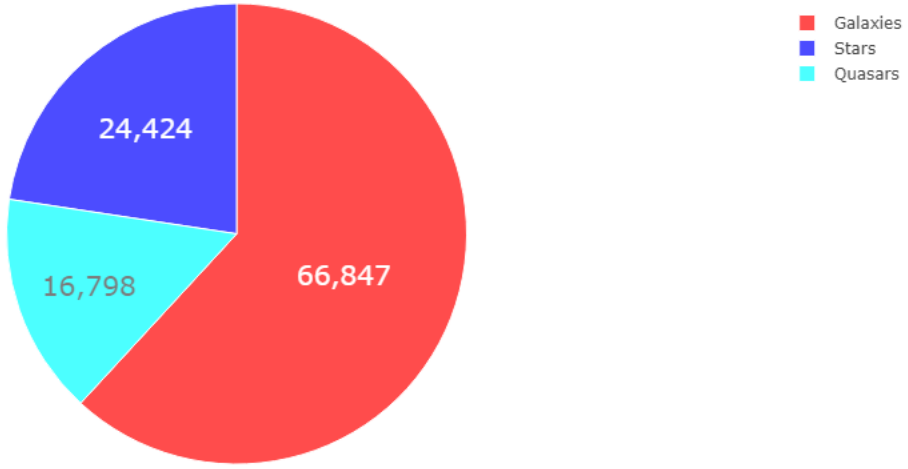
Σχήμα 1.1: Πηγή εικόνας: `https://www.sdss.org/dr12/imaging/imaging_basics/`

1.2 Σύνολο δεδομένων

Η SDSS ανά τακτά χρονικά διαστήματα δημοσιεύει τα δεδομένα (εικόνες, αριθμητικά δεδομένα κτλ) που συνέλεξε προς το κοινό ούτως ώστε ο κάθε ερασιτέχνης ή επαγγελματίας να μπορεί να έχει πρόσβαση.

Μέσω της πλατφόρμας Kaggle (`www.kaggle.com`) εντοπίσαμε ένα σύνολο δεδομένων της SDSS, το οποίο περιείχε 108.070 αστρονομικές παρατηρήσεις. Οι παρατηρήσεις ανήκουν σε τρεις κλάσεις: **Αστέρες, Γαλαξίες και Κβάζαρς** [1]. Η κάθε παρατήρηση είναι ένα διάνυσμα αριθμητικών τιμών που αποτελείται από 17 χαρακτηριστικές μεταβλητές και μία επιπλέον που καθορίζει την κλάση του διανύσματος. Παρακάτω απεικονίζεται ο αριθμός των διανυσμάτων που διαθέτουμε από κάθε κλάση:

[1] Ως κβάζαρ (quasar) χαρακτηρίζεται κάθε εξαιρετικά λαμπρός και μακρινός ενεργός πυρήνας γαλαξία. Αποτελούν τα πιο λαμπερά σώματα του σύμπαντος και γι' αυτό το λόγο είναι ορατά από μεγάλες αποστάσεις. Το όνομά τους προέκυψε από το quasi-stellar (παρόμοιος με αστέρα), αφού στο ορατό φως εμφανίζονται ως σημειακές πηγές (σαν αστέρες), παρά ως εκτεταμένα σώματα (όπως οι γαλαξίες). Εκπέμπουν τεράστια ποσά ενέργειας τα οποία ισοδυναμούν με την ενέργεια ενός τρισεκατομμυρίων ήλιων, περίπου. Μερικά κβάζαρ πιστεύεται ότι παράγουν από δέκα έως εκατό φορές περισσότερη ενέργεια από ότι ολόκληρος ο γαλαξίας μας.

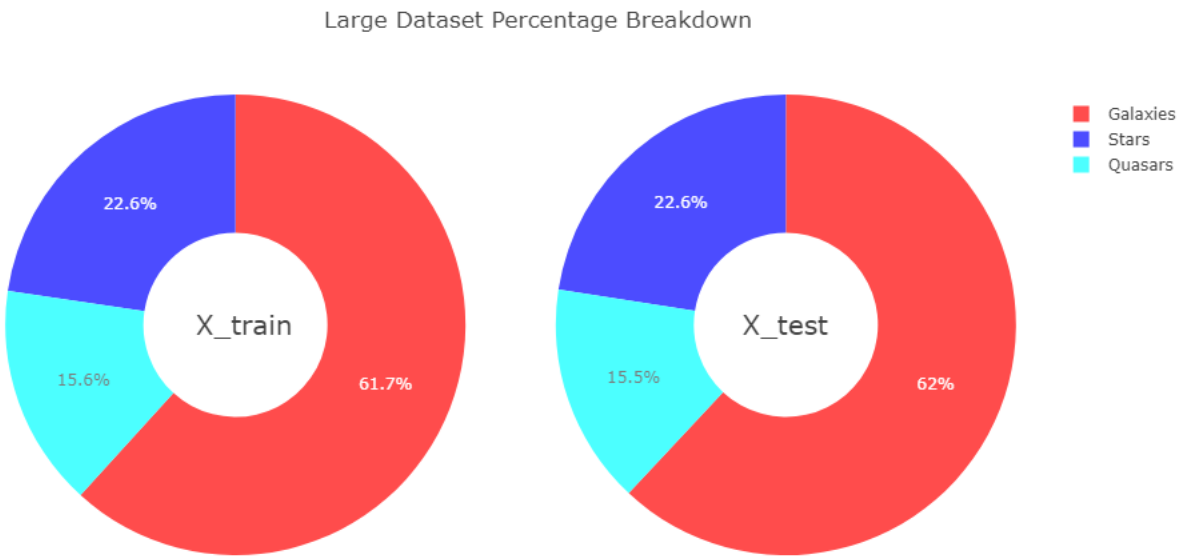


1.3 Περιγραφή του προβλήματος

Χρησιμοποιώντας τα δεδομένα που έχουμε στη διάθεσή μας, θα κατασκευάσουμε και στη συνέχεια θα εκπαιδεύσουμε διάφορα μοντέλα Μηχανικής Μάθησης. Τα μοντέλα θα λειτουργούν ως ταξινομητές, δηλαδή θα κατατάσσουν τα δεδομένα εισόδου σε κάποια από τις τρεις κλάσεις.

Θα ξεκινήσουμε τις δοκιμές με την απλούστερη μορφή ταξινομητή, που είναι ο γραμμικός ταξινομητής. Συγκεκριμένα θα εκτελέσουμε τη μέθοδο ταξινόμησης του **Fisher** και στη συνέχεια τη μέθοδο **Linear Discriminant Analysis μέσω της μεθόδου SVD**. Ακολουθως, θα ασχοληθούμε με πιο σύνθετα μοντέλα ταξινόμησης, όπως **Νευρωνικά Δίκτυα** και **Ενδυναμωμένα Δέντρα Απόφασης**, τα οποία έχουν περισσότερες δυνατότητες. Στο τέλος της υλοποίησης και της εκτέλεσης των μοντέλων αυτών, θα συγκρίνουμε τα αποτελέσματά τους για να καταλήξουμε σε αυτό που έχει την καλύτερη απόδοση, δηλαδή αυτό με τα χαμηλότερα σφάλματα ταξινόμησης.

Έχουμε στη διάθεσή μας 108.070 παρατηρήσεις, οπότε διαθέτουμε 54.035 παρατηρήσεις για εκπαίδευση (σύνολο X train) και 54.035 για αξιολόγηση (σύνολο X test) των μοντέλων μηχανικής μάθησης.



1.4 Μελέτη των μεταβλητών

Μελετώντας τις μεταβλητές των δεδομένων παρατηρήσαμε ότι ορισμένες περιέγραφαν την εκάστοτε (φωτογραφική) λήψη των αστρονομικών αντικειμένων. Οι επιστήμονες επέλεξαν να τις συμπεριλάβουν ως μεταβλητές για να είναι εύκολη η εύρεσή τους από τη βάση δεδομένων (αλλά και για πληρότητα πληροφοριών αφού ορισμένες σχετίζονταν με τον τρόπο και το χρόνο λήψης). Επειδή δεν προσφέρουν κάποια πληροφορία που θα μπορούσε να βοηθήσει τα μοντέλα μηχανικής μάθησης να ταξινομήσουν τα αστρονομικά αντικείμενα στις 3 κλάσεις, αποφασίσαμε να τις αφαιρέσουμε. Έτσι, οι εναπομείναντες μεταβλητές του δείγματος είναι οι **αποκρίσεις των φίλτρων** $\mathbf{u}, \mathbf{g}, \mathbf{r}, \mathbf{i}$ και $\mathbf{z}$ και η μεταβλητή που περιγράφει τη **μετατόπιση προς το ερυθρό** (**redshift**) . Μετά την αφαίρεση, τα δεδομένα μας παίρνουν την παρακάτω μορφή:

	u	g	r	i	z	redshift
0	19.19619	17.83329	17.52225	17.40237	17.35182	-0.000220
1	21.65541	19.13715	17.92577	17.44741	17.15818	-0.000008
2	20.70867	19.20954	18.55966	18.24395	18.10117	0.000096
3	22.88806	21.20900	19.90560	19.33555	19.08966	-0.000247
4	17.82932	16.11081	15.39808	15.13612	15.00507	-0.000131

Σχήμα 1.2: Ένα μικρό δείγμα του συνόλου δεδομένων

Επίσης εξετάσαμε για την ύπαρξη ακραίων τιμών (outliers), όπου εντοπίσαμε μία μόνο, οπότε αφαιρέσαμε το διάνυσμα που την περιείχε από το δείγμα. Συνολικά έχουμε **108.069 παρατηρήσεις** που η κάθε μία περιέχει μετρήσεις **6 μεταβλητών** συν μίας που περιγράφει την πραγματική τιμή της κλάσης.

1.5 Επεξήγηση των μεταβλητών

Αποκρίσεις των φίλτρων u, g, r, i, z

Το χρώμα κάθε αστρονομικού αντικειμένου σχετίζεται με το μήκος κύματος του φωτός που εκπέμπει. Κάθε φίλτρο που τοποθετείται στους ανιχνευτές CCDs συλλαμβάνει φως συγκεκριμένου εύρους συχνοτήτων. Συγκεκριμένα, το σύστημα φωτομετρίας του SDSS χρησιμοποιεί τα φίλτρα u, g, r, i, z. Το φίλτρο u συλλαμβάνει την υπεριώδη (ultraviolet) ακτινοβολία, το g την πράσινη (green), το r την κόκκινη (red), το i την ακτινοβολία με μήκος κύματος κοντά στο υπέρυθρο (near infrared) και το z την υπέρυθρη ακτινοβολία (infrared). Συνδυάζοντας τις αποκρίσεις των 5 φίλτρων, καθορίζεται το πραγματικό χρώμα των ουράνιων σωμάτων.

Μετατόπιση προς το ερυθρό (redshift)

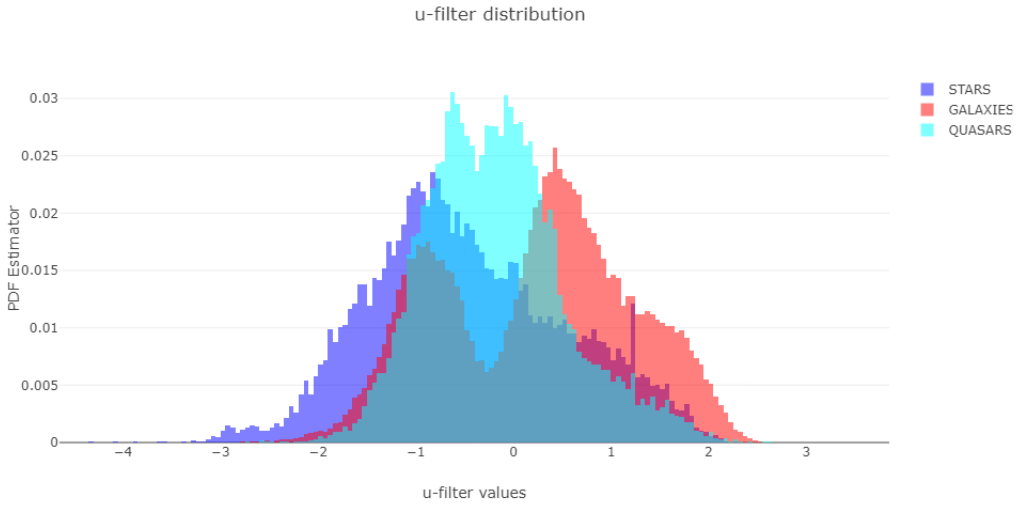
Η μετατόπιση προς το ερυθρό συμβαίνει όταν η ηλεκτρομαγνητική ακτινοβολία (στη συγκεκριμένη περίπτωση το φως) από μία πηγή, αυξάνει σε μήκος κύματος (άρα μειώνεται η συχνότητα) ή ισοδύναμα μετατοπίζεται προς το ερυθρό φάσμα της ηλεκτρομαγνητικής ακτινοβολίας. Η μετατόπιση προς το ερυθρό εμφανίζεται όταν η πηγή απομακρύνεται από τον παρατηρητή. Σύμφωνα με τον νόμο του Hubble, λόγω της διαστολής του σύμπαντος για πολύ μακρινές πηγές φωτός (μερικά εκατομμύρια έτη φωτός μακριά) ισχύει ότι η μετατόπιση προς το ερυθρό σχετίζεται με την απόσταση τους από τη Γη. Με άλλα λόγια το σύμπαν διαστέλλεται, αλλά όχι με σταθερό ρυθμό αφού τα αντικείμενα που βρίσκονται σε μεγάλη απόσταση από τη Γη, διαστέλλονται γρηγορότερα, οπότε εμφανίζουν υψηλές τιμές μετατόπισης προς το ερυθρό, όπως συμβαίνει για παράδειγμα με τα Κβάζαρς.

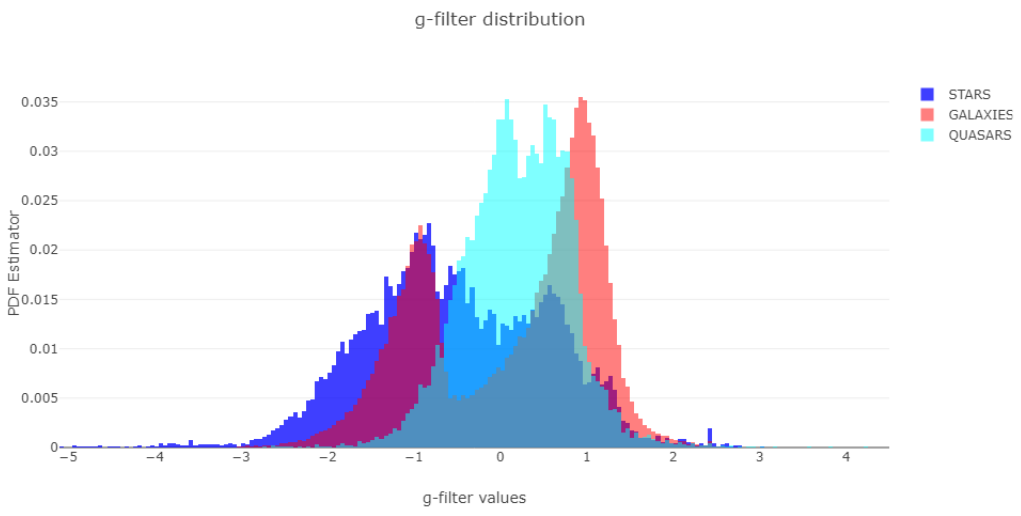
1.6 Διαχωριστική ικανότητα μεταβλητών

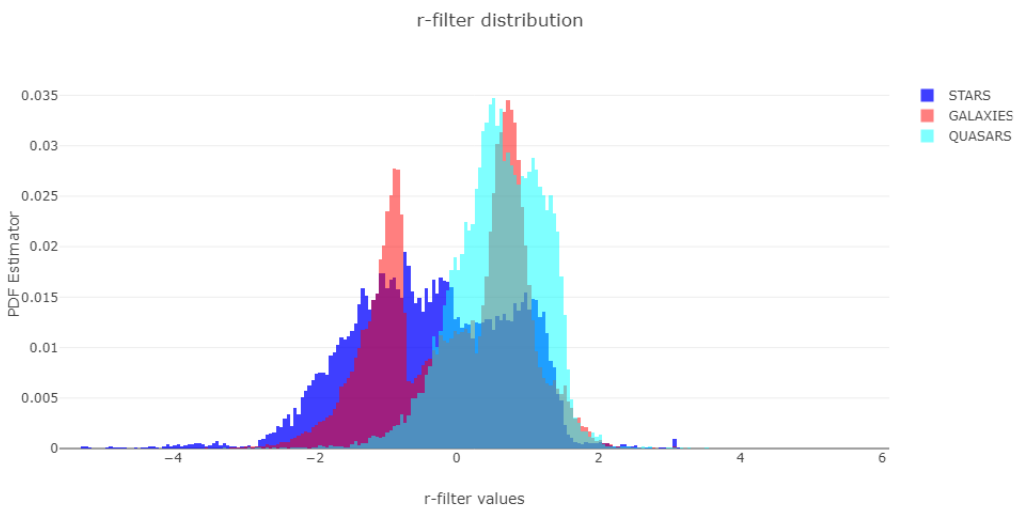
Για να μελετίσουμε την διαχωριστική ικανότητα των μεταβλητών, δηλαδή κατά πόσο θα μπορούσαν να συμβάλουν στην ταξινόμηση των παρατηρήσεων στις 3 κλάσεις, χωρίζουμε τα δεδομένα στις κλάσεις, για να σχεδιάσουμε γραφικά τις μονοδιάστατες προβολές της εκάστοτε μεταβλητής (όλων των κλάσεων σε κοινό διάγραμμα) ούτως ώστε να εξετάσουμε τόσο την κατανομή που ακολουθούν, όσο και την ενδεχόμενη επικάλυψη που έχουν οι τιμές της μεταβλητής της μιας κλάσης, με των υπολοίπων.

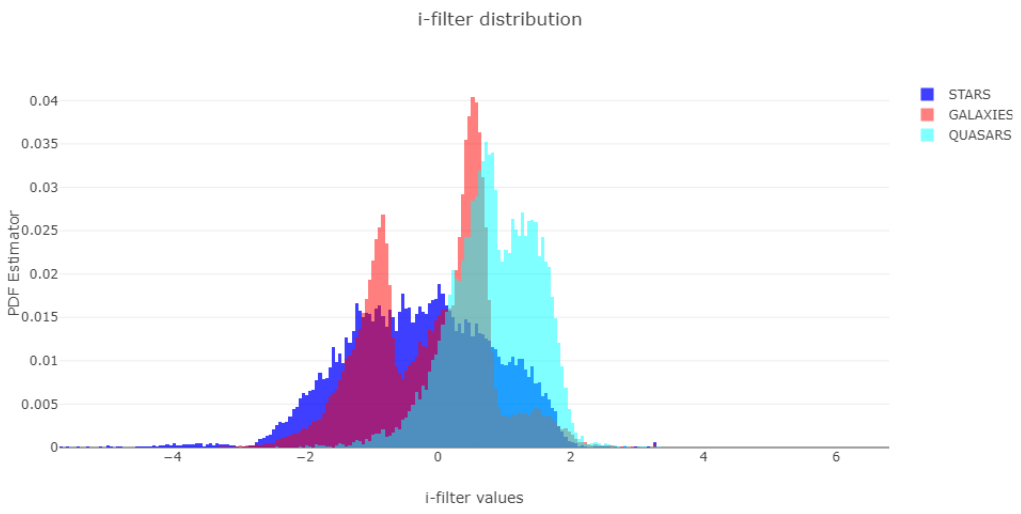
Πριν προχωρήσουμε στην κατασκευή των ιστογραμμάτων, τυποποιούμε τα δεδομένα που διαθέτουμε, αφαιρώντας από τις αρχικές τιμές την μέση τιμή της μεταβλητής και διαιρώντας με την τυπική απόκλιση. Έτσι, όλες οι μεταβλητές έχουν την ίδια μέση τιμή $\mu = 0$ και ίδια τυπική απόκλιση $\sigma = 1$.

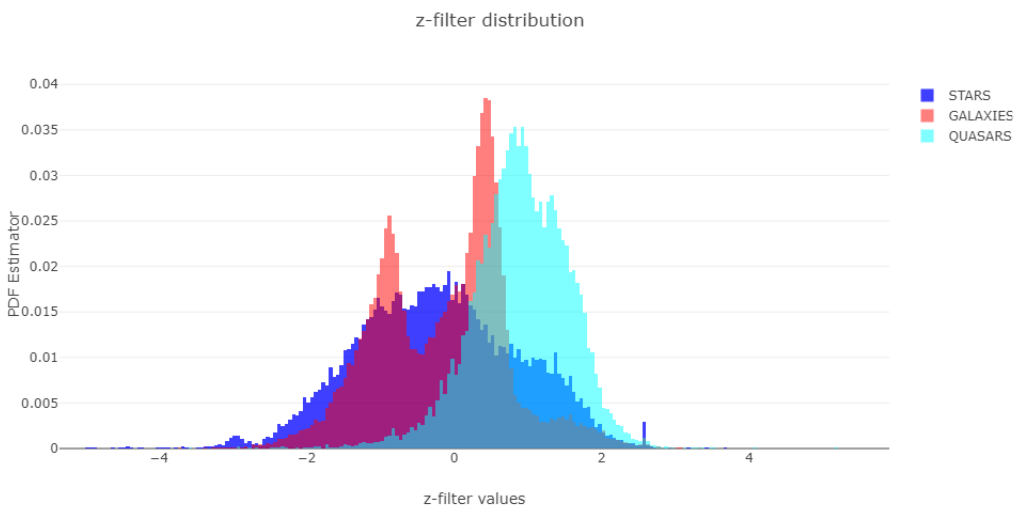
Τα ιστογράμματα απεικονίζονται παρακάτω:

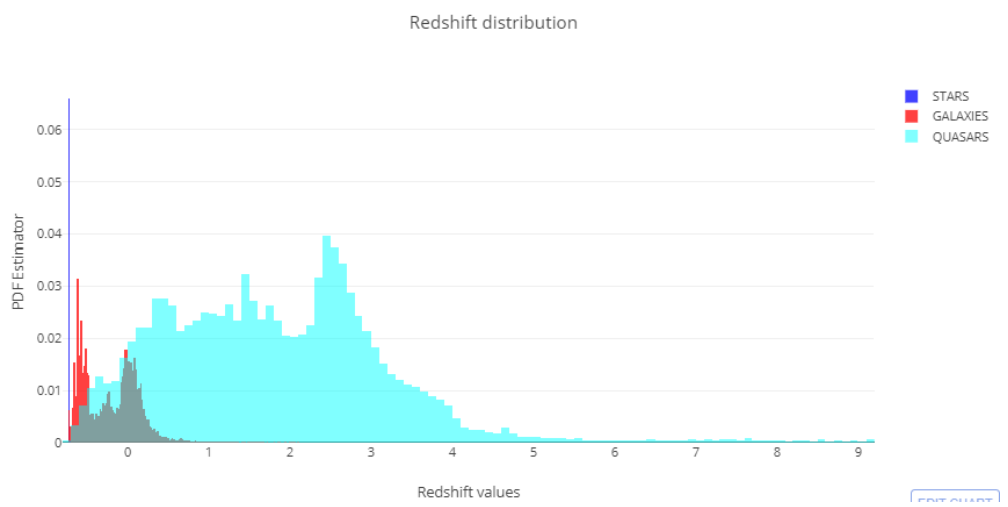



g-filter distribution
STARS
GALAXIES
QUASARS
PDF Estimator
0.035
0.03
0.025
0.02
0.015
0.01
0.005
0
−5
−4
−3
−2
−1
0
1
2
3
4
g-filter values

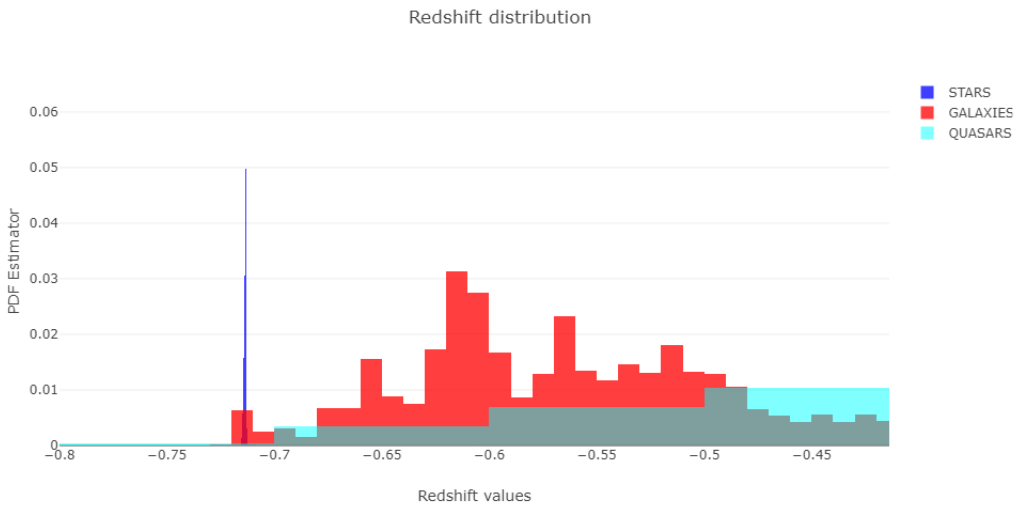

r-filter distribution
STARS
GALAXIES
QUASARS
PDF Estimator
0.035
0.03
0.025
0.02
0.015
0.01
0.005
0
−4
−2
0
2
4
6
r-filter values

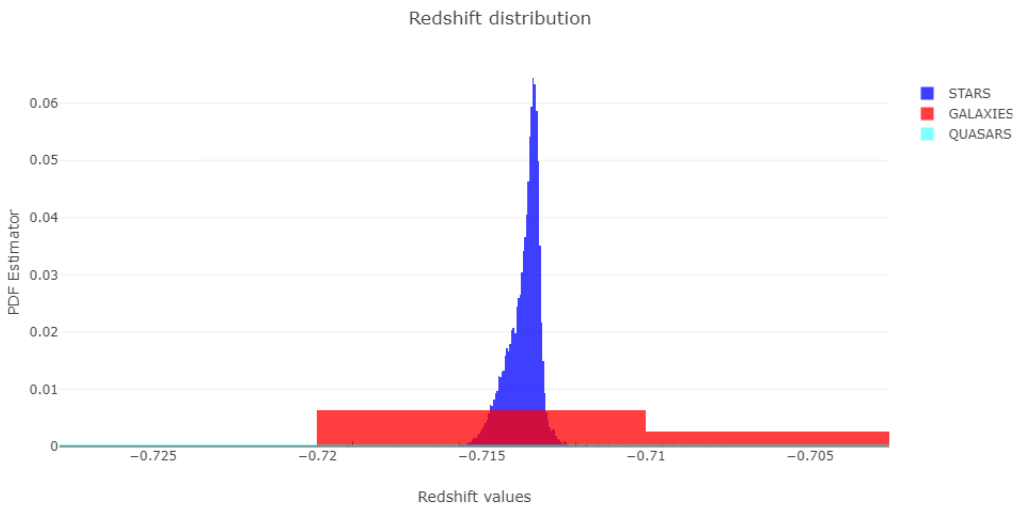

i-filter distribution
STARS
GALAXIES
QUASARS
PDF Estimator
0.04
0.035
0.03
0.025
0.02
0.015
0.01
0.005
0
−4
−2
0
2
4
6
i-filter values


z-filter distribution
STARS
GALAXIES
QUASARS
PDF Estimator
0.04
0.035
0.03
0.025
0.02
0.015
0.01
0.005
0
−4
−2
0
2
4
z-filter values


Redshift distribution
STARS
GALAXIES
QUASARS
PDF Estimator
0.06
0.05
0.04
0.03
0.02
0.01
0
0
1
2
3
4
5
6
7
8
9
Redshift values

Μεγεθύνοντας το τελευταίο διάγραμμα στην περιοχή γύρω από το 0 έχουμε:





Παρατηρούμε ότι οι κατανομές όλων των μεταβλητών επικαλύπτονται, δηλαδή καμία μεταβλητή δεν έχει καλή διαχωριστική ικανότητα. Ωστόσο, στο τελευταίο διάγραμμα παρατηρούμε ότι η κατανομή των τιμών redshift των Αστέρων έχει περιορισμένο εύρος, και μικρή επικάλυψη με τις τιμές των Γαλαξίων, κάτι που ίσως βοηθήσει στην εκπαίδευση των μοντέλων. Γενικά, όμως, συμπεραίνουμε ότι για να εκπαιδεύσουμε τα μοντέλα μηχανικής μάθησης, είναι απαραίτητο να συνδυάσουμε την πληροφορία που μας προσφέρουν όλες οι μεταβλητές.

1.7 Συσχέτιση μεταβλητών

Προτού εργαστούμε με τα μοντέλα μηχανικής μάθησης, μελετούμε εάν οι μεταβλητές είναι συσχετισμένες. Αρχικά κατασκευάζουμε τον πίνακα συσχετίσεων ο οποίος περιλαμβάνει τους συντελεστές γραμμικής συσχέτισης για όλες τις παρατηρήσεις ενώ στη συνέχεια κατασκευάζουμε τον πίνακα συσχετίσεων για τις παρατηρήσεις της κάθε κλάσης ξεχωριστά.

Σχήμα 1.3: Πίνακας συσχετίσεων ολόκληρου του δείγματος

	u	**g**	**r**	**i**	**z**	**redshift**
u	1.0	0.88	0.76	0.66	0.6	0.2
g	0.88	1.0	0.94	0.86	0.81	0.33
r	0.76	0.94	1.0	0.97	0.93	0.44
i	0.66	0.86	0.97	1.0	0.98	0.49
z	0.6	0.81	0.93	0.98	1.0	0.5
redshift	0.2	0.33	0.44	0.49	0.5	1.0

Σχήμα 1.4: Πίνακας συσχετίσεων Αστέρων

	u	**g**	**r**	**i**	**z**	**redshift**
u	1.0	0.93	0.85	0.75	0.67	0.069
g	0.93	1.0	0.96	0.88	0.81	0.034
r	0.85	0.96	1.0	0.96	0.9	-0.019
i	0.75	0.88	0.96	1.0	0.97	-0.069
z	0.67	0.81	0.9	0.97	1.0	-0.09
redshift	0.069	0.034	-0.019	-0.069	-0.09	1.0

Σχήμα 1.5: Πίνακας συσχετίσεων Γαλαξιών

	u	g	r	i	z	redshift
u	1.0	0.87	0.8	0.76	0.73	0.73
g	0.87	1.0	0.97	0.93	0.9	0.87
r	0.8	0.97	1.0	0.98	0.97	0.87
i	0.76	0.93	0.98	1.0	0.98	0.83
z	0.73	0.9	0.97	0.98	1.0	0.8
redshift	0.73	0.87	0.87	0.83	0.8	1.0

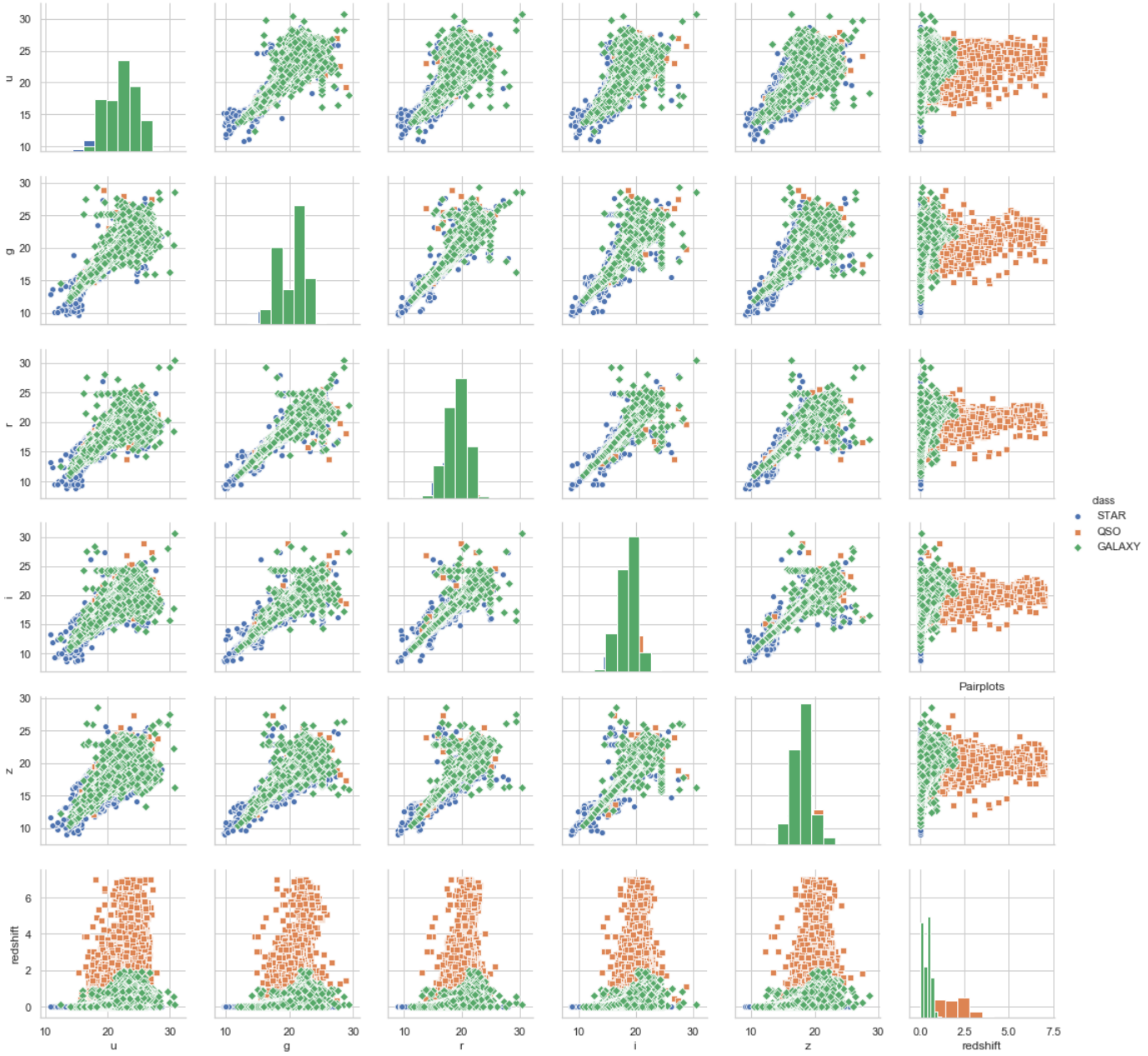
Σχήμα 1.6: Πίνακας συσχετίσεων Κβάζαρς

	u	g	r	i	z	redshift
u	1.0	0.86	0.77	0.71	0.68	0.37
g	0.86	1.0	0.93	0.87	0.82	0.25
r	0.77	0.93	1.0	0.96	0.92	0.27
i	0.71	0.87	0.96	1.0	0.95	0.3
z	0.68	0.82	0.92	0.95	1.0	0.3
redshift	0.37	0.25	0.27	0.3	0.3	1.0

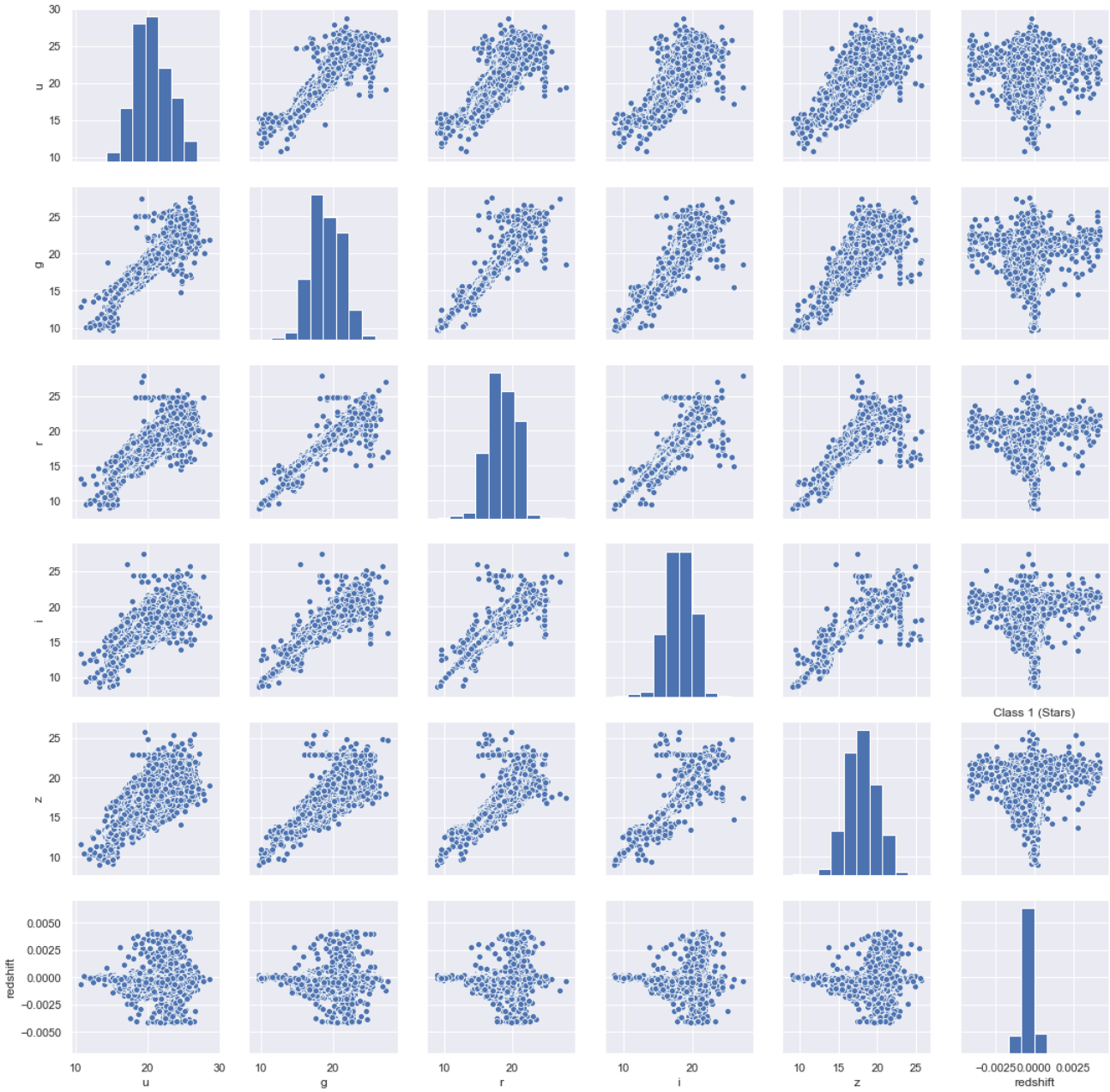
Στον γενικό πίνακα συσχέτισης (που αφορά όλες τις παρατηρήσεις) παρατηρούμε ότι υπάρχουν έντονες συσχετίσεις μεταξύ των μεταβλητών u, g, r, i, z (ανά δύο) της τάξης του 60-98%, ενώ οι συσχετίσεις των u, g, r, i, z με το redshift είναι χαμηλότερες. Τα ίδια ισχύουν και για την κλάση των Κβάζαρς. Στην κλάση των Αστέρων υπάρχουν υψηλές συσχετίσεις μεταξύ των u, g, r, i, z ενώ το redshift φαίνεται να μην συσχετίζεται με αυτές. Στην κλάση των Γαλαξιών παρουσιάζονται έντονες συσχετίσεις μεταξύ όλων των μεταβλητών από 73% μέχρι 98%.

Προκειμένου να βεβαιωθούμε για τα πιο πάνω συμπεράσματα, προβάλλουμε το χώρο των χαρακτηριστικών σε όλα τα διδιάστατα επίπεδα $x_i - x_j$. Το πρώτο διάγραμμα περιλαμβάνει κοινά διαγράμματα για όλες τις κλάσεις ενώ τα ακόλουθα 3 αφορούν την κάθε κλάση ξεχωριστά.

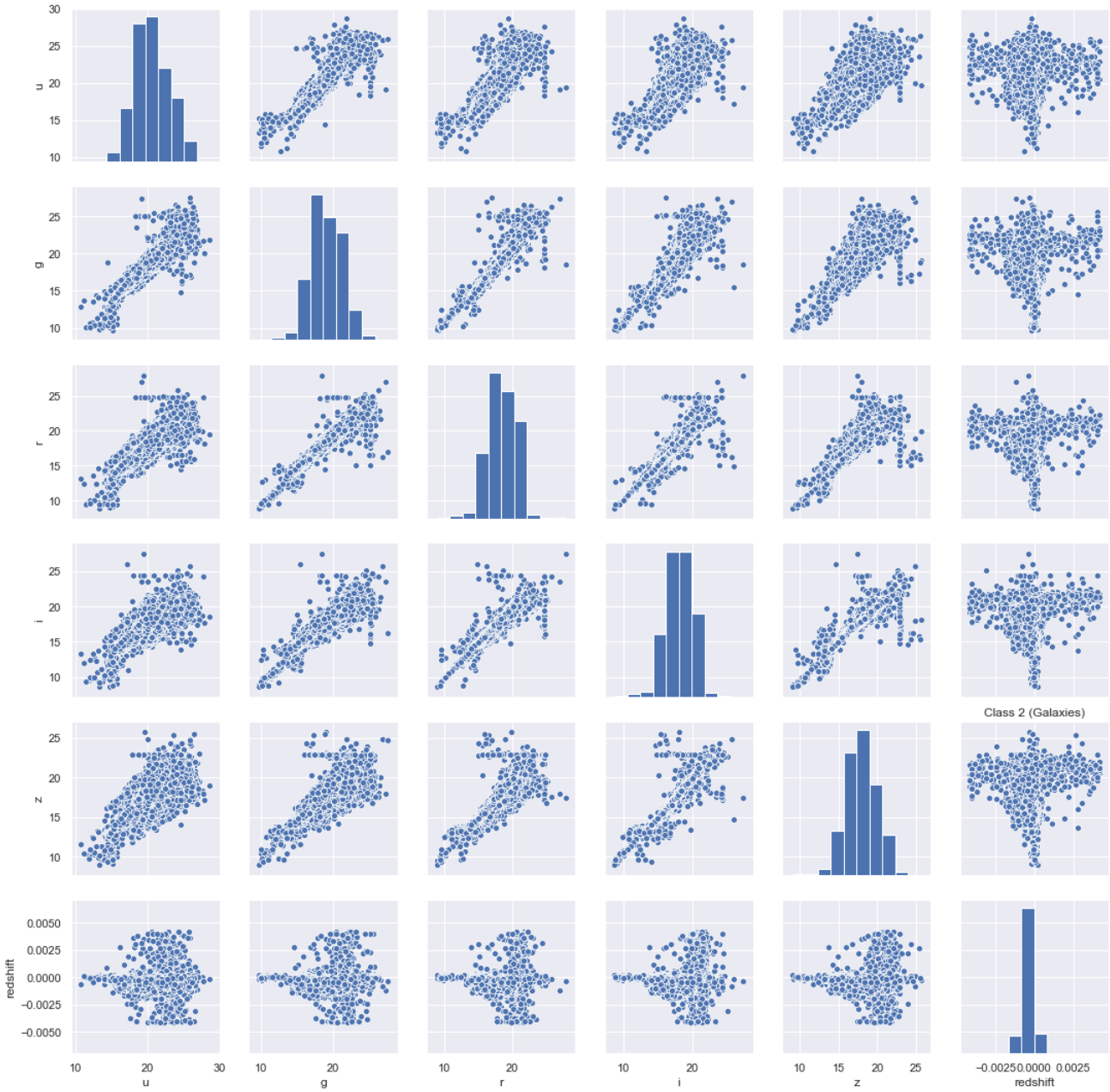
Σχήμα 1.7: Pairplots και των 3 κλάσεων



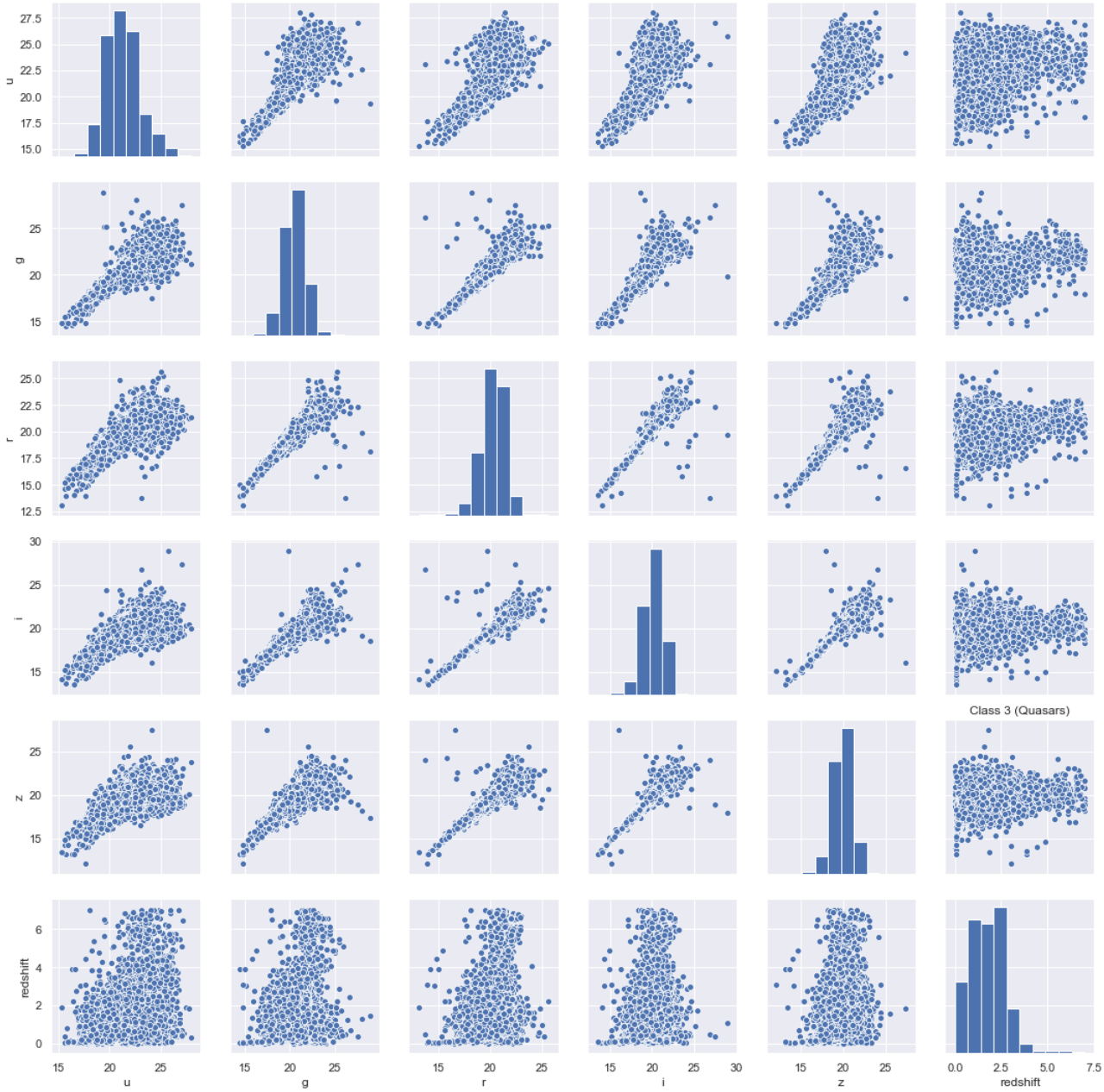
Σχήμα 1.8: Pairplots Αστέρων



Σχήμα 1.9: Pairplots Γαλαξιών



Σχήμα 1.10: Pairplots Κβάζαρς



Παρατηρούμε ότι στην κλάση των γαλαξιών οι συσχετίσεις των u, g, r, i, z με το redshift δεν είναι γραμμικές, οπότε ο συντελεστής γραμμικής συσχέτισης δεν είναι αντιπροσωπευτικός της πραγματικότητας. Επομένως, συνδυάζοντας τους πίνακες συσχέτισης με τις γραφικές παραστάσεις καταλήγουμε στο συμπέρασμα ότι υπάρχει **έντονη γραμμική συσχέτιση μεταξύ των μεταβλητών u, g, r, i, z (ανά δύο)** ενώ η μεταβλητή redshift είναι ασυσχέτιστη με αυτές.

1.8 Τυποποίηση δεδομένων

Για ορισμένα από τα μοντέλα Μηχανικής Μάθησης, είναι απαραίτητο να τυποποιούμε τα δεδομένα πριν εκπαιδεύσουμε τους ταξινομητές. Συγεκριμένα, τα δεδομένα πρέπει να τυποποιούνται πριν την εφαρμογή των μεθόδων Linear Discriminant Analysis, καθώς και πριν την εκπαίδευση των Νευρωνικών Δικτύων. Η τυποποίηση των δεδομένων είναι απαραίτητη όταν οι διακριτές μεταβλητές έχουν μεγάλες διαφορές στα εύρη τιμών τους, ή είναι ποσότητες μετρημένες σε διαφορετικές μονάδες μέτρησης. Αυτές οι διαφορές κλίμακας τιμών προκαλούν προβλήματα σε αρκετά μοντέλα μηχανικής μάθησης, όπως σε αυτά που βασίζονται σε υπολογισμούς αποστάσεων. Σκοπός είναι τα μετασχηματισμένα δεδομένα να έχουν την ίδια κλίμακα. Τα δεδομένα τυποποιήθηκαν αφαιρώντας από τις αρχικές τιμές την μέση τιμή της μεταβλητής και διαιρώντας με την τυπική απόκλιση. Έτσι, όλες οι μεταβλητές έχουν την ίδια μέση τιμή $\mu = 0$ και ίδια τυπική απόκλιση $\sigma = 1$.

Για να εφαρμόσουμε τη μέθοδο της τυποποίησης υποθέτουμε ότι τα δεδομένα ακολουθούν Κανονική κατανομή. Ωστόσο, αυτή δεν είναι απαραίτητη προϋπόθεση για την εφαρμογή της τεχνικής, αλλά εάν η υπόθεση της κανονικότητας ισχύει για κάθε μεταβλητή, τότε η μέθοδος της τυποποίησης είναι πιο αποτελεσματική. Στην περίπτωση του προβλήματος που έχουμε στη διάθεσή μας, καμία από τις 6 μεταβλητές δεν ακολουθεί Κανονική κατανομή, όπως είδαμε στο Υποκεφάλαιο 1.6.

Κεφάλαιο 2

Εισαγωγή στη Μηχανική Μάθηση

2.1 Ορισμός

Η **Μηχανική Μάθηση (Machine Learning)** αποτελεί κλάδο της Επιστήμης των Υπολογιστών και συγκεκριμένα ανήκει στο πεδίο της Τεχνητής Νοημοσύνσης (Artificial Intelligence). Σύμφωνα με τον πρωτοπόρο της Τεχνητής Νοημοσύνης , και εμπνευστή της έννοιας της Μηχανικής Μάθησης, Arthur Samuel, *"η Μηχανική Μάθηση είναι ένα πεδίο μελέτης που δίνει στους υπολογιστές την ικανότητα να μαθαίνουν, χωρίς να έχουν ρητά προγραμματιστεί"*.

Με τη βοήθεια των αλγορίθμων Μηχανικής Μάθησης οι επιστήμονες επιχειρούν να δημιουργήσουν υπολογιστικά συστήματα που είναι ικανά να μαθαίνουν. Δηλαδή μπορούν να βελτιώνουν την απόδοσή τους κατά την εκτέλεση μιας συγκεκριμένης εργασίας, αξιοποιώντας τη γνώση που αποκόμισαν κατά την εκπαίδευσή τους με δεδομένα. Η λειτουργία τέτοιων αλγορίθμων βασίζεται στην κατασκευή μοντέλων, με στόχο την εξαγωγή αποφάσεων.

2.2 Είδη Μηχανικής Μάθησης

Ο τομέας της Μηχανικής Μάθησης αποτελείται από αλγόριθμους που ακολουθούν τρεις τρόπους μάθησης:

1. **Επιβλεπόμενη Μάθηση (Supervised Learning)**

   Αποτελεί τη διαδικασία κατά την οποία ο αλγόριθμος απεικονίζει δεδομένες εισόδους σε γνωστές επιθυμητές εξόδους, με απώτερο στόχο τη γενίκευση της διαδικασίας για εισόδους με άγνωστη έξοδο. Χρησιμοποιείται σε προβλήματα:

   - *Ταξινόμησης (Classification):*

     Κατά την εκπαίδευση κάθε διάνυσμα του δείγματος αντιπροσωπεύεται από μια τιμή η οποία είναι μοναδική για κάθε κλάση (class label). Ο αλγόριθμος καλείται να ταξινομήσει τις παρατηρήσεις στις κλάσεις.

- *Παλινδρόμησης (Regression):*

  Τα μοντέλα παλινδρόμησης χρησιμοποιούνται για να προβλέψουν συνεχείς τιμές. Κατά την εκπαίδευση κάθε διάνυσμα του δείγματος διαθέτει μια επιπλέον μεταβλητή, τη μεταβλητή απόκρισης η οποία αποτελεί την επιθυμητή τιμή εξόδου.

2. **Μη Επιβλεπόμενη Μάθηση (Unsupervised Learning)**

   Στο μοντέλο εισάγονται παρατηρήσεις χωρίς τις αντίστοιχες επιθυμητές εξόδους. Η Μη Επιβλεπόμενη Μάθηση χρησιμοποιείται σε προβλήματα:

   - *Ανάλυσης Συσχετισμών (Association Analysis):*

     Χρησιμοποιείται για ανίχνευση ενδιαφερουσών συσχετίσεων που υπάρχουν σε μεγάλα σύνολα δεδομένων.

   - *Ομαδοποίησης (Clustering):*

     Το μοντέλο εκπαιδεύεται με δεδομένα έτσι ώστε να καταλάβει τις ιδιότητές τους. Έτσι, αποκτά τη δυνατότητα να χωρίζει τα δεδομένα σε ομάδες, κατατάσσοντας στην ίδια ομάδα αυτά που παρουσιάζουν ομοιότητες.

3. **Ενισχυτική Μάθηση (Reinforcement Learning):**

   Ο αλγόριθμος μαθαίνει μια στρατηγική ενεργειών μέσα από άμεση αλληλεπίδραση με το περιβάλλον. Χρησιμοποιείται κυρίως σε προβλήματα *Σχεδιασμού (Planning)*, όπως για παράδειγμα ο έλεγχος κίνησης ενός ρομπότ και η βελτιστοποίηση εργασιών σε εργοστασιακούς χώρους.

2.3 Εφαρμογές Μηχανικής Μάθησης

Η Μηχανική Μάθηση κρύβεται πίσω από πολλές σύγχρονες τεχνολογίες που χρησιμοποιούμε στην καθημερινότητά μας, ή άλλες πιο εξελιγμένες εφαρμογές που βρίσκονται ακόμα υπό μελέτη. Μερικά παραδείγματα είναι:

- Παράδειγμα Επιτηρούμενης Μάθησης:

  Οι περισσότερες πλατφόρμες ηλεκτρονικού ταχυδρομείου ταξινομούν την ηλεκτρονική μας αλληλογραφία σε μηνύματα Spam (κακόβουλη αλληλογραφία) και not Spam, χρησιμοποιώντας τον κατάλληλο αλγόριθμο ταξινόμησης.

- Παράδειγμα μη Επιτηρούμενης Μάθησης:

  Μέσω των αναζητήσεων μας σε ιστοσελίδες με καταναλωτικά αγαθά, αλγόριθμοι μηχανικής μάθησης καταγράφουν τις καταναλωτικές μας συνήθειες. Με αυτό τον τρόπο μπορούν να μας κατατάξουν σε ομάδες καταναλωτών βάσει των ενδιαφερόντων μας. Έτσι, είναι

εφικτό κάθε φορά που μπαίνουμε στην ιστοσελίδα να προωθούνται διάφορα προϊόντα που πιθανό να μας ενδιαφέρουν.

- Παράδειγμα Ενισχυτικής Μάθησης:

  Προκειμένου να μάθει ένα ρομπότ να περπατά, ένας τρόπος είναι να το ανταμείβουμε αναλόγως της προόδου του. Δηλαδή κάθε φορά που περπατά και πέφτει, να παίρνει χαμηλό σκορ, ενώ κάθε φορά που καταφέρνει να περπατήσει, να παίρνει υψηλό σκορ. Έτσι, μετά από επανειλημμένες δοκιμές, το ρομπότ θα σημειώνει βελτίωση στην κίνησή του.

2.4 Μαζικά Δεδομένα (Big Data)

Ως Μαζικά Δεδομένα χαρακτηρίζονται οι πολύ μεγάλες ποσότητες δεδομένων. Αποτελούν πληροφορίες που συλλέγονται σε πραγματικό χρόνο από πηγές όπως το διαδίκτυο, τα μέσα κοινωνικής δικτύωσης, αισθητήρες, τηλεσκόπια ή δίκτυα.

Σήμερα, χάρη στους ισχυρούς επεξεργαστές και τους αλγόριθμους, τα δεδομένα αυτά μπορούν να καταγραφούν και να αναλυθούν με τον τρόπο που επιθυμούμε. Για την επεξεργασία μεγάλων όγκων δεδομένων απαιτούνται εξελιγμένες τεχνικές ανάλυσης δεδομένων που χρησιμοποιούν μεθόδους Μηχανικής Μάθησης, Ανάλυσης Φυσικής Γλώσσας (Natural Language Processing), Εξόρυξης Δεδομένων (Data Mining) κτλ.

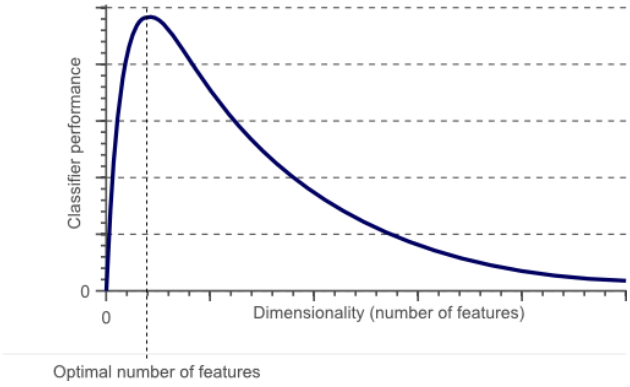
Με τις τεχνικές αυτές, μπορούμε να ανακτήσουμε πληροφορίες από τα δεδομένα οι οποίες βοηθούν τον επιχειρηματικό ή τον επιστημονικό τομέα στη λήψη αποφάσεων και στην εξαγωγή συμπερασμάτων. Κάποιοι από τους τομείς που επωφελούνται από τα Μαζικά Δεδομένα είναι το μάρκετινγκ και η διαφήμιση, ο αστικός σχεδιασμός, οι μεταφορές, το περιβάλλον και η αστρονομία. Ένα παράδειγμα συλλογής Μαζικών Δεδομένων από τηλεσκόπια είναι η συλλογή πληροφοριών που διεξάγεται κατά την Sloan Digital Sky Survey.

Κεφάλαιο 3

Linear Discriminant Analysis

3.1 Curse of dimensionality

Τα μοντέλα Μηχανικής Μάθησης προέχουν στο να αναλύουν δεδομένα πολλών διαστάσεων. Ωστόσο, όσο αυξάνεται ο αριθμός των μεταβλητών του προβλήματος, απαιτείται μεγαλύτερη υπολογιστική ισχύ (CPU power) και μνήμη RAM για την επεξεργασία των δεδομένων. Επιπλέον, με την αύξηση των μεταβλητών, ο αριθμός των δεδομένων που απαιτούνται για εκπαίδευση του μοντέλου, αυξάνεται εκθετικά. Σύμφωνα με το πιο κάτω διάγραμμα, για κάθε πρόβλημα υπάρχει κάποιος βέλτιστος αριθμός χαρακτηριστικών μεταβλητών, που το μοντέλο παρουσιάζει την καλύτερη απόδοση.



Σχήμα 3.1: Πηγή εικόνας:
https://medium.com/diogo-menezes-borges/give-me-the-antidote-for-the-curse-of-dimensionality-b14bce4bf4d2

Ένας τρόπος να μειώσουμε τη διάσταση του προβλήματος ούτως ώστε να μελετήσουμε την απόδοση του μοντέλου, είναι η μέθοδος Linear Discriminant Analysis. Παρακάτω μελετάμε την απόδοση που έχουν οι γραμμικοί ταξινομητές, εάν μειώσουμε τη διάσταση του προβλήματος από 6 σε 2 διαστάσεις.

3.2 Εισαγωγή στην Linear Discriminant Analysis

Η Linear Discriminant Analysis (LDA) αποτελεί τεχνική μείωσης των διαστάσεων ενός προβλήματος. Στόχο έχει να προβάλει το σύνολο των δεδομένων σε έναν υπόχωρο μικρότερης διάστασης στον οποίο οι κλάσεις εμφανίζουν τη μέγιστη δυνατή διαχωριστική ικανότητα έτσι ώστε να αποφεύγονται οι αλληλοεπικαλύψεις των σημείων. Πιο συγκεκριμένα, η LDA ψάχνει την προβολή του συνόλου δεδομένων η οποία μεγιστοποιεί την διασπορά μεταξύ των κλάσεων και ελαχιστοποιεί τη διασπορά εντός των κλάσεων. Προκειμένου να εφαρμοστεί η μέθοδος LDA στα δεδομένα, θεωρητικά πρέπει να τηρούνται οι παρακάτω προϋποθέσεις:

1. Τα δεδομένα ακολουθούν Κανονική Κατανομή.

   Στα διαθέσιμα δεδομένα του SDSS, από τα ιστογράμματα των μεταβλητών (βλ. υποκεφάλαιο 1.6), φαίνεται ότι καμία από τις μεταβλητές δεν ακολουθεί Κανονική Κατανομή.

2. Οι μεταβλητές είναι στατιστικά ανεξάρτητες.

   Από τους πίνακες συσχετίσεων και από τα *pairplots* φαίνεται ότι υπάρχουν γραμμικές συσχετίσεις μεταξύ των μεταβλητών u, g, r, i, z (ανά δύο μεταβλητές). Οι συσχετίσεις είναι έντονες αφού είναι της τάξης του 85%-98%. Ο συντελεστής συσχέτισης ανιχνεύει γραμμική εξάρτηση μεταξύ δύο μεταβλητών. Επομένως οι μεταβλητές εδώ είναι στατιστικά εξαρτημένες.

3. Οι πίνακες διασποράς είναι ίδιοι σε όλες τις κλάσεις.

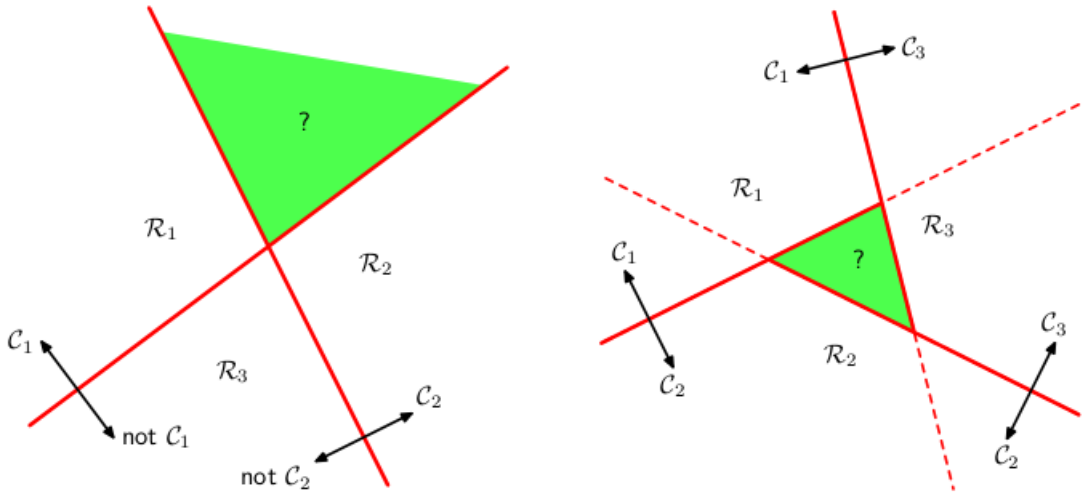
   Στο συγκεκριμένο πρόβλημα, οι πίνακες διασποράς διαφέρουν πολύ ανά κλάση.

Ωστόσο, βάσει της αναφοράς [2], η LDA, δεν απαιτεί να τηρούνται οι προϋποθέσεις της Κανονικής Κατανομής των δεδομένων, και του ίδιου πίνακα διασποράς των κλάσεων. Ακόμα και όταν παραβιάζεται η δεύτερη προϋπόθεση, η μέθοδος μπορεί να εφαρμοστεί.

Έστω πρόβλημα ταξινόμησης παρατηρήσεων D - διαστάσεων, σε $K > 2$ κλάσεις. Προκειμένου να ταξινομήσουμε τις παρατηρήσεις σε $K > 2$ κλάσεις θα μπορούσαμε να συνδυάσουμε ένα αριθμό ταξινομητών που διαχωρίζουν τις παρατηρήσεις σε 2 κλάσεις ο καθένας. Ωστόσο, αυτή η λογική μπορεί να δημιουργήσει προβλήματα.

Έστω ότι εφαρμόζεται η μέθοδος *one-versus-the-rest*, όπου κατασκευάζονται (K-1) ταξινομητές, που ο καθένας διαχωρίζει τις παρατηρήσεις της κλάσης C_k από τις παρατηρήσεις των υπόλοιπων κλάσεων, για $k = 1, 2, ...(K-1)$. Με αυτό τον τρόπο, δημιουργούνται περιοχές στον φασικό χώρο οι οποίες δεν ανήκουν σε κάποια συγκεκριμένη κλάση (βλ. σχήμα 3.2(α)). Μια άλλη μέθοδος, γνωστή ως *one-versus-one*, κατασκευάζει ένα ταξινομητή ανα δύο κλάσεις, οπότε $\frac{K(K-1)}{2}$ διαχωριστικές επιφάνειες. Όμως και πάλι δημιουργούνται περιοχές που δεν ανήκουν σε κάποια κλάση (βλ. σχήμα 3.2(β)).

Σχήμα 3.2: Πηγή: Βιβλίο *Pattern Recognition and Machine Learning*, C. M. Bishop, σελ. 183



Για να αποφευχθούν αυτά τα φαινόμενα, κατασκευάζεται ένας ταξινομητής που συνδυάζει K γραμμικές συναρτήσεις, της μορφής:

$$a_k(\mathbf{x}) = \mathbf{v}_k^T \mathbf{x} + v_{k_0}$$

για $k = 1, 2, ...K$. Η κάθε παρατήρηση $\mathbf{x}$ κατατάσσεται στην κλάση C_k αν $a_k(\mathbf{x}) > a_j(\mathbf{x})$ για κάθε j $\neq$ k. Οπότε το όριο απόφασης μεταξύ των κλάσεων C_k και C_j δίνεται από τις εξισώσεις $a_k(\mathbf{x}) = a_j(\mathbf{x})$, οι οποίες αντιστοιχούν σε υπερεπίπεδα διάστασης $(D-1)$.

3.3 Θεωρία LDA

Η μέθοδος Linear Discriminant Analysis (LDA) χωρίζεται σε δύο στάδια:

Αρχικά επιδιώκει να μειώσει τη διάσταση του προβλήματος, διατηρώντας όση δυνατόν περισσότερη πληροφορία από τα δεδομένα. Επιχειρήσαμε να μειώσουμε τις διαστάσεις του προβλήματός μας χρησιμοποιώντας την μέθοδο που πρότεινε ο Fisher, και στη συνέχεια τη μέθοδο LDA μέσω SVD. Και οι δύο διαδικασίες περιγράφονται αναλυτικά στα ακόλουθα κεφάλαια.

Μετά την μείωση των διαστάσεων, κατασκευάζονται γραμμικές συναρτήσεις των μεταβλητών, προκειμένου να λειτουργήσουν ως όρια απόφασης. Για την κατασκευή των όριων απόφασης χρησιμοποιείται ο **Κανόνας Bayes**, που εφαρμόζει τη συνθήκη ελαχιστοποίησης του αναμενόμενου σφάλματος. Υπολογίζονται αρχικά οι *posteriori* πιθανότητες της κάθε κλάσης και στη συνέχεια οι παρατηρήσεις κατατάσσονται στην κλάση με την μεγαλύτερη *posterior* πιθανότητα.

Ταξινόμηση βάσει του κανόνα Bayes:

Θεωρούμε ότι έχουμε στη διάθεσή μας ένα πρόβλημα με D χαρακτηριστικές μεταβλητές και οι παρατηρήσεις ανήκουν σε $K > 2$ κλάσεις. Σε κάθε κλάση προσαρμόζεται η Κανονική Κατανομή, δηλαδή κάθε κλάση k θεωρούμε ότι έχει συνάρτηση πιθανοφάνειας:

$$p(\mathbf{x}|C_k) = \frac{1}{(2\pi)^{D/2}|\Sigma|^{\frac{1}{2}}} \exp\left(-\frac{1}{2}(\mathbf{x}-\mu_k)^T\Sigma^{-1}(\mathbf{x}-\mu_k)\right)$$

Οι προβλέψεις του μοντέλου γίνονται βάσει του κανόνα Bayes:

$$p(C_k|\mathbf{x}) = \frac{p(\mathbf{x}|C_k)p(C_k)}{\sum_j p(\mathbf{x}|C_j)p(C_j)} = \frac{exp(\alpha_k)}{\sum_j exp(\alpha_j)}$$

με $\alpha_k = ln(p(\mathbf{x}|C_k)p(C_k))$

Για να χρησιμοποιήσουμε το παραπάνω μοντέλο ως ταξινομητή χρειάζεται να υπολογίσουμε από τα δεδομένα εκπαίδευσης τις priori πιθανότητες $P(C_k)$ (από το πηλίκο του αριθμού των δεδομένων εκπαίδευσης της κλάσης k προς το συνολικό αριθμό δεδομένων εκπαίδευσης), τους μέσους των κλάσεων μ_k, καθώς και τους πίνακες συνδιασποράς Σ_k. Εξισώνοντας τις συναρτήσεις των *posteriori* πιθανοτήτων, και θεωρώντας ότι οι πίνακες διασποράς είναι οι ίδιοι για κάθε κλάση, μετά από πράξεις προκύπτουν τα εξής γραμμικά όρια απόφασης:

$$\alpha_k(\mathbf{x}) = \mathbf{v}_k^T\mathbf{x} + v_{k_0}$$

όπου

$$\mathbf{v}_k = \Sigma^{-1}\mu_k$$

και

$$v_{k_0} = -\frac{1}{2}\mu_k^T\Sigma^{-1}\mu_k + lnP(C_k)$$

Το διάνυσμα $\mathbf{v}$ καθορίζει τη διεύθυνση της επιφάνειας διαχωρισμού αφού είναι κάθετο σε αυτήν.

3.4 Μέτρα αξιολόγησης της LDA

Αφού ολοκληρωθεί η υλοποίηση του αλγορίθμου, αξιολογούμε την απόδοσή του για να συγκριθεί στη συνέχεια με την απόδοση των υπόλοιπων αλγορίθμων μηχανικής μάθησης. Για την αξιολόγηση προβλημάτων ταξινόμησης σε περισσότερες από δύο κλάσεις, χρησιμοποιούνται συνήθως το F1-score, η ποσότητα log-loss και τα confusion matrices.

- **F1-score:**

  Το F1-score μετράει το πόσο ακριβής είναι ο ταξινομητής καθώς επίσης και την *ανθεκτικότητά* του (robustness). Ισούται με τον βεβαρυμένο μέσο δύο άλλων ποσοτήτων, των precision και recall. Παίρνει τιμές στο διάστημα $[0, 1]$. Όσο υψηλότερη είναι η τιμή του, τόσο καλύτερος είναι ο ταξινομητής.

$$F1 - score = \frac{2 * Precision * Recall}{Precision + Recall}$$

- **Precision:**

  Η πιθανότητα να ταξινομηθεί σωστά κάποιο στοιχείο της κλάσης i. Δηλαδή είναι το πηλίκο των σημείων που ταξινομήθηκαν ορθά στην κλάση i προς το σύνολο των σημείων που ταξινομήθηκαν στην κλάση i.

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

- **Recall:**

  Το ποσοστό των σωστών ταξινομήσεων, δηλαδή το πηλίκο των σημείων που ταξινομήθηκαν ορθά στην κλάση i προς το σύνολο των σημείων της κλάσης i.

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

- **Log-Loss:**

  Το Log-Loss (logarithmic loss) ποινικοποιεί τα διανύσματα που ταξινομήθηκαν σε λανθασμένη κλάση. Για κάθε παρατήρηση, ο ταξινομητής επιστρέφει την πιθανότητα να ανήκει σε μία από τις 3 κλάσεις. Αν υποθέσουμε ότι υπάρχουν N δείγματα και K κλάσεις, τότε το Log-Loss υπολογίζεται ως εξής:

$$Log - loss = \frac{-1}{N} \sum_{i=1}^{N} \sum_{j=1}^{K} y_{ij} * log(p_{ij})$$

όπου y_{ij} είναι 1 ή 0 αναλόγως αν η παρατήρηση $\mathbf{x_i}$ ανήκει στην κλαση C_j ή όχι, ενώ p_{ij} είναι η πιθανότητα το $\mathbf{x_i}$ να ανήκει στην κλάση C_j. Το Log-Loss παίρνει τιμές στο διάστημα $[0, +\infty)$. Όσο πιο κοντά στο 0 είναι, τόσο καλύτερη η απόδοση του ταξινομητή, ενώ όσο απομακρύνεται από το 0, τόσο χειρότερη η απόδοση.

- **Confusion Matrices:**

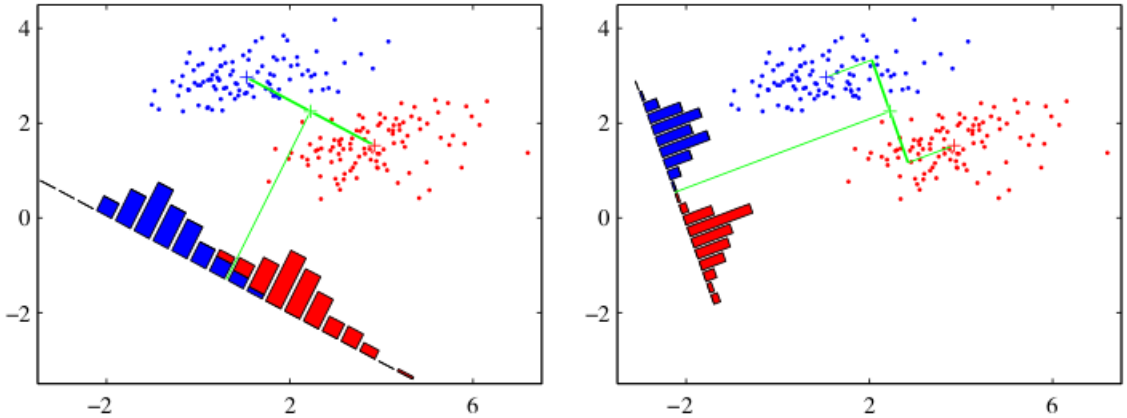
  Κάθε Confusion Matrix έχει στο κελί (i, j) τον αριθμό των παρατηρήσεων που ανήκουν στην κλάση i και ταξινομήθηκαν στην κλάση j. Για $i = j$ (διαγώνια στοιχεία) έχουμε τον αριθμό των στοιχείων που ταξινομήθηκαν ορθά, ενώ για $i \neq j$ έχουμε τον αριθμό των στοιχείων που ταξινομήθηκαν λανθασμένα.

3.5 Μέθοδος 1: Μέθοδος Fisher

Μείωση των διαστάσεων μέσω της μεθόδου Fisher:

Μέσω της μεθόδου Fisher υπολογίζεται η κατεύθυνση στην οποία αν προβάλουμε τα δεδομένα επιτυγχάνουμε τον μέγιστο διαχωρισμό μεταξύ των κλάσεων. Ουσιαστικά, η Fisher μειώνει τη διάσταση του προβλήματος μέσω γραμμικού μετασχηματισμού των διανυσμάτων. Όμως, μειώνοντας τη διάσταση του προβλήματος, χάνεται μέρος της πληροφορίας που περιέχεται στα δεδομένα και αυτό οδηγεί σε χειρότερο διαχωρισμό μεταξύ των παρατηρήσεων που ανήκουν σε διαφορετικές κλάσεις (βλ.σχήμα 3.3[α]). Ωστόσο, με κατάλληλη επιλογή της προβολής, μπορεί να επιτευχθεί μεγιστοποίηση του διαχωρισμού, δηλαδή μικρότερη αλληλεπικάλυψη μεταξύ των σημείων που ανήκουν σε διαφορετικές κλάσεις (βλ. σχήμα 3.3[β]).

Σχήμα 3.3: Πηγή: Βιβλίο *Pattern Recognition and Machine Learning*, C. M. Bishop, σελ. 188



Αρχικά θα περιγράψουμε την μέθοδο Fisher για **K = 2 κλάσεις και D-διαστάσεις** και στη συνέχεια θα επεκταθούμε στις $K > 2$ κλάσεις.
Θεωρούμε ότι η κλάση C_1 έχει N_1 παρατηρήσεις ενώ η C_2 έχει N_2 παρατηρήσεις. Το κριτήριο Fisher ορίζεται ως το πηλίκο της απόστασης μεταξύ των μέσων τιμών (μετά την προβολή) προς το άθροισμα των διασπορών των κλάσεων (μετά την προβολή):

$$J(\mathbf{w}) = \frac{(\widetilde{m_2} - \widetilde{m_1})^2}{\widetilde{s_1^2} + \widetilde{s_2^2}}$$

όπου

$$\begin{aligned}
\mathbf{m}_1 &= \frac{1}{N_1}\sum_{n\in C_1}\mathbf{x}_n \\
\mathbf{m}_2 &= \frac{1}{N_2}\sum_{n\in C_2}\mathbf{x}_n \\
\widetilde{m_2}-\widetilde{m_1} &= \mathbf{w}^T(\mathbf{m}_2-\mathbf{m}_1) \\
\widetilde{s_k^2} &= \sum_{n\in C_k}(y_n-\widetilde{m_k})^2
\end{aligned}$$

όπου

$$y_n = \mathbf{w}^T\mathbf{x}_n$$

η προβολή της παρατήρησης x_n σε μία διάσταση.

Ο αριθμητής της $J(\mathbf{w})$ μεγιστοποιείται ενω ο παρονομαστής ελαχιστοποιείται, οπότε ουσιαστικά η μέθοδος Fisher επιδιώκει να βρει το διάνυσμα $\mathbf{w}$ το οποίο μεγιστοποιεί τη συνάρτηση $J(\mathbf{w})$.

Για να εκφραστεί το $J(\mathbf{w})$ συναρτήσει του $\mathbf{w}$ ορίζεται ο πίνακας συνδιασποράς εντός των κλάσεων (Within-class scatter matrix):

$$\begin{aligned}
S_W &= S_1+S_2 \\
&= \sum_{n\in C_1}(\mathbf{x}_n-\mathbf{m}_1)(\mathbf{x}_n-\mathbf{m}_1)^T+\sum_{n\in C_2}(\mathbf{x}_n-\mathbf{m}_2)(\mathbf{x_n}-\mathbf{m}_2)^T
\end{aligned}$$

Μετά από πράξεις έχουμε:

$$\widetilde{s_1}^2+\widetilde{s_2}^2 = ... = \mathbf{w}^TS_1\mathbf{w}+\mathbf{w}^TS_2\mathbf{w} = \mathbf{w}^TS_W\mathbf{w}$$

και ο πίνακας συνδιασποράς μεταξύ των κλάσεων (Between-class scatter matrix):

$$S_B = (\mathbf{m}_2-\mathbf{m}_1)(\mathbf{m}_2-\mathbf{m}_1)^T$$

$$(\widetilde{m_2}-\widetilde{m_1})^2 = ... = \mathbf{w}^TS_B\mathbf{w}$$

$$\Rightarrow J(\mathbf{w}) = \frac{(\widetilde{m_2}-\widetilde{m_1})^2}{\widetilde{s_1}^2+\widetilde{s_2}^2} = \frac{\mathbf{w}^TS_B\mathbf{w}}{\mathbf{w}^TS_W\mathbf{w}}$$

Για την μεγιστοποίηση της αντικειμενικής συνάρτησης $J(\mathbf{w})$, παραγωγίζουμε ως προς $\mathbf{w}$:

$$\frac{dJ(\mathbf{w})}{d\mathbf{w}} = 0$$

$$\Leftrightarrow (\mathbf{w}^TS_B\mathbf{w})S_W\mathbf{w} = (\mathbf{w}^TS_W\mathbf{w})S_B\mathbf{w}$$

$$\Leftrightarrow \frac{(\mathbf{w}^T S_B \mathbf{w})}{(\mathbf{w}^T S_W \mathbf{w})} S_W \mathbf{w} = S_B \mathbf{w}$$

Θέτοντας:

$$\lambda = \frac{(\mathbf{w}^T S_B \mathbf{w})}{(\mathbf{w}^T S_W \mathbf{w})}$$

έχουμε να επιλύσουμε ένα *πρόβλημα εύρεσης ιδιοτιμών - ιδιοδιανυσμάτων*:

$$\lambda S_W \mathbf{w} = S_B \mathbf{w}$$

Αν ο S_W είναι αντιστρέψιμος τότε έχουμε:

$$\lambda \mathbf{w} = S_W^{-1} S_B \mathbf{w}$$

Αφού όμως το $S_B \mathbf{w}$ έχει την ίδια κατεύθυνση με το $(\mathbf{m}_2 - \mathbf{m}_1)$ και αφού μας ενδιαφέρει μόνο η διεύθυνση του $\mathbf{w}$ μπορούμε να λύσουμε το πρόβλημα ιδιοτιμών - ιδιοδιανυσμάτων απευθείας:

$$\mathbf{w} = S_W^{-1}(\mathbf{m}_2 - \mathbf{m}_1)$$

Θα συνεχίσουμε γενικεύοντας την παραπάνω θεωρία για $\mathbf{K > 2}$ **κλάσεις**. Με τη μέθοδο Fisher μπορούμε να μειώσουμε τις διαστάσεις του προβλήματος σε $(K-1)$ το πολύ διαστάσεις. Υποθέτουμε ότι έχουμε στη διάθεσή μας D χαρακτηριστικές μεταβλητές, με $D > K$. Έχουμε $D' > 1$ γραμμικούς συνδυασμούς των μεταβλητών:

$$y_k = \mathbf{w}_k^T \mathbf{x}$$

για $k = 1, 2, ..., D'$, που ομαδοποιούνται στο διάνυσμα $\mathbf{y}$. Τα διανύσματα βαρών $\mathbf{w}_k$ αποτελούν τις στήλες του πίνακα $\mathbf{W}$ οπότε έχουμε:

$$\mathbf{y} = \mathbf{W}^T \mathbf{x}$$

Παρόμοια με προηγουμένως θέτουμε τον πίνακα συνδιασποράς εντός των κλάσεων:

$$S_W = \sum_{k=1}^{K} S_k$$

όπου

$$S_k = \sum_{n \in C_k} (\mathbf{x}_n - \mathbf{m}_k)(\mathbf{x}_n - \mathbf{m}_k)^T$$

και

$$\mathbf{m}_k = \frac{1}{N_k} \sum_{n \in C_k} \mathbf{x}_n$$

Επίσης ορίζουμε τον πίνακα συνολικής συνδιασποράς (Total covariance matrix):

$$S_T = \sum_{n=1}^{N} (\mathbf{x}_n - \mathbf{m})(\mathbf{x}_n - \mathbf{m})^T$$

όπου $\mathbf{m}$ ο μέσος όρος όλων των παρατηρήσεων του συνόλου δεδομένων:

$$\mathbf{m} = \frac{1}{N} \sum_{n=1}^{N} \mathbf{x}_n = \frac{1}{N} \sum_{k=1}^{K} N_k \mathbf{m}_k \ \ (*)$$

με $N = \sum_k N_k$

Έχουμε, λοιπόν:

$$S_T = S_W + S_B$$

όπου

$$S_B = \sum_{k=1}^{K} N_k (\mathbf{m}_k - \mathbf{m})(\mathbf{m}_k - \mathbf{m})^T$$

Οι παραπάνω πίνακες ορίζονται στον αρχικό φασικό χώρο. Ορίζουμε τους αντίστοιχους στον μετασχηματισμένο χώρο D'-διαστάσεων:

$$\begin{aligned} \widetilde{s_W} &= \sum_{k=1}^{K} \sum_{n \in C_k} (\mathbf{y}_n - \widetilde{\mathbf{m}}_k)(\mathbf{y}_n - \widetilde{\mathbf{m}}_k)^T \\ \widetilde{s_B} &= \sum_{k=1}^{K} N_k (\widetilde{\mathbf{m}}_k - \widetilde{\mathbf{m}})(\widetilde{\mathbf{m}}_k - \widetilde{\mathbf{m}})^T \end{aligned}$$

όπου

$$\begin{aligned} \widetilde{\mathbf{m}_k} &= \frac{1}{N_k} \sum_{n \in C_k} \mathbf{y}_n \\ \widetilde{\mathbf{m}} &= \frac{1}{N} \sum_{k=1}^{K} N_k \widetilde{\mathbf{m}}_k \end{aligned}$$

Το κριτήριο Fisher για $K > 2$ κλάσεις ορίζεται με διάφορες παρόμοιες συναρτήσεις, οι οποίες αποσκοπούν στο να βρεθεί ο πίνακας $\mathbf{W}$ για τον οποίο μεγιστοποιείται η διασπορά μεταξύ των κλάσεων και ελαχιστοποιείται η διασπορά εντός των κλάσεων. Για παράδειγμα μπορεί να χρησιμοποιηθεί ως αντικειμενική συνάρτηση η:

$$J(\mathbf{W}) = \frac{det(\mathbf{W}^T S_B \mathbf{W})}{det(\mathbf{W}^T S_W \mathbf{W})}$$

Οπότε λύνουμε το γενικευμένο πρόβλημα ιδιοτιμών-ιδιοδιανυσμάτων:

$$S_B \mathbf{w} = \lambda S_W \mathbf{w}$$

Μπορούμε να βρούμε το πολύ $(K-1)$ διαφορετικές ιδιοτιμές. Έστω $\mathbf{w}_1, \mathbf{w}_2, ...\mathbf{w}_{K-1}$ τα αντίστοιχα ιδιοδιανύσματα. Ο βέλτιστος πίνακας $\mathbf{W}$ που προβάλλει τα αρχικά διανύσματα σε υπόχωρο μικρότερης διάστασης δίνεται από τα ιδιοδιανύσματα που αντιστοιχούν στις D' μεγαλύτερες ιδιοτιμές.

3.5.1 Αλγόριθμος Fisher

Παρακάτω περιγράφεται βήμα-βήμα ο αλγόριθμος Fisher:

Βηκα 1:
Υπολογισμός του διανύσματος μέσης τιμής που αντιστοιχεί σε κάθε κλάση. Στην περίπτωση του προβλήματός μας, κάθε διάνυσμα έχει διάσταση 6:

$$m_k = [m_u, m_g, m_r, m_i, m_z, m_{redshift}]$$

για $k = 1, 2, 3$.
Στη συνέχεια υπολογίζονται οι πίνακες συνδιασποράς S_W, S_B, S_T.

Βηκα 2:
Επιλύεται το πρόβλημα ιδιοτιμών-ιδιοδιανυσμάτων:

$$\lambda \mathbf{w} = S_W^{-1} S_B \mathbf{w}$$

Βηκα 3:
Αφού καταταγούν οι ιδιοτιμές σε φθίνουσα σειρά, εντοπίζονται σε ποια ιδιοδιανύσματα αντιστοιχούν οι δύο μεγαλύτερες ιδιοτιμές, προκειμένου από αυτά να κατασκευαστεί ο νέος 2-διάστατος χώρος. Θα προκύψουν $K-1 = 3-1 = 2$ μη-μηδενικές ιδιοτιμές. Δηλαδή η διάσταση του προβλήματος μειώνεται από 6 σε 2 διαστάσεις. Αυτό συμβαίνει διότι ο πίνακας S_B είναι το άθροισμα $K = 3$ πινάκων όπου ο καθένας έχει βαθμό 1. Όμως, μόνο $K-1 = 2$ από αυτούς τους πίνακες είναι ανεξάρτητοι ως αποτέλεσμα του περιορισμού (*). Οπότε ο S_B έχει βαθμό το πολύ $(K-1) = 2$.
Ακολούθως επιλέγουμε τα 2 ιδιοδιανύσματα με τις μεγαλύτερες ιδιοτιμές και κατασκευάζουμε τον πίνακα $\mathbf{W}_{(6x2)}$ ο οποίος έχει σαν στήλες τα 2 ιδιοδιανύσματα.

Βημα 4:
Τα δεδομένα μετασχηματίζονται, προβάλλοντάς τα στο νέο 2-διάστατο χώρο που παράγεται από τα 2 ιδιοδιανύσματα με τις μεγαλύτερες ιδιοτιμές. Μετασχηματίζουμε δηλαδή τον πίνακα $\mathbf{X}$ πολλαπλασιάζοντάς τον με τον $\mathbf{W}$ για να πάρουμε τα μετασχηματισμένα δεδομένα:

$$\mathbf{Y} = \mathbf{XW}$$

Βημα 5:
Τα δεδομένα ταξινομούνται βάσει της θεωρίας Bayes που περιγράφεται πιο πάνω.

3.5.2 Υλοποίηση

Αρχικά τρέξαμε την μέθοδο Fisher βήμα-βήμα και στη συνέχεια την τρέξαμε με τη βοήθεια της βιβλιοθήκης της Python:

$$sklearn.discriminant_analysis.LinearDiscriminantAnalysis$$

με ορίσματα:

$$(solver = eigen, n_components = 2, tol = 0.0001)$$

Παρατηρήσαμε ότι τα αποτελέσματα που προέκυπταν ήταν τα ίδια, οπότε συνεχίσαμε την μελέτη με την βιβλιοθήκη.

Επιλογές παραμέτρων που διαθέτει η βιβλιοθήκη *sklearn*:

- **solver**:

  Θέσαμε solver = eigen ο οποίος εκτελεί την μέθοδο Fisher.

- **n_components**:

  Η διάσταση του προβλήματος μειώνεται στον αριθμό των συνιστωσών που θα θέσουμε. Θέτουμε να είναι ίσο με 2 ούτως ώστε να μειώσουμε την διάσταση του προβλήματος από 6 σε $K - 1 = 2$.

- **priors** :

  Ορίζουμε ένα διάνυσμα με τις priori πιθανότητες, δηλαδή με το ποσοστό των διανυσμάτων που ανήκουν σε κάθε κλάση, π.χ. priors $= [0.2265, 0.6175, 0.1560]$

3.5.3 Εκτέλεση και Αποτελέσματα

Με την εκτέλεση της μεθόδου, επιστρέφονται τα μετασχηματισμένα 2-διάστατα διανύσματα του συνόλου δεδομένων.

Με την εντολή $lda.scalings_$ δίνεται ο πίνακας $\mathbf{W}$, του οποίου οι στήλες αποτελούν τα ιδιοδιανύσματα του προβλήματος ιδιοτιμών-ιδιοδιανυσμάτων. Τα ιδιοδιανύσματα που αντιστοιχούν στις 2 μεγαλύτερες ιδιοτιμές θα χρησιμοποιηθούν ως βάση του νέου 2-διάστατου χώρου στον οποίο θα προβληθούν τα αρχικά διανύσματα.

Πίνακας $\mathbf{W}$:

$$[[-2.0467 \cdot 10^{-1}, -2.6798 \cdot 10^{-2}, 1.2851 \cdot 10^{-1}, 2.1286 \cdot 10^{-1}, 1.5065 \cdot 10^{-2}, 2.4387 \cdot 10^{-1}],$$

$$[-4.3784 \cdot 10^{-1}, 8.3729 \cdot 10^{-1}, -3.1239 \cdot 10^{-1}, -5.1604 \cdot 10^{-1}, 1.7706 \cdot 10^{-1}, 8.3984 \cdot 10^{-2}],$$

$$[1.5539 \cdot 10^{-1}, -3.9544 \cdot 10^{-1}, 5.7591 \cdot 10^{-1}, 3.6428 \cdot 10^{-1}, 2.1848 \cdot 10^{-1}, -8.6460 \cdot 10^{-1}],$$

$$[3.4515 \cdot 10^{-1}, -1.2232 \cdot 10^{-1}, -6.8081 \cdot 10^{-1}, 5.0210 \cdot 10^{-1}, 4.5620 \cdot 10^{-1}, 3.1197 \cdot 10^{-1}],$$

$$[2.9423 \cdot 10^{-2}, -2.7994 \cdot 10^{-1}, 3.0051 \cdot 10^{-1}, -5.5099 \cdot 10^{-1}, 5.9074 \cdot 10^{-1}, 2.9754 \cdot 10^{-1}],$$

$$[7.8884 \cdot 10^{-1}, 2.2031 \cdot 10^{-1}, 2.0471 \cdot 10^{-2}, 3.8062 \cdot 10^{-4}, -6.0299 \cdot 10^{-1}, 9.1908 \cdot 10^{-3}]]$$

Για την ταξινόμηση των παρατηρήσεων στις 3 κλάσεις, η LDA υπολογίζει 3 επιφάνειες διαχωρισμού, δηλαδή 3 γραμμικές συναρτήσεις της μορφής:

$$a_k(\mathbf{x}) = \mathbf{v}_k^T \mathbf{x} + v_{k0}$$

για $k = 1, 2, 3$

Συγκεκριμένα για το πρόβλημά μας έχουμε

$$a_1(x) = 0.09 * x_u - 0.09 * x_g + 0.06 * x_r - 0.10 * x_i + 0.08 * x_z - 0.39 * x_{redshift} - 1.62$$

$$a_2(x) = 0.06 * x_u + 0.24 * x_g - 0.10 * x_r - 0.12 * x_i - 0.04 * x_z - 0.21 * x_{redshift} - 0.53$$

$$a_3(x) = -0.36 * x_u - 0.80 * x_g + 0.29 * x_r + 0.62 * x_i + 0.06 * x_z + 1.39 * x_{redshift} - 3.43$$

από τα οποία προκύπτουν οι τρεις επιφάνειες διαχωρισμού:

$$a_1(x) = a_2(x)$$

$$\leftrightarrow 0.03 * x_u - 0.33 * x_g + 0.16 * x_r + 0.02 * x_i + 0.12 * x_z - 0.18 * x_{redshift} - 1.09 = 0$$

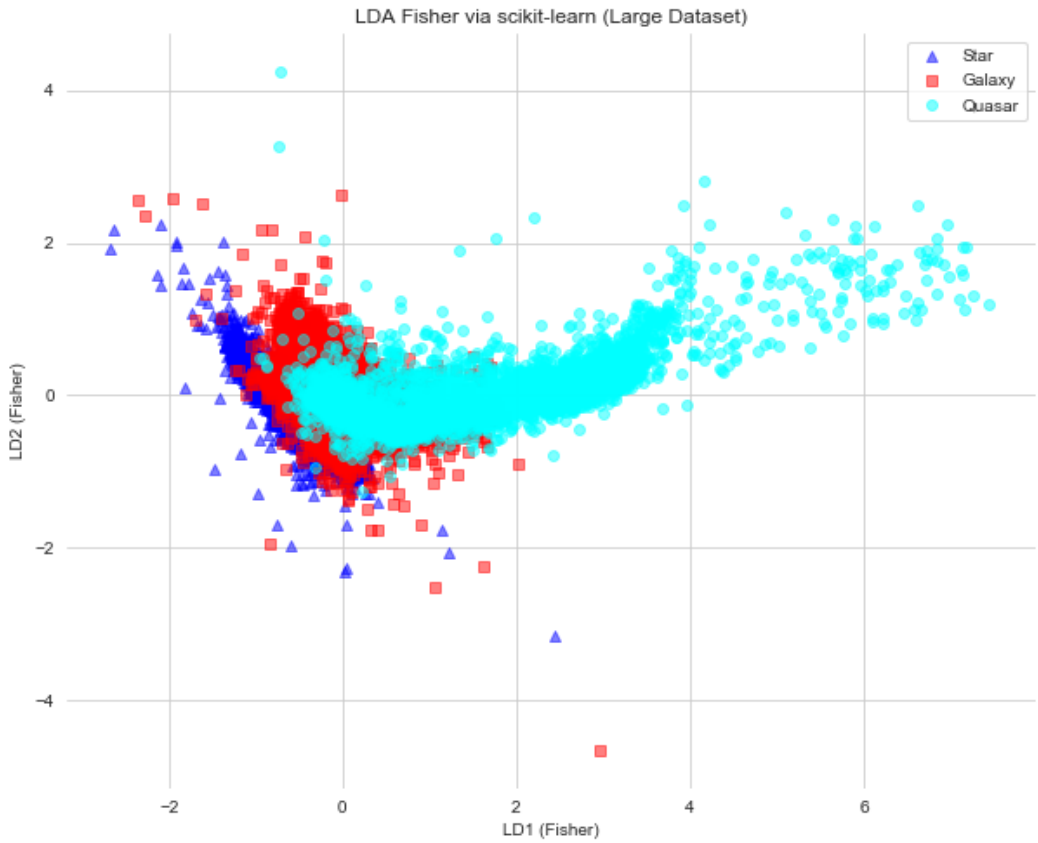
$$a_1(x) = a_3(x)$$

$$\leftrightarrow 0.45 * x_u + 0.71 * x_g - 0.23 * x_r - 0.72 * x_i + 0.02 * x_z - 1.78 * x_{redshift} + 1.81 = 0$$

$$a_2(x) = a_3(x)$$

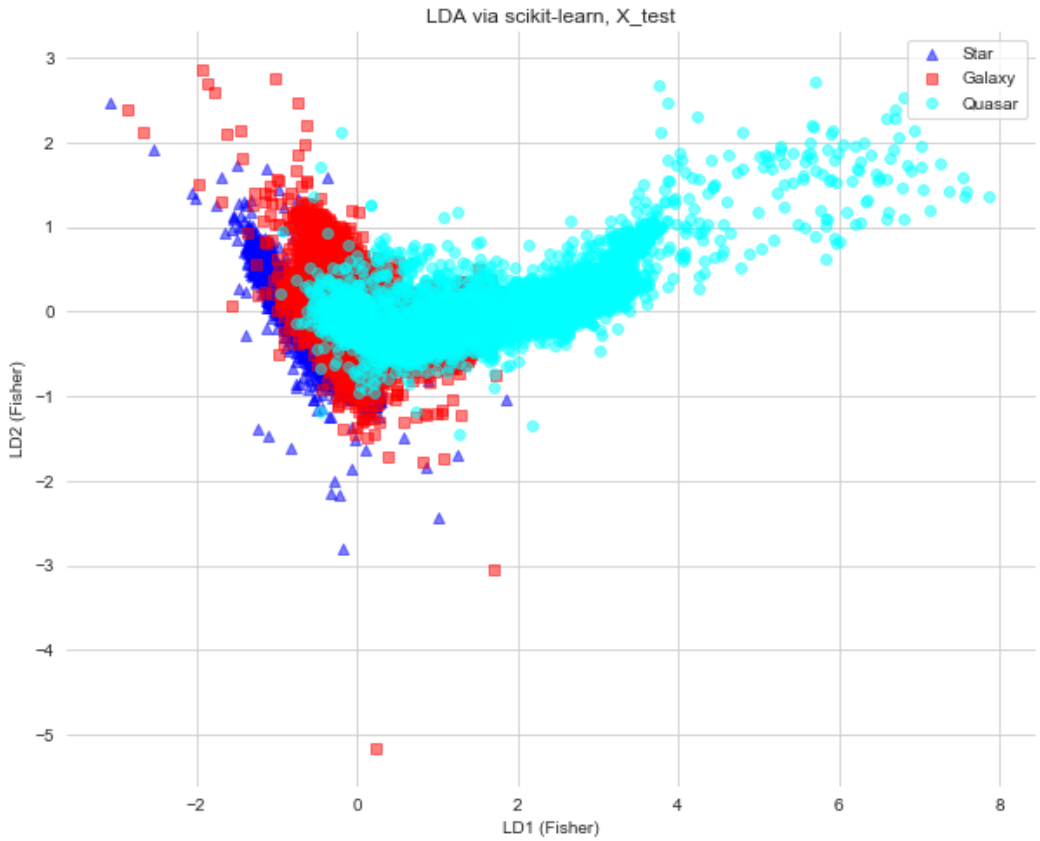
$$\leftrightarrow 0.42 * x_u + 1.04 * x_g - 0.39 * x_r - 0.74 * x_i - 0.10 * x_z - 1.60 * x_{redshift} + 2.90 = 0$$

Εφαρμόζουμε τη μέθοδο του Fisher στο σύνολο εκπαίδευσης και στη συνέχεια στο σύνολο αξιολόγησης. Τα μετασχηματισμένα 2-διάστατα διανύσματα του συνόλου εκπαίδευσης αποτυπώνονται ως εξής:

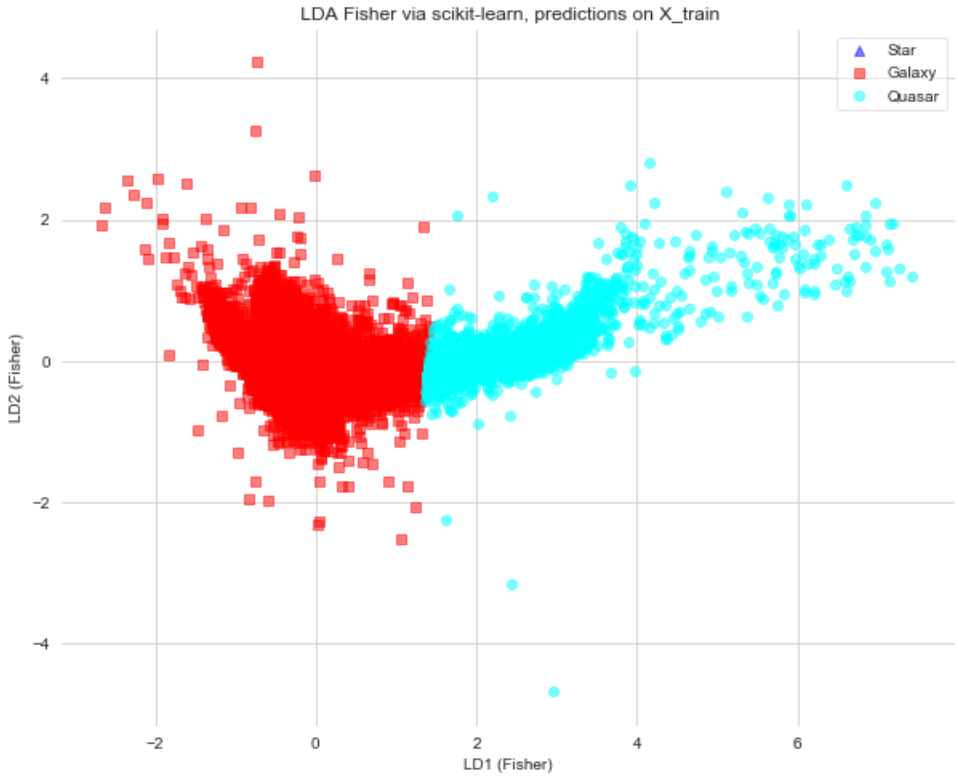


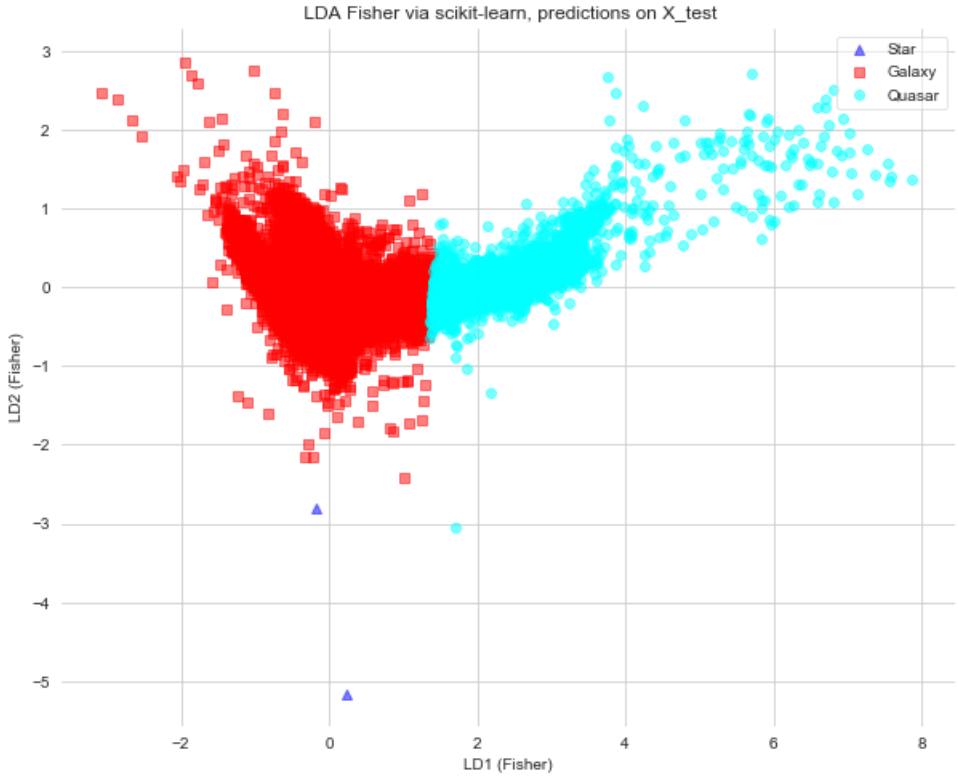
Από το διάγραμμα παρατηρούμε ότι αν προβάλουμε τα σημεία στην διάσταση LD1 (x-άξονας) θα έχουμε αρκετά καλή διαχωρισιμότητα μεταξύ της κλάσης των Αστέρων και των Κβαζαρς ενώ θα έχουμε μεγάλη επικάλυψη μεταξύ Αστέρων - Γαλαξιών και Κβαζαρς - Γαλαξιων. Αν προβάλουμε στη διάσταση LD2 (y-άξονας), τότε δεν θα έχουμε καθόλου καλή διαχωρισιμότητα μεταξύ των 3 κλάσεων. Με την εντολή $sklearn_lda.explained_variance_ratio_$ επιστρέφεται το ποσοστό της πληροφορίας (ή της διασποράς) που προσφέρει η κάθε διάσταση. Για την διάσταση της LD1 έχουμε 87% ενώ για την LD2 έχουμε 13%.

Τα ίδια ποσοστά προκύπτουν και για το σύνολο αξιολόγησης:



Στη συνέχεια απεικονίζονται τα σημεία χρωματισμένα αναλόγως των κλάσεων στις οποίες ταξινομήθηκαν από τον αλγόριθμο:



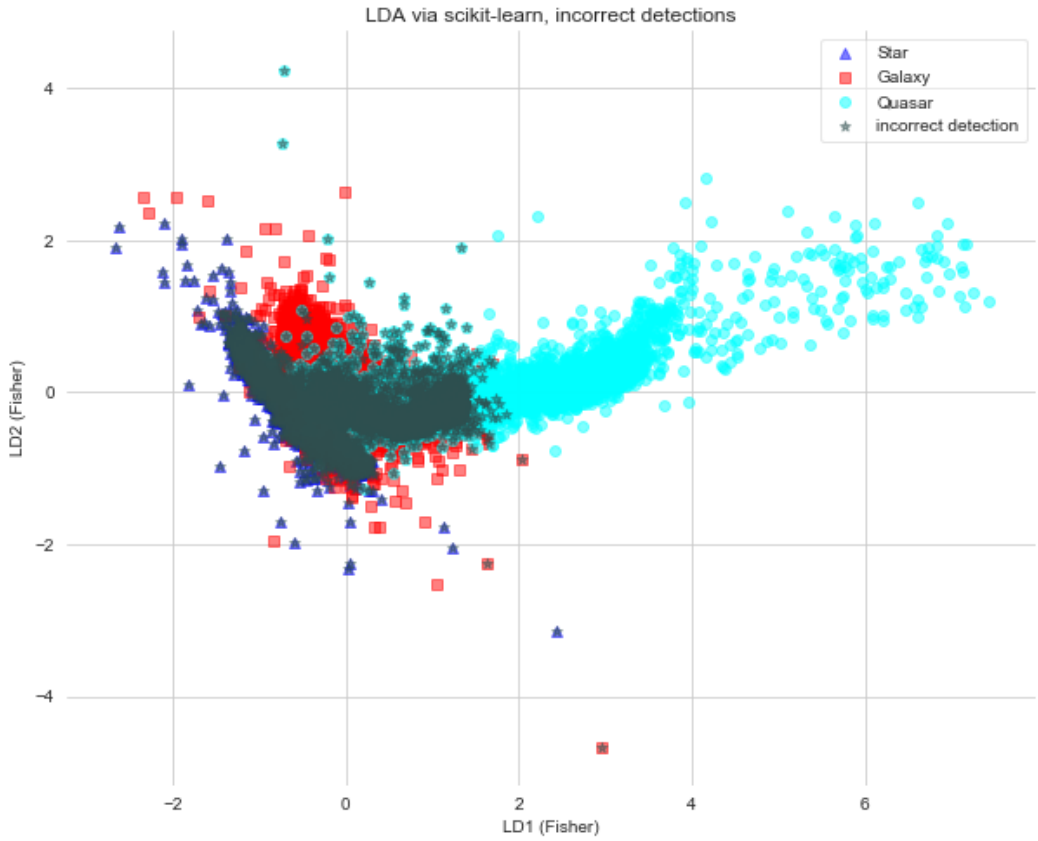


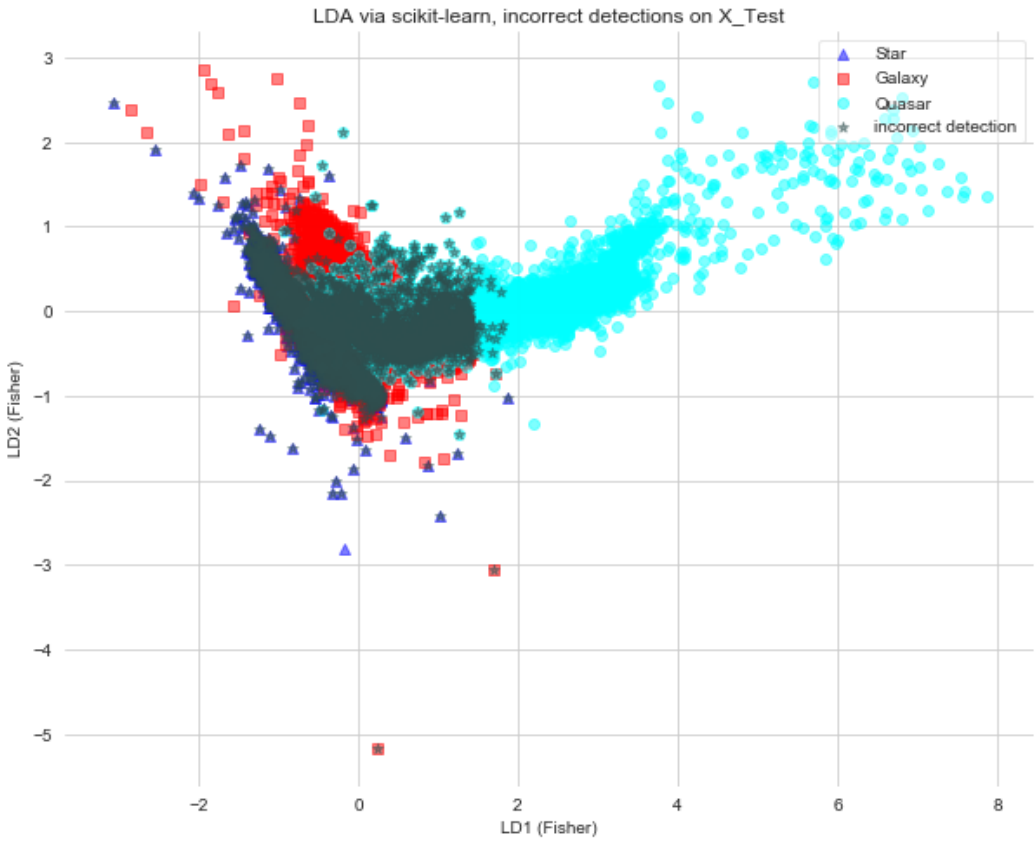
Παρατηρούμε ότι όσο αφορά στα δεδομένα εκπαίδευσης, όλα ταξινομήθηκαν ως Γαλαξίες

και Κβάζαρς, και καμία παρατήρηση δεν ταξινομήθηκε ως Αστέρι. Παρομοίως, στο σύνολο αξιολόγησης όλες οι παρατηρήσεις ταξινομήθηκαν ως Γαλαξίες και Κβάζαρς, εκτός από δύο που ταξινομήθηκαν ως Αστέρες.

Λόγω του ότι τα δεδομένα μας έχουν 6 διαστάσεις, οι επιφάνειες διαχωρισμού που προκύπτουν έχουν επίσης 6 διαστάσεις, οπότε δεν μπορούμε να τα απεικονίσουμε σε ένα 2-διάστατο γράφημα.

Στη συνέχεια απεικονίζονται τα προβεβλημένα σημεία, και με μαύρο αστερίσκο σημαδεύονται αυτά που έχουν ταξινομηθεί σε λανθασμένη κλάση:





Όπως είναι αναμενόμενο, τα περισσότερα λάθος ταξινομημένα σημεία εντοπίζονται στις περιοχές όπου υπάρχουν επικαλύψεις σημείων.

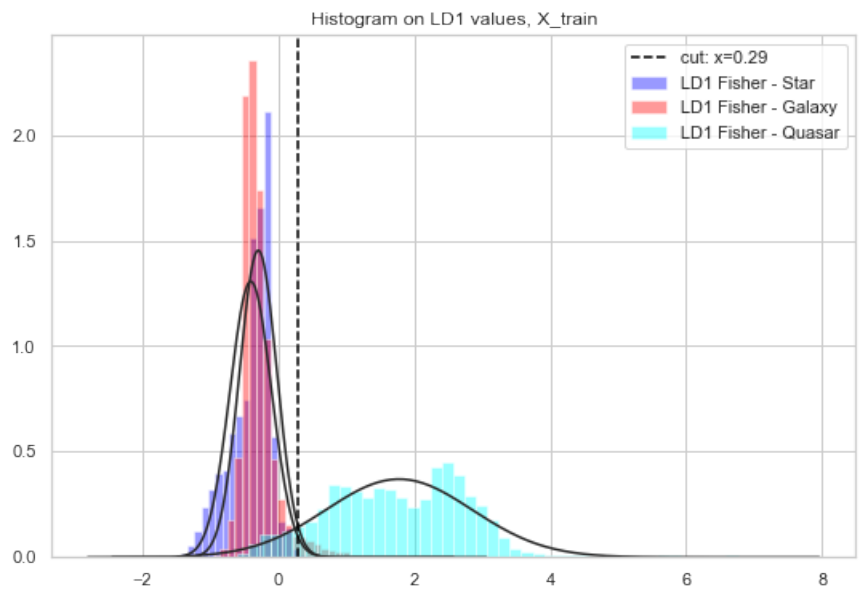
3.5.4 Εναλλακτικός τρόπος ταξινόμησης

Πέρα από τον τρόπο ταξινόμησης που εκτελεί η βιβλιοθήκη του *sklearn* επιχειρήσαμε να ταξινομήσουμε τα δεδομένα και με μία άλλη, απλούστερη μέθοδο. Για να βρούμε τα όρια απόφασης του παραπάνω ταξινομητή προβάλλουμε τα μετασχηματισμένα δεδομένα στην εκάστοτε διάσταση ούτως ώστε από τις μονοδιάστατες κατανομές και βάσει του κανόνα Bayes να βρούμε τα σημεία στα οποία έχουμε την καλύτερη διαχωριστική ικανότητα. Συγκεκριμένα, σχεδιάζουμε σε κάθε διάσταση τα ιστογράμματα των posteriori πιθανοτήτων ($p(w_i|\mathbf{x}) = p(w_i)p(\mathbf{x}|w_i)$), που για να σχεδιαστούν είναι απαραίτητο να θεωρήσουμε ότι τα δεδομένα ακολουθούν κάποια συγκεκριμένη κατανομή, αφού πρέπει να έχουμε τις τιμές της συνάρτησης πιθανοφάνειας. Θεωρούμε ότι τα δεδομένα ακολουθούν Κανονική Κατανομή με ίσους πίνακες διασποράς. Επίσης, επειδή στον ουρανό θεωρούμε ότι υπάρχει ίσος αριθμός Αστέρων, Γαλαξιών και Κβάζαρς, μπορούμε να θεωρήσουμε ότι οι 3 κλάσεις είναι ισοπίθανες (ανεξαρτήτως αν το δείγμα που διαθέτουμε δεν έχει

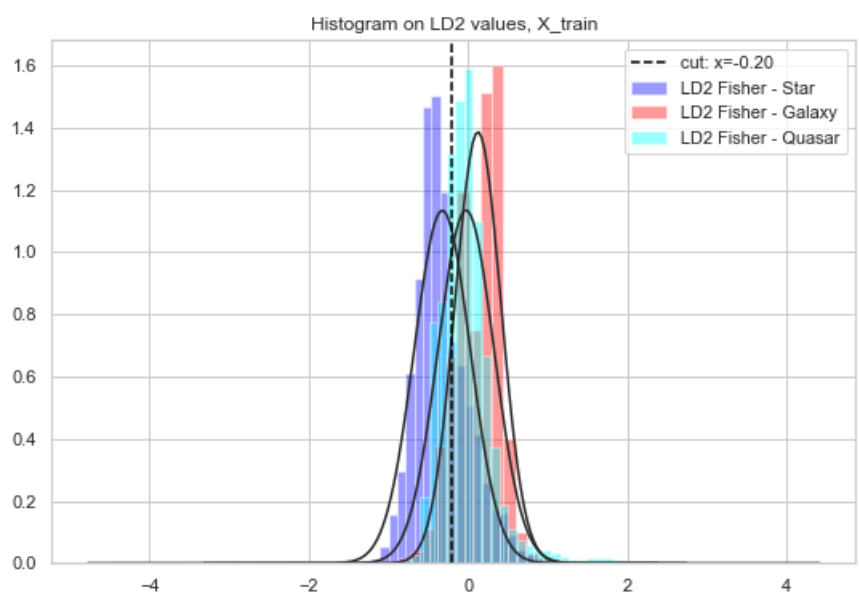
ίσο αριθμό δειγμάτων από την εκάστοτε κλάση), επομένως θεωρούμε $p(C_k) = 1/3,\ k = 1, 2, 3$ (έτσι τα ιστογράμματα της πιθανοφάνειας ταυτίζονται με τα ιστογράμματα της posterior).

Προσαρμόζουμε στα ιστογράμματα καμπύλες Κανονικής Κατανομής και εντοπίζουμε τις περιοχές απόφασης για την κάθε κλάση, δηλαδή τις περιοχές όπου ελαχιστοποιείται το σφάλμα ταξινόμησης. Ακολουθούμε την ίδια διαδικασία για την διάσταση LD1 και LD2 και έτσι χωρίζουμε την περιοχή σε 3 περιοχές απόφασης.

Ιστόγραμμα LD1

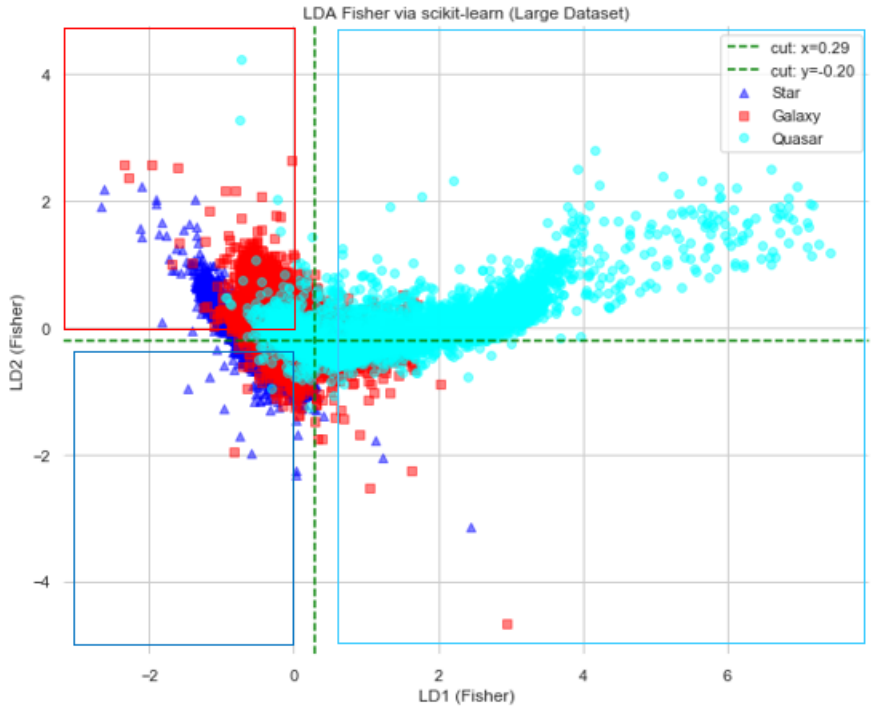


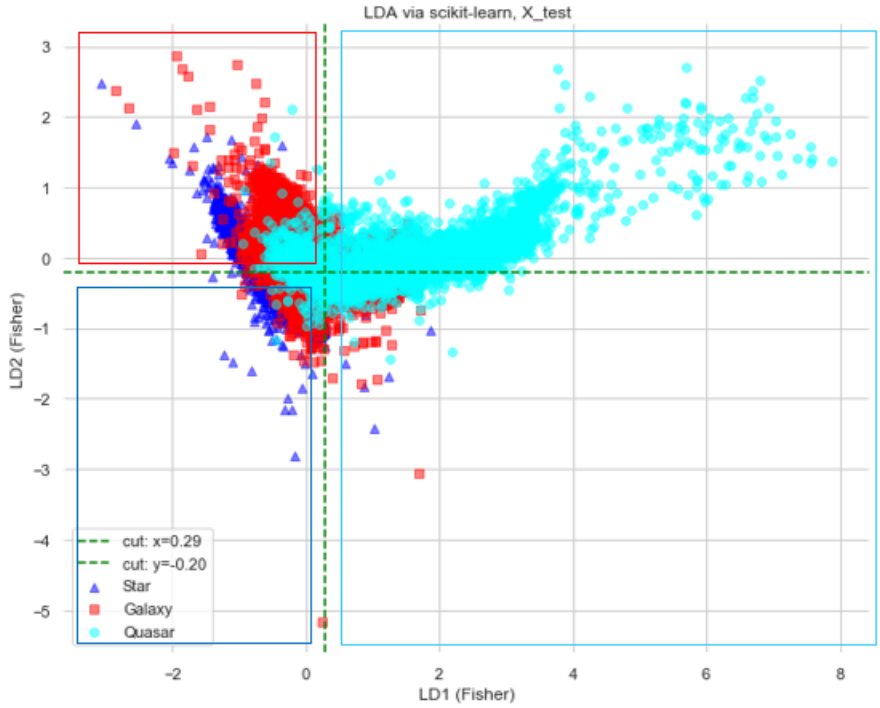
Ιστόγραμμα LD2



Οπότε μπορούμε να θεωρήσουμε ότι τα σημεία ταξινομούνται ως εξής: Αν $\mathrm{LD1}(\mathbf{x}) > 0.29$ τότε το $\mathbf{x}$ κατατάσσεται στα Κβάζαρς. Αλλιώς αν το $\mathrm{LD2}(\mathbf{x}) < -0.20$ τότε το $\mathbf{x}$ κατατάσσεται στους Αστέρες. Αλλιώς κατατάσσεται στους Γαλαξίες.

Δηλαδή χωρίσαμε τον φασικό χώρο στις παρακάτω περιοχές:





3.5.5 Αξιολόγηση των αποτελεσμάτων της μεθόδου Fisher

Παρακάτω παρουσιάζονται τα αποτελέσματα της ταξινόμησης που εκτέλεσε η βιβλιοθήκη *sklearn*.

Αποτελέσματα του Συνόλου Εκπαίδευσης:

F1-scores:

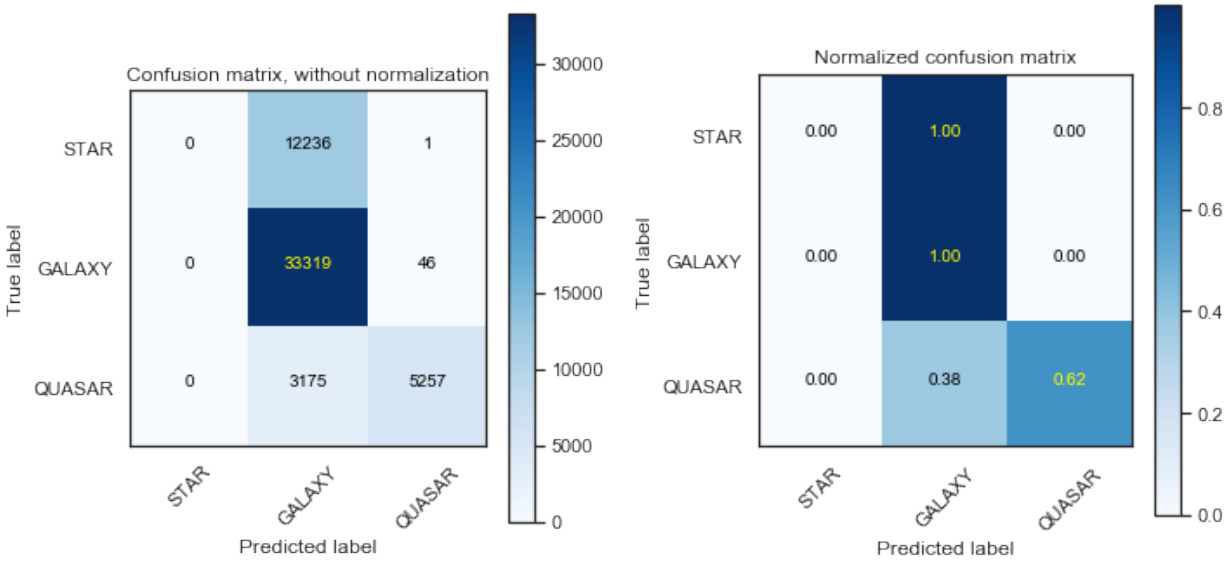
Αστερες: 0

Γαλαξιες: 0.81

Κβαζαρς: 0.77

Log-Loss = 0.65

Confusion matrices:



Συνολικό σφάλμα $= \frac{12236+1+46+3175}{54035} = \frac{15458}{54034} = \mathbf{0.29}$

Αποτελέσματα Συνόλου Αξιολόγησης:

F1-scores:

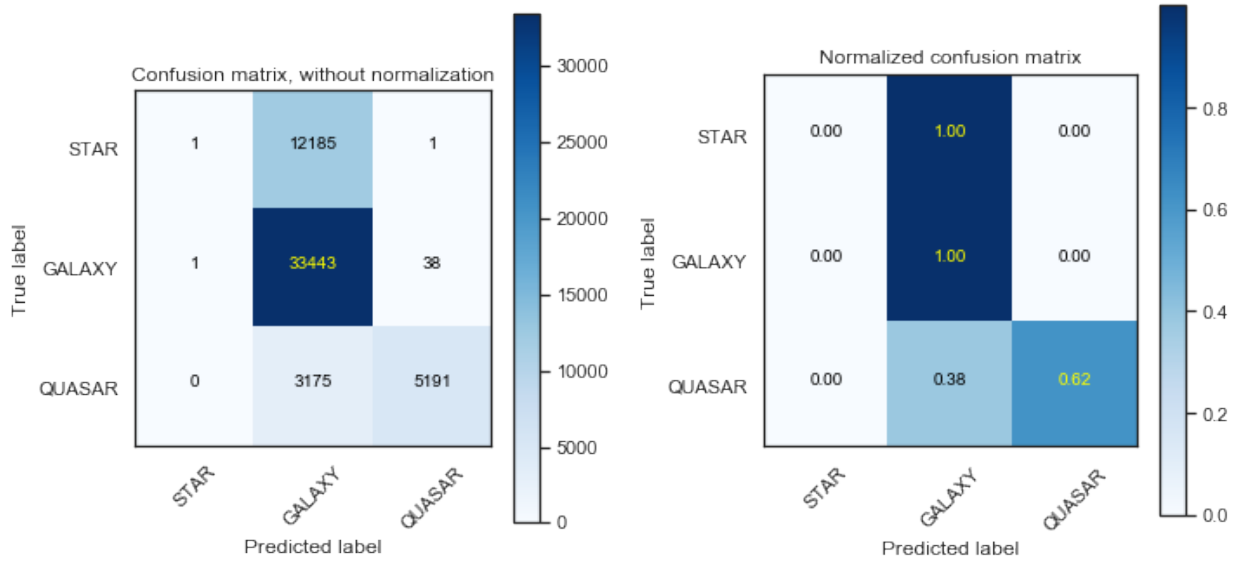
Αστερες: 0

Γαλαξιες: 0.81

Κβαζαρς: 0.76

Log-Loss = 0.64

Confusion matrices:



Συνολικό σφάλμα $= \frac{12185+1+1+38+31}{54035} = \frac{15400}{54035} = \mathbf{0.29}$

Από τα confusion matrices παρατηρούμε ότι το μεγαλύτερο ποσοστό των Γαλαξίων ταξινομήθηκε ορθά. Ωστόσο, υπάρχει μεγάλο περιθώριο βελτίωσης του ταξινομητή, αφού ο συγκεκριμένος κατέταξε μόνο έναν από τους Αστέρες ορθά (στο σύνολο αξιολόγησης) ενώ η κλάση των Κβάζαρς παρουσιάζει επίσης υψηλά σφάλματα, της τάξης του 38%.

Συνεχίζουμε παρουσιάζοντας τα Confusion matrices του εναλλακτικού τρόπου ταξινόμησης:

Αποτελέσματα Συνόλου Εκπαίδευσης:

Πραγματική κλάση		ΑΣΤΕΡΙΑ	ΓΑΛΑΞΙΕΣ	ΚΒΑΖΑΡΣ
	ΑΣΤΕΡΙΑ	8626	3598	13
	ΓΑΛΑΞΙΕΣ	3640	28260	1465
	ΚΒΑΖΑΡΣ	251	445	7736
		ΑΣΤΕΡΙΑ	ΓΑΛΑΞΙΕΣ	ΚΒΑΖΑΡΣ
		Προβλεπόμενη κλάση		

Πραγματική κλάση		ΑΣΤΕΡΙΑ	ΓΑΛΑΞΙΕΣ	ΚΒΑΖΑΡΣ
	ΑΣΤΕΡΙΑ	$\frac{8626}{12237} = 0.70$	$\frac{3598}{12237} = 0.29$	$\frac{13}{12237} \simeq 0$
	ΓΑΛΑΞΙΕΣ	$\frac{3640}{33365} = 0.11$	$\frac{28260}{33365} = 0.85$	$\frac{1465}{33365} = 0.04$
	ΚΒΑΖΑΡΣ	$\frac{251}{8432} = 0.03$	$\frac{445}{8432} = 0.05$	$\frac{7736}{8432} = 0.92$
		ΑΣΤΕΡΙΑ	ΓΑΛΑΞΙΕΣ	ΚΒΑΖΑΡΣ
		Προβλεπόμενη κλάση		

Συνολικό σφάλμα $= \frac{3598+13+3640+1465+251+445}{54034} = \frac{9412}{54034} = \mathbf{0.17}$

Αποτελέσματα Συνόλου Αξιολόγησης:

Πραγματική κλάση		ΑΣΤΕΡΙΑ	ΓΑΛΑΞΙΕΣ	ΚΒΑΖΑΡΣ
	ΑΣΤΕΡΙΑ	8553	3623	11
	ΓΑΛΑΞΙΕΣ	3598	28388	1496
	ΚΒΑΖΑΡΣ	254	418	7694
		ΑΣΤΕΡΙΑ	ΓΑΛΑΞΙΕΣ	ΚΒΑΖΑΡΣ
		Προβλεπόμενη κλάση		

Πραγματική κλάση		ΑΣΤΕΡΙΑ	ΓΑΛΑΞΙΕΣ	ΚΒΑΖΑΡΣ
	ΑΣΤΕΡΙΑ	$\frac{8553}{12187} = 0.70$	$\frac{3623}{12187} = 0.30$	$\frac{11}{12187} \simeq 0$
	ΓΑΛΑΞΙΕΣ	$\frac{3598}{33482} = 0.11$	$\frac{28388}{33482} = 0.85$	$\frac{1496}{33482} = 0.04$
	ΚΒΑΖΑΡΣ	$\frac{254}{8366} = 0.03$	$\frac{418}{8366} = 0.05$	$\frac{7694}{8366} = 0.92$
		ΑΣΤΕΡΙΑ	ΓΑΛΑΞΙΕΣ	ΚΒΑΖΑΡΣ
		Προβλεπόμενη κλάση		

Συνολικό σφάλμα $= \frac{3623+11+3598+1496+254+418}{54035} = \frac{9400}{54035} = \mathbf{0.17}$

Παρατηρούμε ότι εάν ταξινομήσουμε τις παρατηρήσεις χρησιμοποιώντας τον εναλλακτικό τρόπο ταξινόμησης, το συνολικό σφάλμα τόσο στο σύνολο εκπαίδευσης όσο και στο σύνολο αξιολόγησης μειώνεται σχεδόν στο μισό αφού από 29% πέφτει στο 17%.

Θα επιδιώξουμε να βελτιώσουμε την απόδοση του γραμμικού ταξινομητή εφαρμόζοντας μία άλλη μέθοδο LDA την οποία θα αποκαλούμε LDA μέσω SVD ή LDA+SVD

3.6 Μέθοδος 2: LDA+SVD

Τόσο από τους πίνακες συσχετίσεων, όσο και από τα pairplots, παρατηρούμε ότι υπάρχει έντονη συσχέτιση μεταξύ των μεταβλητών u, g, r, i, z (οπότε παραβιάζεται η δεύτερη από τις προϋποθέσεις που αναφέρθηκαν στην εισαγωγή). Η μέθοδος LDA+SVD επιλύει αυτό το πρόβλημα αφού στο πρώτο στάδιο με τη βοήθεια της μεθόδου SVD όλες οι μεταβλητές του προβλήματος μετατρέπονται σε στατιστικά ανεξάρτητες (δηλαδή εφαρμόζει decorrelation στις μεταβλητές). Άρα μέσω αυτής της μεθόδου **λύνουμε το πρόβλημα των συσχετίσεων** που υπάρχουν. Στο δεύτερο στάδιο εφαρμόζεται η μέθοδος LDA που περιγράψαμε νωρίτερα, στον νέο (μετασχηματισμένο) πίνακα δεδομένων X^*, για μείωση των διαστάσεων και για την ταξινόμηση των παρατηρήσεων.

Πλεονέκτημα:
Σε προβλήματα όπου ο αριθμός των χαρακτηριστικών είναι μεγάλος, η μέθοδος LDA+SVD πλεονεκτεί συγκριτικά με την μέθοδο Fisher διότι δεν βασίζεται στον υπολογισμό του πίνακα συνδιασποράς. Στην περίπτωση του προβλήματός μας, αν και ο αριθμός των χαρακτηριστικών είναι μόνο 6, όπως θα δούμε στη συνέχεια, ο αλγόριθμος LDA+SVD έχει αρκετά υψηλότερη απόδοση συγκριτικά με τον αλγόριθμο του Fisher.

3.6.1 Θεωρία LDA+SVD

Αρχίζουμε με το παρακάτω θεώρημα, το οποίο παίζει σημαντικό ρόλο στην κατανόηση της θεωρίας. Το θεώρημα ισχύει αν οι μεταβλητές $\mathbf{x_i}$ έχουν μηδενική μέση τιμή.

Θεώρημα:
Έστω ότι $X = U\Lambda V^T$ είναι η αποσύνθεση SVD του πίνακα X_{Nxd} , (δηλαδή η αναγωγή του σε γινόμενο τριών πινάκων U, Λ, V όπου U, V ορθομοναδιαίοι και Λ διαγώνιος. Ο Λ στη διαγώνιο του περιλαμβάνει τα $\sqrt{\lambda_i}$, όπου λ_i οι μη μηδενικές ιδιοτιμές του πίνακα X^TX). Ο πίνακας διασποράς Σ του πίνακα X,

$$\Sigma = \frac{1}{N}X^TX$$

έχει τα ίδια ιδιοδιανύσματα με τον πίνακα V και οι ιδιοτιμές του, λ_i, σχετίζονται με τις ιδιόμορφες τιμές (singular values) σ_i του X μέσω της σχέσης

$$\lambda_i = \frac{\sigma_i^2}{N}$$

Υποθέτουμε ότι όλες οι κλάσεις έχουν το ίδιο πίνακα διασποράς Σ. Σκοπός είναι να μετασχηματίσουμε τον πίνακα X, με τέτοιο τρόπο ώστε ο αντίστοιχος πίνακας διασποράς Σ να είναι

ο ταυτοτικός πίνακας, I, δηλαδή οι διασπορές θα ισούνται με 1 και οι συνδιασπορές με 0, οπότε δεν θα υπάρχουν συσχετίσεις μεταξύ των μετασχηματισμένων τυχαίων μεταβλητών αφού

$$Corr(x, y) = \frac{Cov(x, y)}{\sigma_x \sigma_y}$$

Για να το πετύχουμε, μετασχηματίζουμε τον πίνακα X πολλαπλασιάζοντας από αριστερά με τον πίνακα:

$$W_{PCA} = D^{-1/2} U^T$$

Ο παραπάνω πίνακας εκτελεί PCA Whiting transformation (ή sphering transformation). Το Whitening transformation είναι ένας γραμμικός μετασχηματισμός που εκτελούμε σε κάποιο πίνακα τυχαίων μεταβλητών με γνωστό πίνακα διασποράς Σ, όταν θέλουμε να τον μετασχηματίσουμε σε ένα νέο πίνακα με πίνακα διασποράς τον ταυτοτικό (I).

Ο νέος πίνακας X^* δίνεται από τον τύπο:

$$X^* = W_{PCA} X = D^{-1/2} U^T X$$

όπου ο πίνακας D περιέχει στη διαγώνιό του τις ιδιοτιμές λ_i του πίνακα διασποράς $\Sigma = \frac{1}{N} X^T X$ και ο πίνακας U έχει ως στήλες τα ιδιοδιανύσματα του XX^T

Στη συνέχεια εκτελείται η μέθοδος μείωσης των διαστάσεων της LDA και η ταξινόμηση των παρατηρήσεων βάσει της Θεωρίας Bayes που περιγράψαμε προηγουμένως.

3.6.2 Υλοποίηση

Τρέξαμε την LDA με τη βοήθεια της βιβλιοθήκης της Python

$$sklearn.discriminant_analysis.LinearDiscriminantAnalysis$$

με ορίσματα:

$$(solver = svd, n_components = 2, tol = 0.0001)$$

Επιλογές παραμέτρων που διαθέτει η βιβλιοθήκη *sklearn*:

- **solver**:

  Δοκιμάσαμε solver=svd, ο οποίος εκτελεί την μέθοδο LDA+SVD.

- **n_components**:

  Θέτουμε να είναι ίσο με 2 ούτως ώστε να μειώσουμε την διάσταση του προβλήματος από 6 σε $K - 1 = 2$.

- **priors** :

  Ορίζουμε ένα διάνυσμα με τις priori πιθανότητες, δηλαδή με το ποσοστό των διανυσμάτων που ανήκουν σε κάθε κλάση.

3.6.3 Εκτέλεση και Αποτελέσματα

Με την εκτέλεση της μεθόδου, επιστρέφονται τα μετασχηματισμένα 2-διάστατα διανύσματα του συνόλου δεδομένων.
Με την εντολή $lda.scalings_$ δίνονται οι συντελεστές των linear discriminant functions. Οι τιμές των συναρτήσεων είναι τα προβεβλημένα σημεία που έχουμε στα διαγράμματα που ακολουθούν, δηλαδή τα 2-διάστατα μετασχηματισμένα διανύσματα.
Συγκεκριμένα για το πρόβλημά μας έχουμε:

$$LD_1 = -0.41 * x_u - 0.88 * x_g + 0.31 * x_r + 0.69 * x_i + 0.06 * x_z + 1.58 * x_{redshift}$$

$$LD_2 = 0.09 * x_u - 2.67 * x_g + 1.26 * x_r - 0.39 * x_i + 0.89 * x_z - 0.7 * x_{redshift}$$

οπότε έτσι φαίνεται το πόσο συμβάλει κάθε μεταβλητή στον υπολογισμό των γραμμικών συνδυασμών LD1 και LD2.

Παρατηρούμε ότι στην LD1 μεγαλύτερο κατά απόλυτη τιμή συντελεστή έχουν οι μεταβλητές g και $redshift$ ενώ στην LD2 μεγαλύτερους συντελεστές έχουν οι μεταβλητές g και r. Αυτό σημαίνει ότι οι συγκεκριμένες μεταβλητές έχουν την καλύτερη διαχωριστική ικανότητα στον κάθε ένα από τους γραμμικούς συνδυασμούς, αντίστοιχα. Επίσης στην LD1 οι μεταβλητές u και g έχουν αρνητικούς συντελεστές ενώ οι υπόλοιπες έχουν θετικούς συντελεστές, δηλαδή το discriminant function αναπαριστά μία αντίθεση στη συμβολή των μεταβλητών με αρνητικούς και θετικούς συντελεστές. Στην LD2 οι μεταβλητές g και $redshift$ έχουν αρνητική συμβολή στην τιμή της συνάρτησης, ενώ οι υπόλοιπες μεταβλητές συμβάλουν θετικά.

Για την ταξινόμηση των παρατηρήσεων στις 3 κλάσεις, η LDA υπολογίζει 3 επιφάνειες διαχωρισμού, δηλαδή 3 γραμμικές συναρτήσεις της μορφής:

$$a_k(\mathbf{x}) = \mathbf{v}_k^T \mathbf{x} + v_{k_0}$$

για $k = 1, 2, 3$
Συγκεκριμένα για το πρόβλημά μας έχουμε

$$a_1(x) = 0.42 * x_u - 2.03 * x_g + 1.04 * x_r - 0.17 * x_i + 0.87 * x_z - 2.02 * x_{redshift} - 2.35$$

$$a_2(x) = 0.21 * x_u + 1.59 * x_g - 0.69 * x_r - 0.57 * x_i - 0.39 * x_z - 0.67 * x_{redshift} - 0.74$$

$$a_3(x) = -1.45 * x_u - 3.33 * x_g + 1.21 * x_r + 2.49 * x_i + 0.28 * x_z + 5.57 * x_{redshift} - 8.18$$

από τα οποία προκύπτουν οι τρεις επιφάνειες διαχωρισμού:

$$a_1(x) = a_2(x)$$

$$\leftrightarrow 0.21 * x_u + 3.62 * x_g + 1.73 * x_r + 0.40 * x_i + 1.26 * x_z - 1.35 * x_{redshift} - 1.61 = 0$$

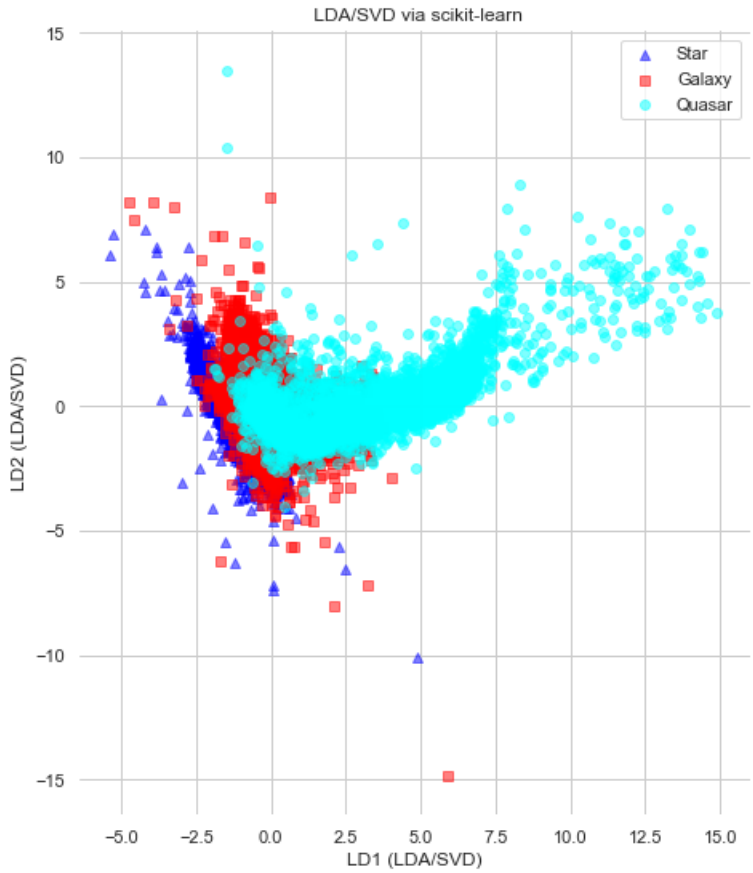
$$a_1(x) = a_3(x)$$

$$\leftrightarrow 1.87 * x_u + 1.30 * x_g - 0.17 * x_r - 2.66 * x_i + 0.59 * x_z - 7.59 * x_{redshift} + 5.83 = 0$$

$$a_2(x) = a_3(x)$$

$$\leftrightarrow 1.66 * x_u + 4.29 * x_g - 1.90 * x_r - 3.06 * x_i - 0.67 * x_z - 6.24 * x_{redshift} + 7.44 = 0$$

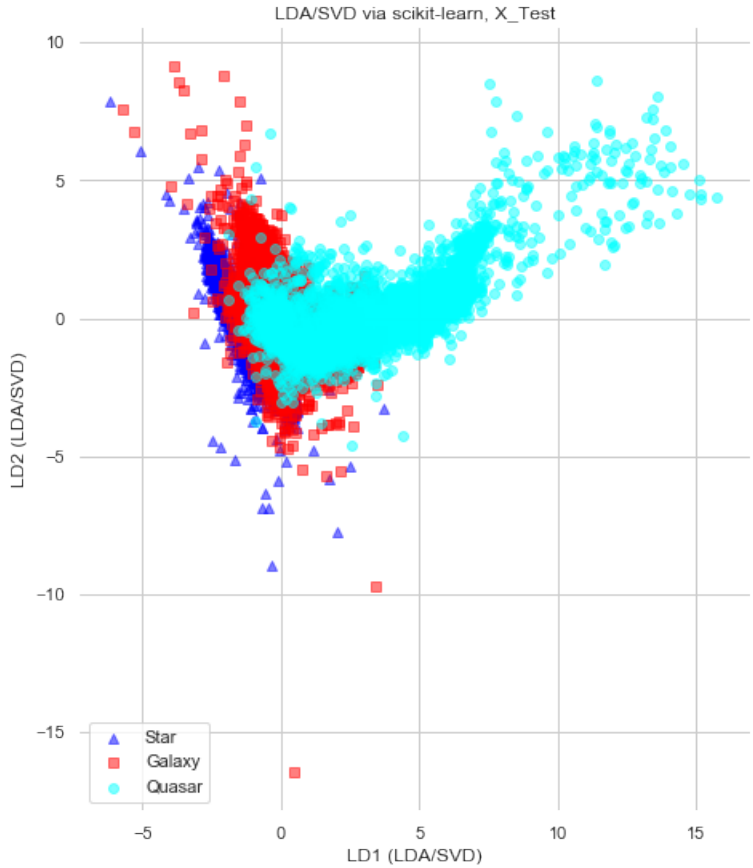
Εφαρμόζουμε τη μέθοδο LDA+SVD στο σύνολο εκπαίδευσης και στη συνέχεια στο σύνολο αξιολόγησης. Τα μετασχηματισμένα 2-διάστατα διανύσματα του προβλήματος αποτυπώνονται ως εξής:



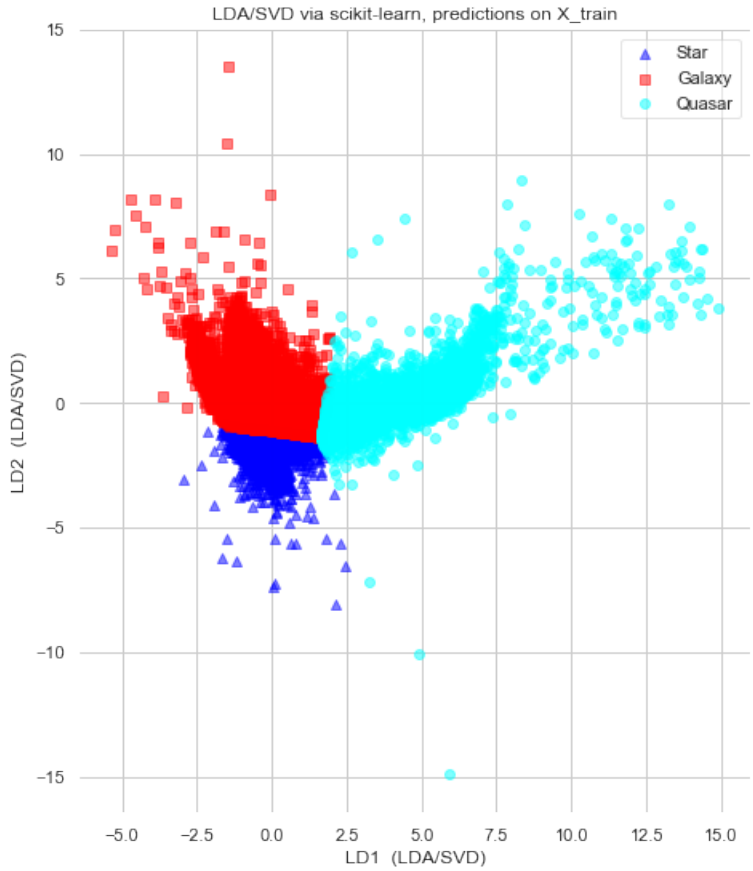
Από το διάγραμμα παρατηρούμε ότι αν προβάλουμε τα σημεία στην διάσταση LD1 (x-άξονας) θα έχουμε αρκετά καλή διαχωρισιμότητα μεταξύ της κλάσης των Αστέρων και των Κβαζαρς ενώ θα έχουμε μεγάλη επικάλυψη μεταξύ Αστέρων - Γαλαξιών και Κβαζαρς - Γαλαξιων. Αν

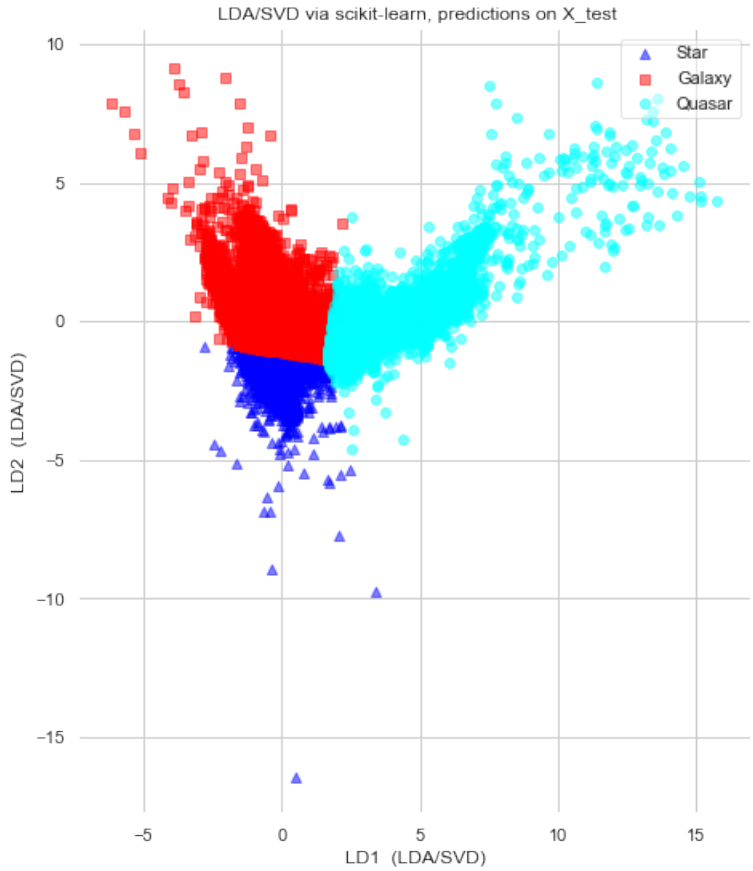
προβάλουμε στη διάσταση LD2 (y-άξονας), τότε δεν θα έχουμε καθόλου καλή διαχωρισιμότητα μεταξύ των 3 κλάσεων. Πιο συγκεκριμένα, στη διάσταση της LD1 έχουμε 87% πληροφορία ενώ για στην LD2 έχουμε 13% πληροφορία.

Το ίδιο ισχύει και για το σύνολο αξιολόγησης:



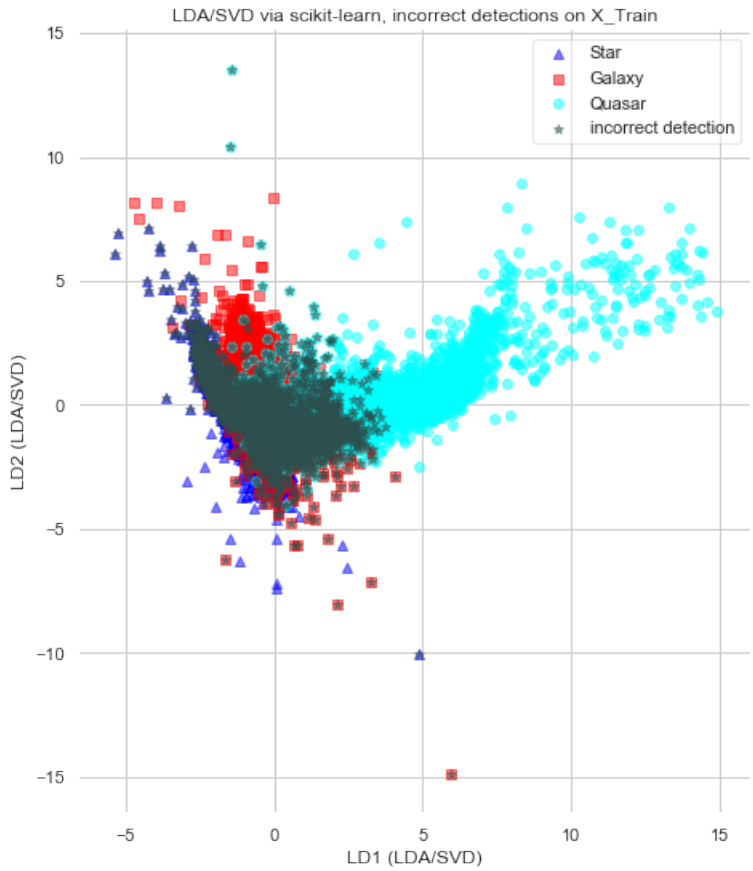
Στη συνέχεια απεικονίζονται τα σημεία χρωματισμένα βάσει των κλάσεων που ταξινομήθηκαν από τον αλγόριθμο:

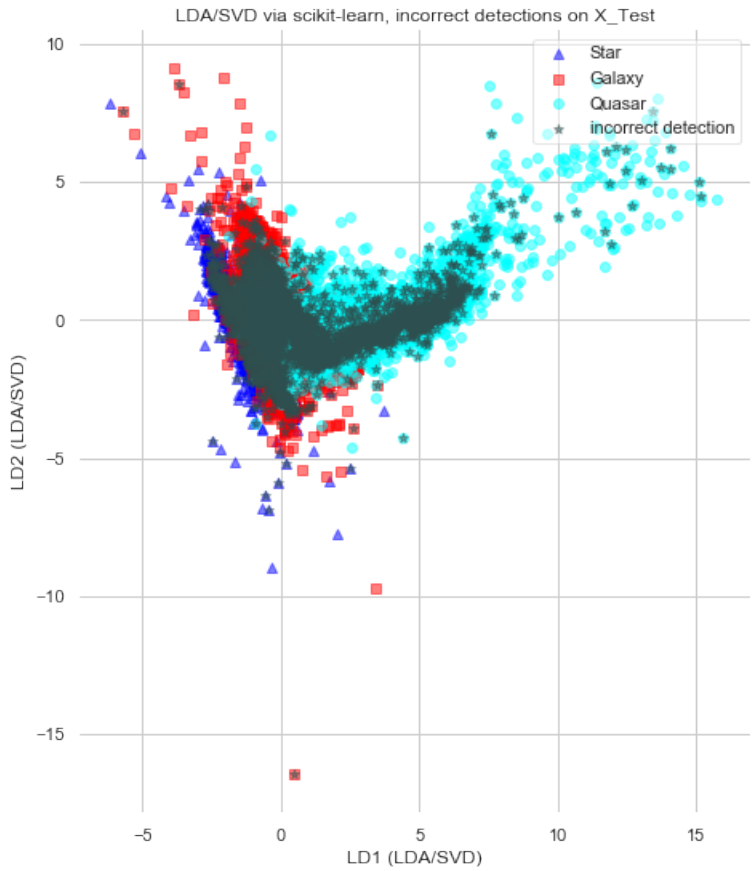

LDA/SVD via scikit-learn, predictions on X_train
Star
Galaxy
Quasar
15
10
5
0
−5
−10
−15
LD2 (LDA/SVD)
−5.0
−2.5
0.0
2.5
5.0
7.5
10.0
12.5
15.0
LD1 (LDA/SVD)



Όπως ήδη αναφέρθηκε, λόγω του ότι τα δεδομένα μας έχουν 6 διαστάσεις, οι επιφάνειες διαχωρισμού που προκύπτουν έχουν επίσης 6 διαστάσεις, οπότε δεν μπορούμε να τα απεικονίσουμε σε ένα 2-διάστατο γράφημα.

Στη συνέχεια απεικονίζονται τα προβεβλημένα σημεία, και με μαύρο αστερίσκο σημαδεύονται αυτά που έχουν ταξινομηθεί σε λανθασμένη κλάση:


LDA/SVD via scikit-learn, incorrect detections on X_Train
Star
Galaxy
Quasar
incorrect detection
LD2 (LDA/SVD)
LD1 (LDA/SVD)
15
10
5
0
−5
−10
−15
−5
0
5
10
15

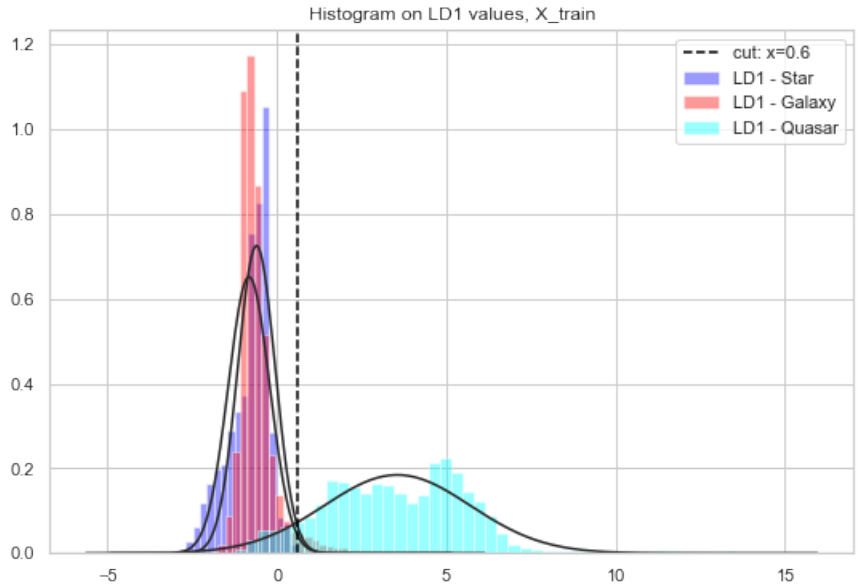


Τα περισσότερα λάθος ταξινομημένα σημεία εντοπίζονται στις περιοχές όπου υπάρχουν επικαλύψεις σημείων. Στο σύνολο αξιολόγησης παρατηρείται μια ουρά λάθος ταξινομημένων κβάζαρς, κάτι που δεν υπάρχει στο διάγραμμα του συνόλου εκπαίδευσης.

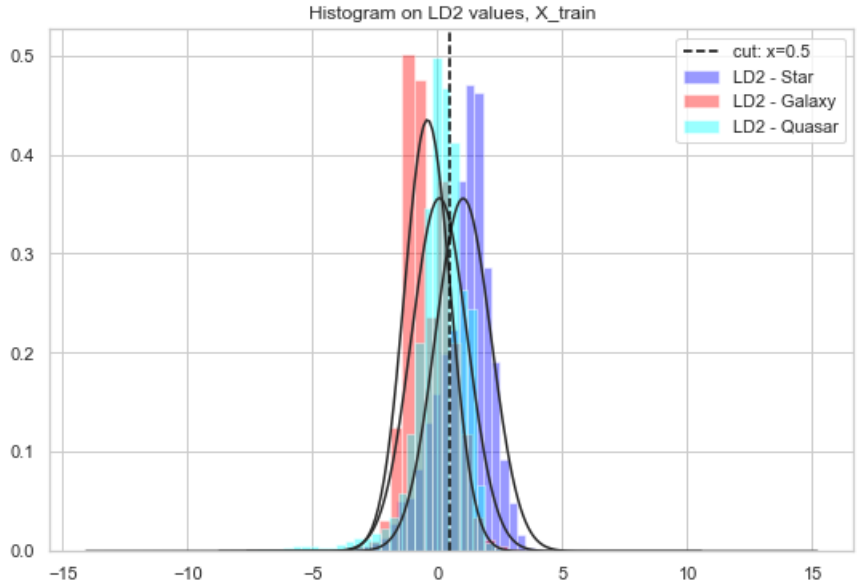
3.6.4 Εναλλακτικός τρόπος ταξινόμησης

Συνεχίζουμε παρουσιάζοντας τον 2ο τρόπο με τον οποίο επιχειρήσαμε να ταξινομήσουμε τα διανύσματα (ο οποίος περιγράφεται στο κεφάλαιο του Fisher).

Ιστόγραμμα LD1

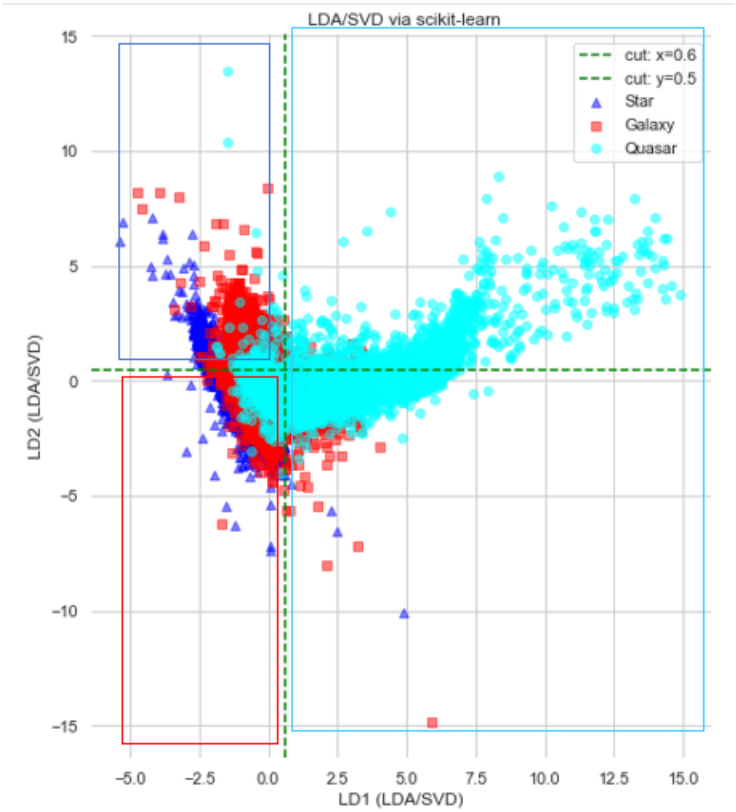


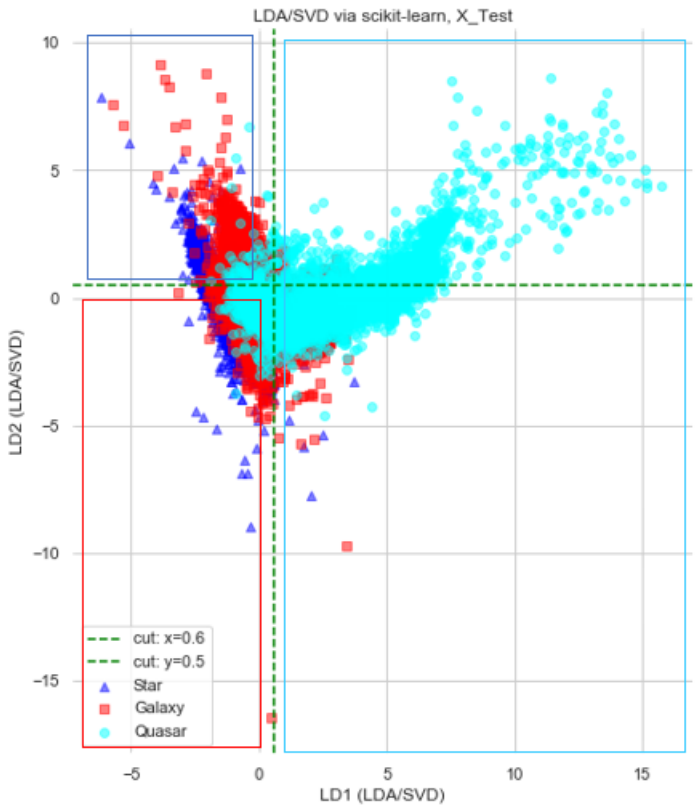
Ιστόγραμμα LD2



Οπότε τα σημεία ταξινομούνται ως εξής: Αν $\mathrm{LD1}(\mathbf{x}) > 0.6$ τότε το $\mathbf{x}$ κατατάσσεται στα Κβάζαρς. Αλλιώς αν το $\mathrm{LD2}(\mathbf{x}) < 0.5$ τότε το $\mathbf{x}$ κατατάσσεται στους Γαλαξίες. Αλλιώς κατατάσσεται στους Αστέρες.

Δηλαδή χωρίσαμε τον φασικό χώρο στις παρακάτω περιοχές:





3.6.5 Αξιολόγηση των αποτελεσμάτων της μεθόδου LDA+SVD

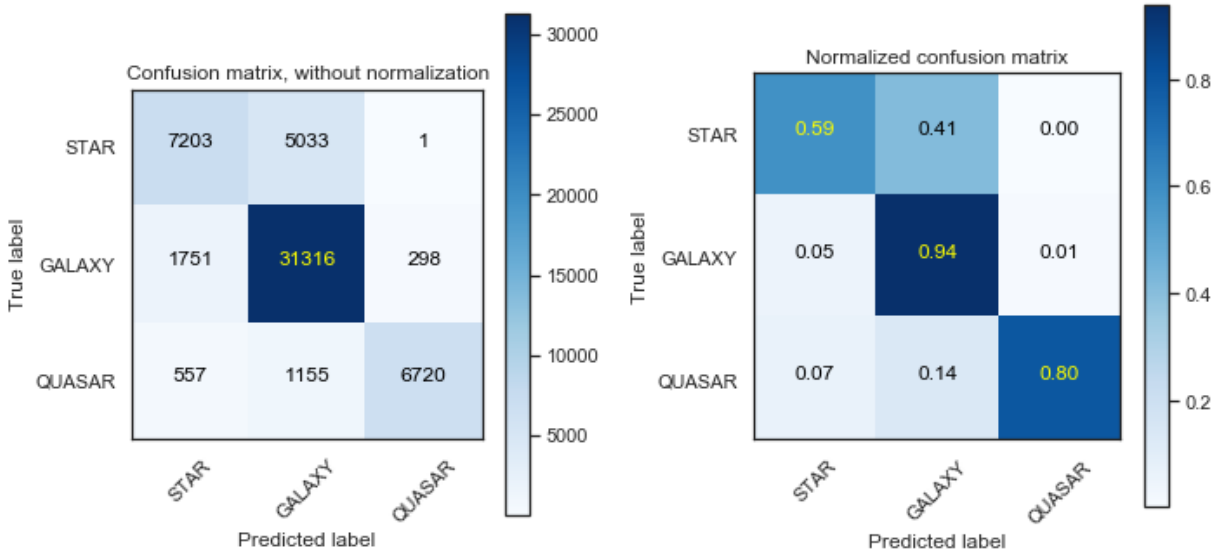
Για την περεταίρω μελέτη και αξιολόγηση των αποτελεσμάτων χρησιμοποιούμε τα μέτρα και τους πίνακες αξιολόγησης που περιγράψαμε προηγουμένως:

Αποτελέσματα Συνόλου Εκπαίδευσης:

F1-scores:
Αστερες: 0.66
Γαλαξιες: 0.88
Κβαζαρς: 0.87

Log-Loss = 0.51

Confusion matrices:



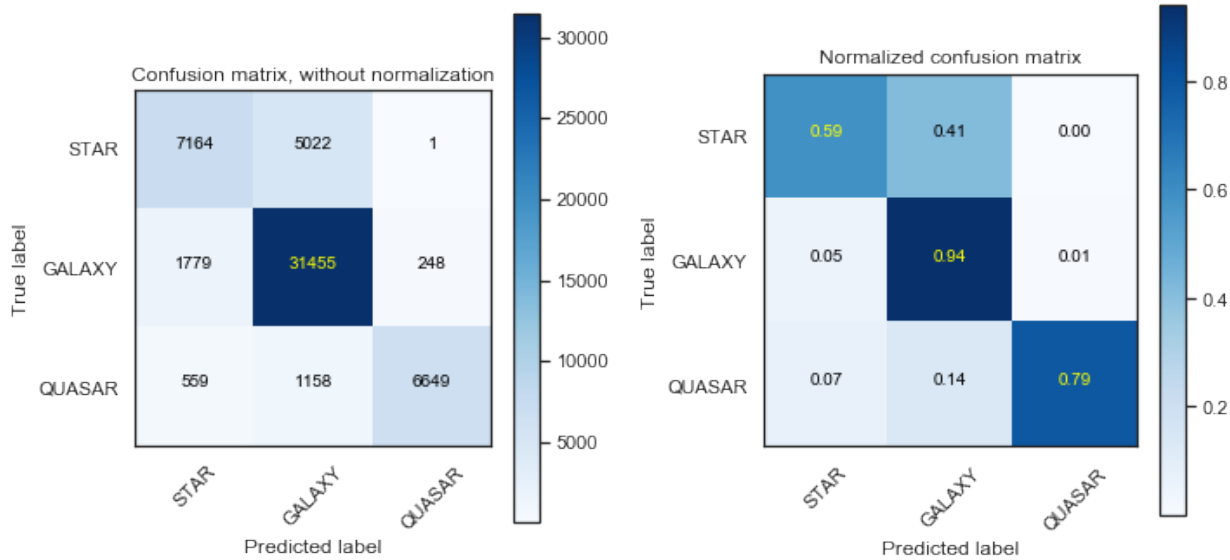
Συνολικό σφάλμα $= \frac{5033+1+1751+298+557+1155}{54034} = \frac{8795}{54034} = \mathbf{0.16}$

Αποτελέσματα Συνόλου Αξιολόγησης:

<u>F1-scores:</u>
Αστερες: 0.66
Γαλαξιες: 0.88
Κβαζαρς: 0.87

<u>Log-Loss</u> = 0.50

<u>Confusion matrices:</u>



Συνολικό σφάλμα $= \frac{5022+1+1779+248+569+1158}{54035} = \frac{8777}{54035} = \mathbf{0.16}$

Συνεχίζουμε παρουσιάζοντας τα Confusion matrices που δημιουργήσαμε για τον εναλλακτικό τρόπο ταξινόμησης.

Αποτελέσματα του Συνόλου Εκπαίδευσης:

Πραγματική κλάση		ΑΣΤΕΡΙΑ	ΓΑΛΑΞΙΕΣ	ΚΒΑΖΑΡΣ
	ΑΣΤΕΡΙΑ	8959	3267	11
	ΓΑΛΑΞΙΕΣ	4543	27389	1433
	ΚΒΑΖΑΡΣ	315	397	7720
		ΑΣΤΕΡΙΑ	ΓΑΛΑΞΙΕΣ	ΚΒΑΖΑΡΣ
		Προβλεπόμενη κλάση		

Πραγματική κλάση		ΑΣΤΕΡΙΑ	ΓΑΛΑΞΙΕΣ	ΚΒΑΖΑΡΣ
	ΑΣΤΕΡΙΑ	0.73	0.27	0
	ΓΑΛΑΞΙΕΣ	0.14	0.82	0.04
	ΚΒΑΖΑΡΣ	0.04	0.05	0.92
		ΑΣΤΕΡΙΑ	ΓΑΛΑΞΙΕΣ	ΚΒΑΖΑΡΣ
		Προβλεπόμενη κλάση		

Συνολικό σφάλμα $= \frac{3267+11+4543+1433+315+397}{54034} = \frac{9966}{54034} = \mathbf{0.18}$

Αποτελέσματα του Συνόλου Αξιολόγησης:

Πραγματική κλάση		ΑΣΤΕΡΙΑ	ΓΑΛΑΞΙΕΣ	ΚΒΑΖΑΡΣ
	ΑΣΤΕΡΙΑ	8885	3292	10
	ΓΑΛΑΞΙΕΣ	4577	27450	1455
	ΚΒΑΖΑΡΣ	309	375	7682
		Προβλεπόμενη κλάση		

Πραγματική κλάση		ΑΣΤΕΡΙΑ	ΓΑΛΑΞΙΕΣ	ΚΒΑΖΑΡΣ
	ΑΣΤΕΡΙΑ	0.73	0.27	0
	ΓΑΛΑΞΙΕΣ	0.14	0.82	0.04
	ΚΒΑΖΑΡΣ	0.04	0.04	0.92
		Προβλεπόμενη κλάση		

Συνολικό σφάλμα $= \frac{3292+10+4477+1455+309+375}{54035} = \frac{10018}{54035} = \mathbf{0.19}$

Παρατηρούμε ότι με τον εναλλακτικό τρόπο ταξινόμησης το τελικό σφάλμα ταξινόμησης αυξάνεται από 16% σε 18-19%. Οπότε θεωρούμε ότι η ταξινόμηση που εκτελεί η βιβλιοθήκη *sklearn* έχει ελαφρώς καλύτερη απόδοση.

3.7 Σύγκριση αποτελεσμάτων μεθόδων LDA

Παρακάτω συνοψίζονται τα αποτελέσματα από τις μεθόδους που υλοποιήσαμε:

Σύγκριση Μεθόδων		
Μέθοδος	Σφάλμα ταξινόμησης (Σύνολο εκπαίδευσης)	Σφάλμα ταξινόμησης (Σύνολο αξιολόγησης)
Fisher [διαχωρισμός φασικού χώρου]	0.17	0.17
Fisher [όρια απόφασης sklearn]	0.29	0.29
LDA+SVD [διαχωρισμός φασικού χώρου]	0.18	0.19
LDA+SVD [όρια απόφασης sklearn]	0.16	0.16

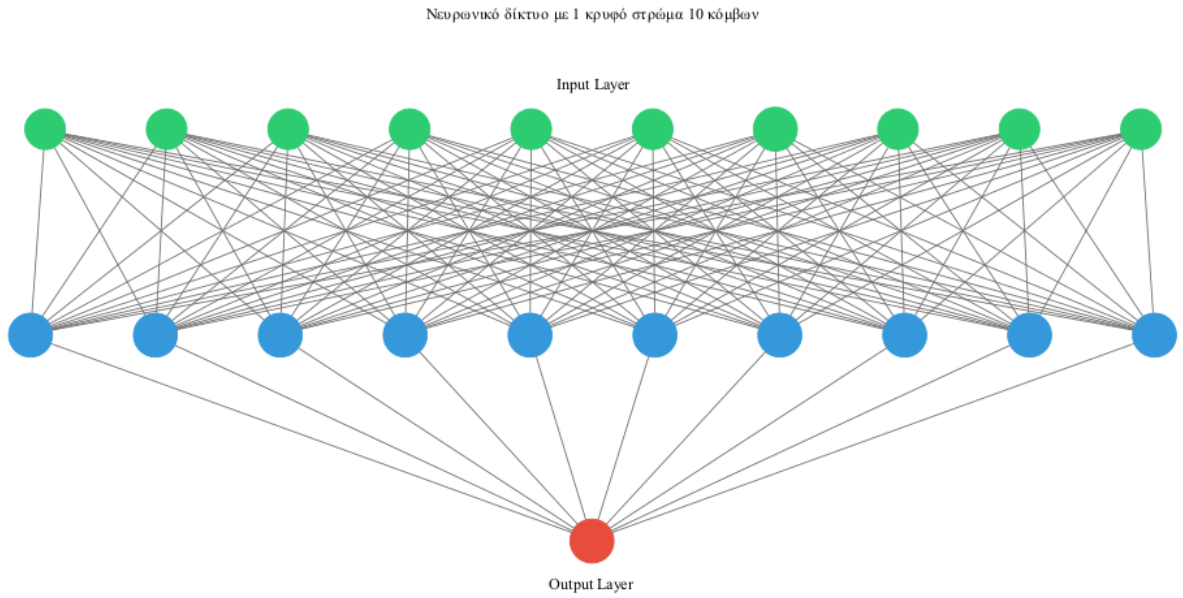
Η ταξινόμηση που εκτελεί η βιβλιοθήκη *sklearn* για την μέθοδο LDA+SVD έχει πολύ καλύτερη απόδοση από τη μέθοδο του Fisher, αφού η LDA+SVD έχει σφάλματα της τάξης του 18-19% ενώ ο Fisher έχει σφάλματα της τάξης του 29%. Αυτό ίσως οφείλεται στο γεγονός ότι η LDA+SVD έχει καλύτερη υπολογιστική ικανότητα, αφού η ανάλυση του πίνακα δεδομένων X είναι αριθμητικά πιο σταθερή από την επίλυση προβλήματος ιδιοτιμών – ιδιοδιανυσμάτων των πινάκων διασποράς που εκτελεί η Fisher. Επίσης, η μέθοδος LDA+SVD προτού ταξινομήσει τα δεδομένα, κάνει decorrelation των δεδομένων (αφού μετατρέπει τον πίνακα διασποράς Σ στον ταυτοτικό πίνακα I) οπότε αυτό συμβάλει στην καλύτερη απόδοση του ταξινομητή.

Κεφάλαιο 4

Νευρωνικά Δίκτυα

Τα *Feed-forward Νευρωνικά δίκτυα* ή *Multilayer Percpetron (MLP)* χρησιμοποιούνται ευρέως για την επίλυση προβλημάτων ταξινόμησης δεδομένων. Τα Νευρωνικά Δίκτυα είναι μια κατηγορία αλγορίθμων μηχανικής μάθησης υπό επίβλεψη (supervised learning) όπου κατά την εκπαίδευση κάθε διάνυσμα του δείγματος αντιπροσωπεύεται από μια τιμή η οποία είναι μοναδική για κάθε κλάση (class label).

Ένα Νευρωνικό δίκτυο αποτελείται από το στρώμα εισόδου (input layer), τα κρυφά στρώματα (hidden layers) και το στρώμα εξόδου (output layer). Κάθε ένα απ' αυτά αποτελείται από νευρώνες (ή κόμβους) οι οποίοι είναι μεταξύ τους συνδεδεμένοι:



Κάθε κόμβος του δικτύου υλοποιεί μια συνάρτηση ενεργοποίησης $f(.)$ (activation function) που έχει τη μορφή

$$y = f(\mathbf{w}^T\mathbf{x} + w_0)$$

όπου $\mathbf{w}$ είναι το διάνυσμα βαρών και w_0 το κατώφλι (bias), $\mathbf{x}$ είναι το διάνυσμα εισόδου (το οποίο αποτελεί κάποιο διάνυσμα του δείγματος αν βρισκόμαστε στο στρώμα εισόδου, αλλιώς αποτελεί το output του προηγούμενου στρώματος). Τα στρώματα του MLP είναι πλήρως συνδεδεμένα, οι παράμετροι κάθε κόμβου είναι ανεξάρτητοι από τους υπόλοιπους κόμβους του στρώματος, οπότε κάθε κόμβος επιστρέφει ένα μοναδικό σύνολο βαρών.

Προκειμένου να αξιολογείται η απόδοση του δικτύου, είναι απαραίτητο να ορίσουμε μια συνάρτηση κόστους. Αν η πρόβλεψη του δικτύου είναι λανθασμένη (δηλαδή το διάνυσμα ταξινομήθηκε σε λανθασμένη κλάση) τότε το κόστος είναι υψηλό, ενώ εάν η πρόβλεψη είναι ορθή τότε το κόστος είναι χαμηλό.

Για την εκπαίδευση του μοντέλου χρειαζόμαστε πέρα από τη συνάρτηση κόστους, και μια διαδικασία βελτιστοποίησης, η οποία υπολογίζει τα διανύσματα βαρών για τα οποία η τιμή της συνάρτηση κόστους μειώνεται.

4.1 Αλγόριθμος

Τα βήματα εκπαίδευσης ενός Νευρωνικού Δικτύου περιγράφονται συνοπτικά πιο κάτω:

Η ακόλουθη διαδικασία εκτελείται για όλα τα διανύσματα του συνόλου εκπαίδευσης:

1. Forward pass

   Εισάγεται το διάνυσμα στο μοντέλο, πολλαπλασιάζεται με το διάνυσμα βαρών και προστίθεται το κατώφλι. Το αποτέλεσμα περνάει από τη συνάρτηση ενεργοποίησης και στη συνέχεια προωθείται διαδοχικά στα επόμενα στρώματα, ακολουθώντας την ίδια διαδικασία. Τελικά το στρώμα εξόδου δίνει μια τελική πρόβλεψη.

2. Υπολογισμος σφαλματος

   Λαμβάνοντας υπόψη την πρόβλεψη του δικτύου, και την πραγματική τιμή της κλάσης στην οποία ανήκει το διάνυσμα, υπολογίζεται μέσω της συνάρτησης κόστους το σφάλμα ταξινόμησης. Σκοπός είναι να υπολογιστούν τα διανύσματα βαρών τα οποία ελαχιστοποιούν την τιμή της συνάρτησης κόστους. Στη συνέχεια, ο αλγόριθμος Backpropagation προωθεί προς τα πίσω το σφάλμα, ούτως ώστε να γίνει αναπροσαρμογή των βαρών.

3. Backward pass

   Με τη μέθοδο απότομης καθόδου (Gradient Descent Method) ενημερώνονται τα βάρη του τελευταίου στρώματος και κατόπιν ενημερώνονται τα βάρη των προηγούμενων στρωμάτων του δικτύου.

Η παραπάνω διαδικασία επαναλαμβάνεται για όσες επαναλήψεις (epochs) ορίσει ο χρήστης.

Στο τέλος της εκπαίδευσης του Νευρωνικού, τα βάρη που υπολογίστηκαν σταθεροποιούνται και ακολουθεί η εισαγωγή των διανυσμάτων του συνόλου αξιολόγησης στο δίκτυο. Για κάθε διάνυσμα εκτελείται η διαδικασία που περιγράφεται στο βήμα 1.

4.2 Υλοποίηση

Για την υλοποίηση του Νευρωνικού Δικτύου χρησιμοποιήσαμε το πακέτο της Python, $Tensorflow$. Το $Tensorflow$ παρέχει την δυνατότητα κατασκευής του αλγόριθμου του Νευρωνικού Δικτύου από το μηδέν, δίνοντάς στο χρήστη την ευκαιρία να επιλέξει ο ίδιος τις διάφορες συναρτήσεις ή μεθόδους για το μοντέλο χωρίς περιορισμούς. Έτσι, ο χρήστης μπορεί να πειραματιστεί με τον συνδυασμό διαφόρων συναρτήσεων και διαδικασιών ούτως ώστε να καταλήξει στο μοντέλο που προσαρμόζεται καλύτερα στο πρόβλημά του.
Οι αρχικοποιήσεις, οι συναρτήσεις και οι μέθοδοι που χρησιμοποιήσαμε περιγράφονται παρακάτω:

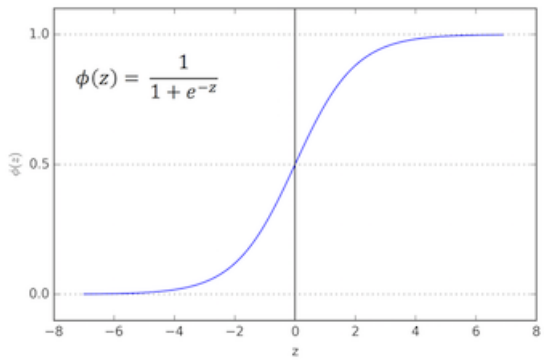
1. **Αρχικοποίηση διανυσμάτων βαρών:**

   Οι αρχικές τιμές των διανυσμάτων βαρών είναι τυχαία επιλεγμένες από την κολοβή Κανονική κατανομή (truncated Normal Distribution) με μέση τιμή $\mu = 0$ και τυπική απόκλιση $\sigma = 1$. Η συγκεκριμένη αρχικοποίηση προτείνεται για μικρά Νευρωνικά Δίκτυα, σαν αυτά που εκπαιδεύουμε.

2. **Συνάρτηση ενεργοποίησης:**

   Χρησιμοποιήσαμε ως συνάρτηση ενεργοποίησης τη **λογιστική** (ή **σιγμοειδή**) συνάρτηση για το κρυφό στρώμα και τη συνάρτηση **softmax** για το στρώμα εξόδου.

   Ως συνάρτηση ενεργοποίησης, συνήθως επιλέγουμε μια μη γραμμική συνάρτηση η οποία να είναι ευέλικτη στην περιγραφή αυθαίρετων σχέσεων. Μία καλή επιλογή είναι η λογιστική συνάρτηση, αφού είναι μη γραμμική, ομαλή και μονότονη με πεπερασμένο εύρος τιμών και η παράγωγός της είναι εύκολα υπολογίσιμη:



Σχήμα 4.1: Πηγή εικόνας: `https://djinit-ai.github.io/2018/02/08/neural-networks.html`

Η συνάρτηση softmax είναι μια γενικευμένη μορφή της λογιστικής συνάρτησης. Η softmax απεικονίζει τις τιμές εισόδου (ή logits) στο διάστημα $[0, 1]$, με τέτοιο τρόπο ώστε το άθροισμα όλων των τιμών εξόδου να είναι 1, οπότε τα αποτελέσματα που μας δίνει μπορούν να θεωρηθούν ως πιθανότητες:

$$S(y_i) = \frac{e^{y_i}}{\sum_j e^{y_j}}$$

3. **Συνάρτηση κόστους:**

   Ως συνάρτηση κόστους (Cost Function) χρησιμοποιήσαμε τη συνάρηση **softmax cross entropy with logits**. Υπολογίζει την μέγιστη εντροπία του αποτελέσματος, αφού εφαρμόσει αρχικά τη συνάρτηση softmax (δηλαδή πρώτα μετατρέπει τις τιμές εισόδου σε πιθανότητες και στη συνέχεια υπολογίζει την μέγιστη εντροπία):

$$-\sum_{k=1}^{K} \mathbf{y_{o,k}} ln(p(C_k|\mathbf{x}))$$

   όπου K ο συνολικός αριθμός των κλάσεων και $\mathbf{y_{o,k}}$ παίρνει την τιμή 1 ή 0 αναλόγως εάν το διάνυσμα ταξινομήθηκε στην ορθή κλάση ή όχι.

4. **Μέθοδος βελτιστοποίησης των βαρών:**

   Η επαγωγική διαδικασία που εφαρμόζεται προκειμένου να υπολογιστούν τα διανύσματα βαρών σε κάθε στρώμα του δικτύου είναι η μέθοδος **Batch Gradient Descent (BGD)**, η οποία είναι μια ειδική περίπτωση της μεθόδου Gradient Descent, η οποία ανανεώνει τα βάρη σύμφωνα με τη σχέση:

$$w_{t+1} = w_t - \rho \nabla E(w_t)$$

όπου ρ είναι ο ρυθμός εκπαίδευσης.

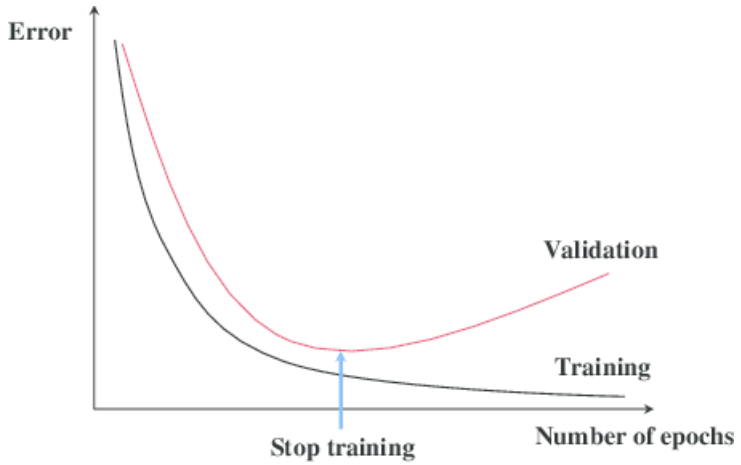
Η BGD υπολογίζει το σφάλμα για κάθε διάνυσμα του συνόλου εκπαίδευσης όμως ενημερώνει τα βάρη αφού υπολογιστεί το σφάλμα για όλα τα διανύσματα εκπαίδευσης **(bulk/batch training)**. Κάθε φορά που ολοκληρώνεται η διαδικασία υπολογισμού του σφάλματος για όλα τα διανύσματα εκπαίδευσης, ολοκληρώνεται ένας κύκλος εκπαίδευσης (training epoch).

4.2.1 Αρχιτεκτονική

Για την κατασκευή ενός ταξινομητή, καλούμαστε να επιλέξουμε τη σωστή αρχιτεκτονική που ταιριάζει στον αριθμό των δεδομένων αλλά και στον αριθμό των μεταβλητών που διαθέτουμε. Όσο πιο περίπλοκο είναι το μοντέλο ταξινόμησής μας, τόσο μεγαλύτερη είναι η πιθανότητα να έχουμε σφάλματα ταξινόμησης. Συγκεκριμένα, για το Νευρωνικό δίκτυο, το απλούστερο μοντέλο είναι αυτό που περιέχει το μικρότερο δυνατό αριθμό ελευθέρων παραμέτρων (δηλαδή βαρών και κατωφλιών) ή ισοδύναμα τον μικρότερο αριθμό νευρώνων. Για να κατασκευάσουμε κάποιο Νευρωνικό Δίκτυο που έχει ιδιότητα γενίκευσης, πρέπει να κατασκευάσουμε το απλούστερο δυνατό Νευρωνικό Δίκτυο για τα δεδομένα που διαθέτουμε.

4.2.2 Υπερεκπαίδευση

Κατά την εκπαίδευση του ταξινομητή, ερχόμαστε αντιμέτωποι με το φαινόμενο της υπερεκπαίδευσης (overtraining), δηλάδη το μοντέλο έχει υψηλή απόδοση στην ταξινόμηση των δεδομένων εκπαίδευσης, ενώ αδυνατεί να ταξινομήσει άγνωστα δεδομένα. Με άλλα λόγια, δεν έχει τη δυνατότητα γενίκευσης της εκπαίδευσης. Για να ελέγξουμε αν συμβαίνει αυτό, για κάθε κύκλο εκπαίδευσης σχεδιάζουμε την αντίστοιχη τιμή της συνάρτησης κόστους. Το κόστος (loss) του συνόλου εκπαίδευσης θα μειώνεται συνέχεια ενώ το κόστος του συνόλου ελέγχου θα εμφανίζει ελάχιστο. Το σημείο ελαχίστου σηματοδοτεί το σημείο στο οποίο εμφανίζεται υπερεκπαίδευση, δηλαδή το σημείο στο οποίο πρέπει να σταματήσει η εκπαίδευση.



Σχήμα 4.2: Πηγή εικόνας: http://fouryears.eu/tags/machine-learning/

4.3 Εκτέλεση και Αποτελέσματα

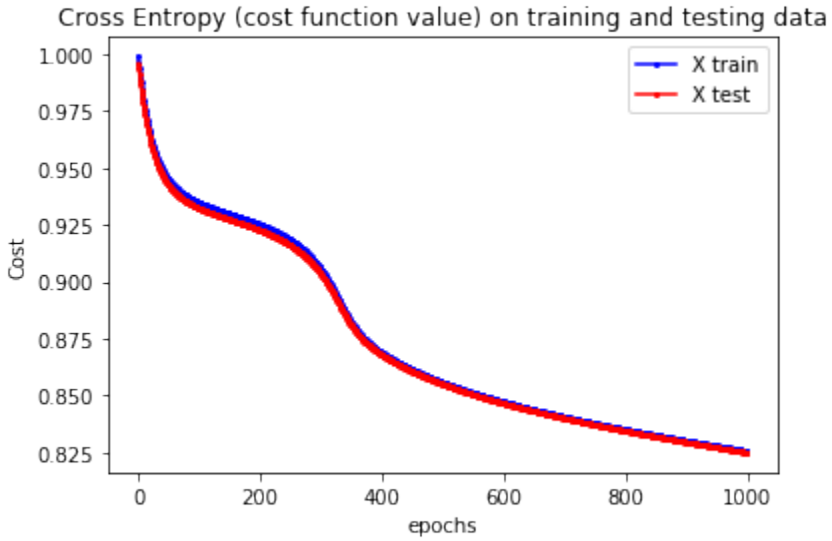
Όπως και προηγουμένως, έτσι και τώρα τυποποιούμε τα δεδομένα που διαθέτουμε, για το λόγο που προαναφέραμε. Παρακάτω αναλύουμε τις δοκιμές που υλοποιήσαμε προκειμένου να καταλήξουμε στο μοντέλο που προσαρμόζεται καλύτερα στο πρόβλημα.

4.3.1 Ένα Νευρωνικό Δίκτυο με ένα κρυφό στρώμα 6 κόμβων

Αρχικά υλοποιήσαμε ένα Νευρωνικό Δίκτυο με ένα κρυφό στρώμα το οποίο περιείχε 6 κόμβους. Επιλέξαμε μια απλή αρχιτεκτονική για το δίκτυο λόγω του μικρού αριθμού μεταβλητών που διαθέτουμε. Αφού εκπαιδεύσαμε το νευρωνικό, μελετήσαμε και αξιολογήσαμε τα αποτελέσματα της ταξινόμησης που εκτελεί, όπως παρουσιάζονται πιο κάτω.

Αποτελέσματα

Παρακάτω απεικονίζεται η καμπύλη της συνάρτησης κόστους:

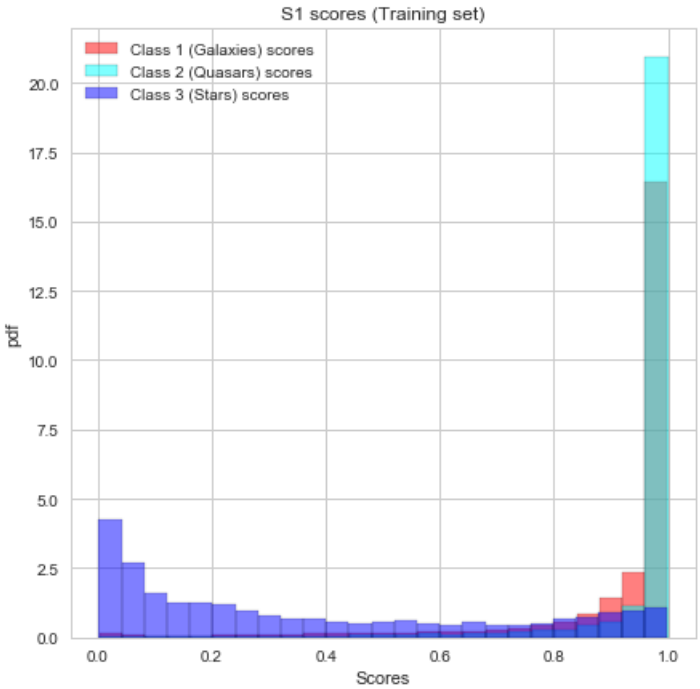


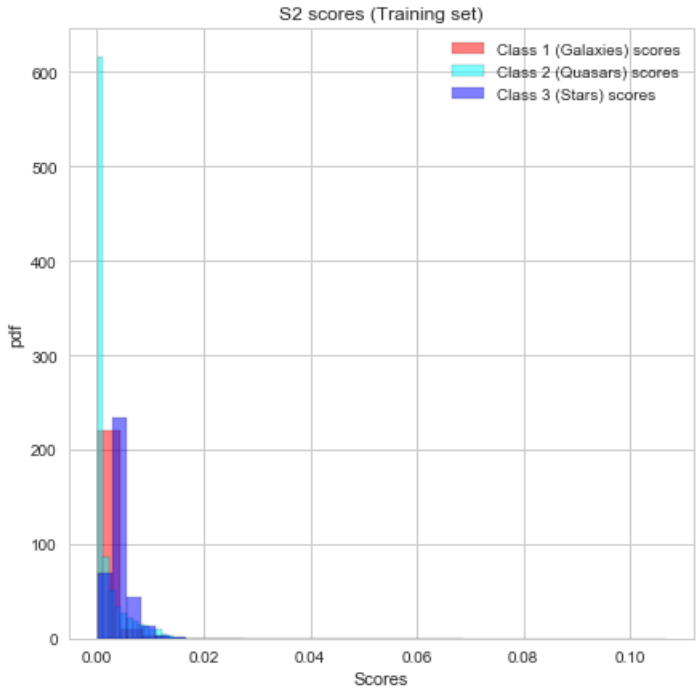
Από την καμπύλη της συνάρτησης κόστους παρατηρούμε ότι δεν εμφανίζεται υπερκεπαίδευση αφού το κόστος του συνόλου αξιολόγησης μειώνεται χωρίς να παρουσιάζει σημείο ελαχίστου.

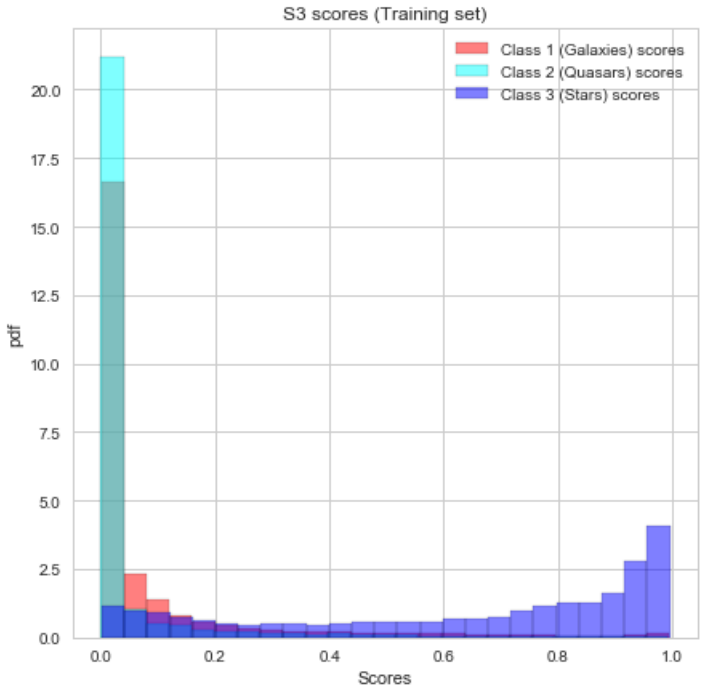
Αξιολόγηση

Στο τέλος της εκτέλεσης των δοκιμών, το νευρωνικό δίκτυο επιστρέφει για κάθε παρατήρηση 3 σκορς όπου το καθένα αντιστοιχεί στην πιθανότητα η παρατήρηση x_i να ανήκει στην κλάση $C_k, k = 1, 2, 3$.

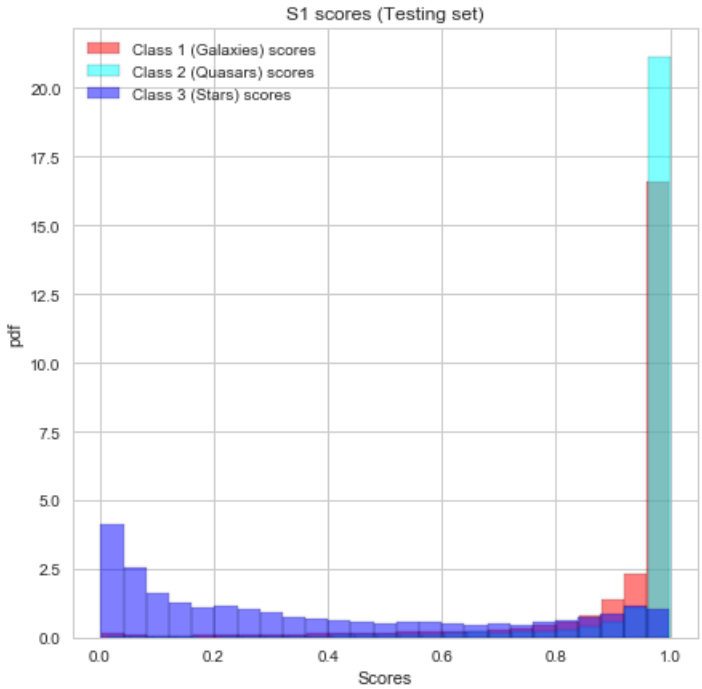
Παρακάτω απεικονίζονται τα ιστογράμματα κατανομής των σκορς που αποδόθηκαν στα διανύσματα του συνόλου εκπαίδευσης:

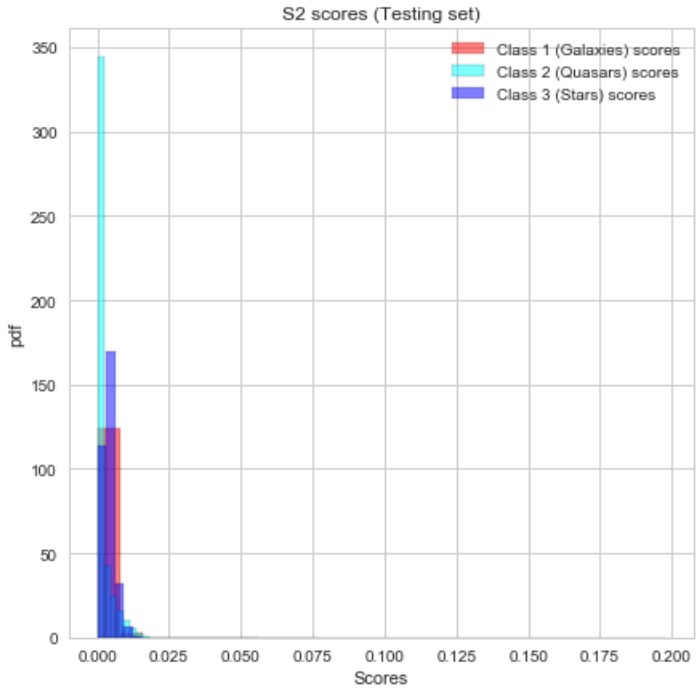

S1 scores (Training set)
Class 1 (Galaxies) scores
Class 2 (Quasars) scores
Class 3 (Stars) scores
pdf
Scores
0.0
2.5
5.0
7.5
10.0
12.5
15.0
17.5
20.0
0.0
0.2
0.4
0.6
0.8
1.0

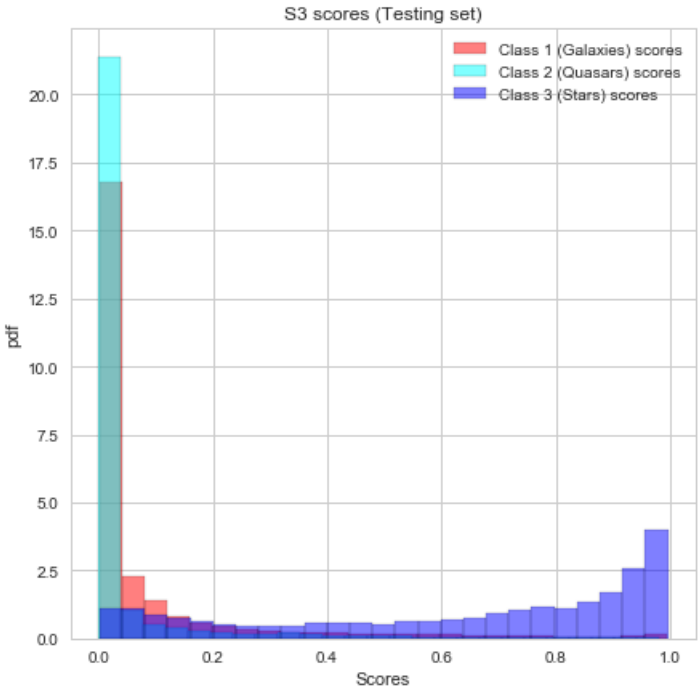

S2 scores (Training set)
Class 1 (Galaxies) scores
Class 2 (Quasars) scores
Class 3 (Stars) scores
pdf
Scores
0
100
200
300
400
500
600
0.00
0.02
0.04
0.06
0.08
0.10



Παρακάτω απεικονίζονται τα ιστογράμματα κατανομής των σκορς που αποδόθηκαν στα διανύσματα του συνόλου αξιολόγησης:



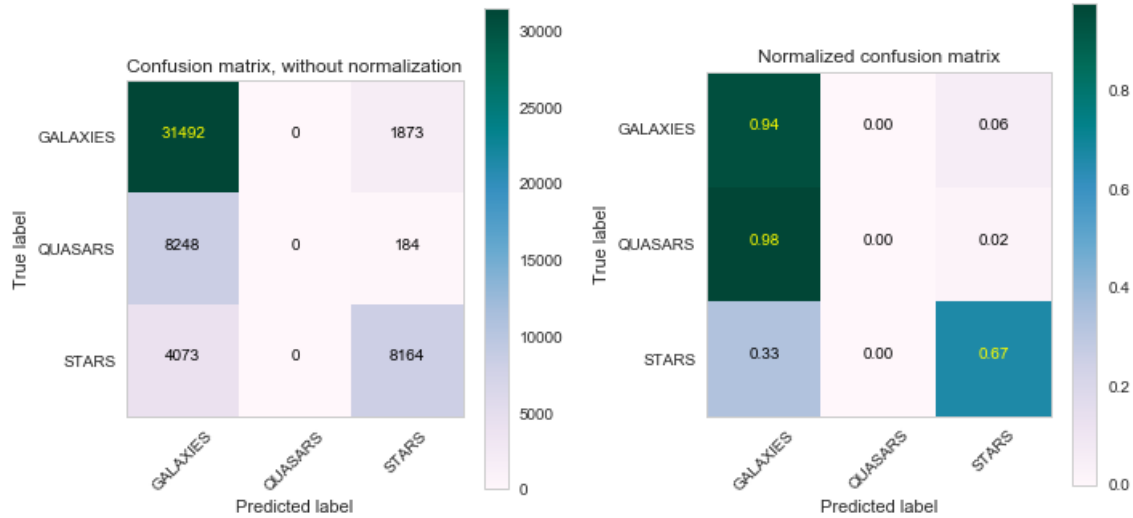




Από τις κατανομές των πιθανοτήτων παρατηρούμε ότι το μοντέλο έχει τη δυνατότητα να διαχωρίζει τις παρατηρήσεις των Αστέρων από τις παρατηρήσεις των άλλων δύο κλάσεων, ενώ αδυνατεί να διαχωρίσει τις παρατηρήσεις των Γαλαξιών με αυτές των Κβάζαρς.

Confusion Matrices

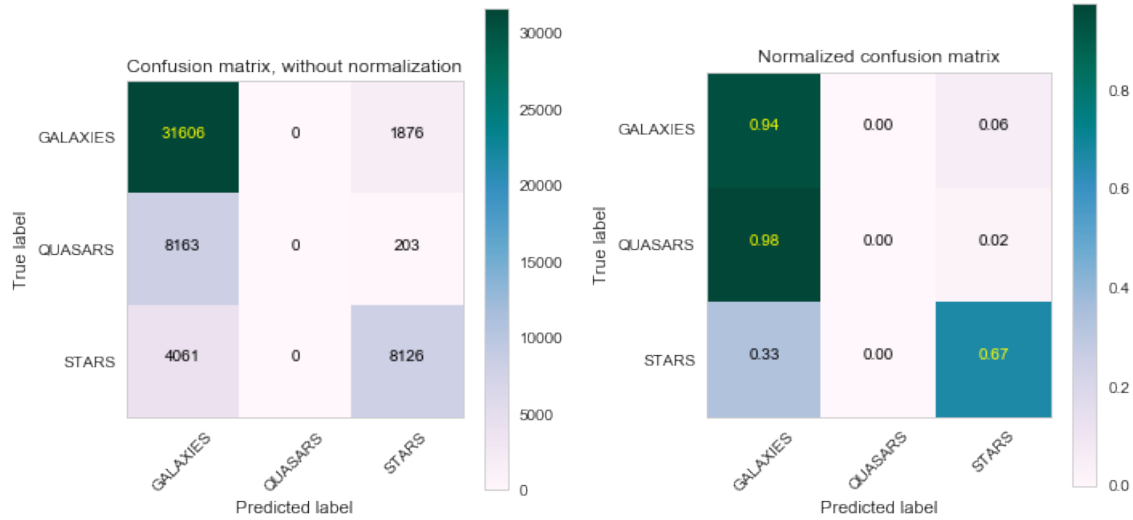
Σύνολο Εκπαίδευσης



Σφάλμα ταξινόμησης:

$$\frac{14372}{54034} = 0.27$$

Σύνολο Αξιολόγησης



Σφάλμα ταξινόμησης:

$$\frac{14303}{54035} = 0.26$$

Παρατηρούμε ότι κανένα από τα διανύσματα των Κβάζαρς δεν ταξινομήθηκε σωστά. Τα συνολικά σφάλματα ταξινόμησης είναι υψηλά αφού βρίσκονται στο 26% - 27%.

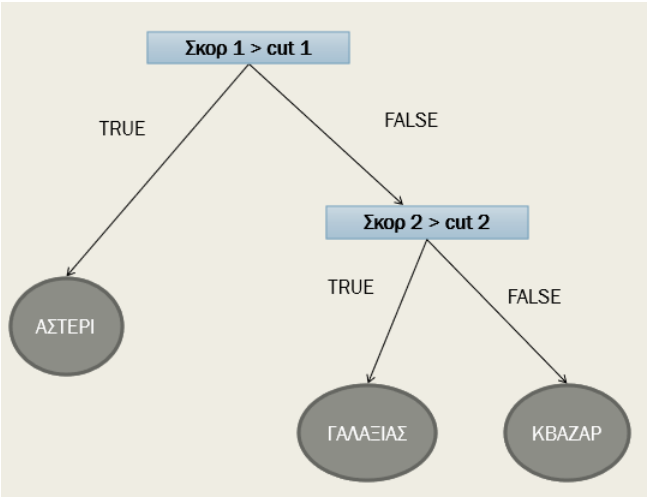
4.3.2 Δύο Νευρωνικά Δίκτυα με ένα κρυφό στρώμα 6 κόμβων το καθένα

Βασιζόμενοι στην παρατήρηση ότι οι Αστέρες διαχωρίζονται από τις άλλες δύο κλάσεις, ενώ το μοντέλο αδυνατεί να διαχωρίσει τους Γαλαξίες από τα Κβάζαρς, επιχειρήσαμε να δοκιμάσουμε τον συνδυασμό δύο Νευρωνικών Δικτύων. Υλοποιήσαμε έναν αλγόριθμο ο οποίος χρησιμοποιεί ένα σκορ από κάθε Νευρωνικό ούτως ώστε να ταξινομήσει τα δεδομένα.

Το πρώτο Νευρωνικό Δίκτυο εκπαιδεύεται χρησιμοποιώντας ολόκληρο το σύνολο εκπαίδευσης. Σκοπός του είναι να διαχωρίσει τις παρατηρήσεις των Αστέρων από τις παρατηρήσεις των άλλων δύο κλάσεων, γι' αυτό το λόγο θεωρούμε τους Αστέρες ως κλάση 0 και τις άλλες δύο κλάσεις τις συγχωνεύουμε στην κλάση 1. Στο τέλος της εκπαίδευσης, δηλαδή αφού υπολογιστούν τα τελικά βάρη του δικτύου, εκτελείται η διαδικασία αξιολόγησης όπου για κάθε διάνυσμα επιστρέφεται ένα σκορ, έστω *σκορ 1*.

Το δεύτερο Νευρωνικό Δίκτυο εκπαιδεύεται χρησιμοποιώντας μόνο τις παρατηρήσεις των Γαλαξιών και των Κβάζαρς, αφού σκοπός του είναι να διαχωρίζει τους Γαλαξίες (που θεωρούμε ως κλάση 0) από τα Κβάζαρς (κλάση 1). Στο τέλος της εκπαίδευσης, εκτελείται η διαδικασία αξιολόγησης (χρησιμοποιώντας ολόκληρο το σύνολο δεδομένων) όπου για κάθε διάνυσμα επιστρέφεται ένα σκορ, έστω *σκορ 2*.

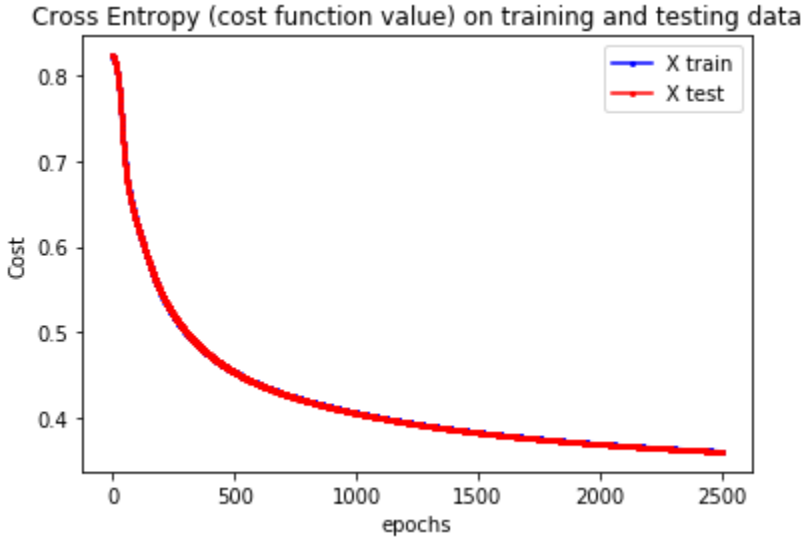
Επομένως, στο τέλος της εκπαίδευσης και των δύο δικτύων, διαθέτουμε δύο σκορς για κάθε παρατήρηση. Η τελική ταξινόμηση των παρατηρήσεων στις 3 κλάσεις εκτελείται ως εξής:



Δοκιμάσαμε διάφορους συνδυασμούς παραμέτρων, χρησιμοποιώντας τον συγκεκριμένο αλγόριθμο.

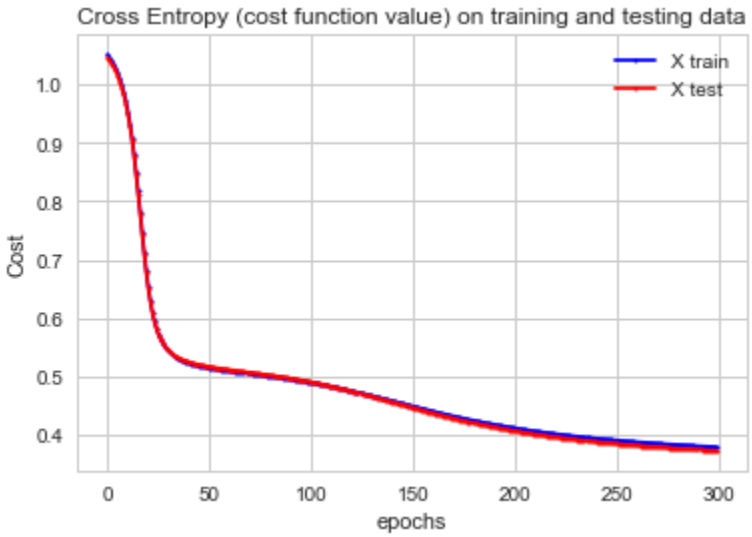
Τα δύο Νευρωνικά Δίκτυα έχουν την ίδια αρχιτεκτονική, δηλαδή αποτελούνται από 1 κρυφό στρώμα με 6 κόμβους. Το 1ο Νευρωνικό εκπαιδεύτηκε με ρυθμό εκπαίδευσης = 0.3 για 2500 κύκλους εκπαίδευσης. Παρακάτω παρουσιάζεται η καμπύλη της συνάρτησης κόστους που προέκειψε:

Αποτελέσματα:



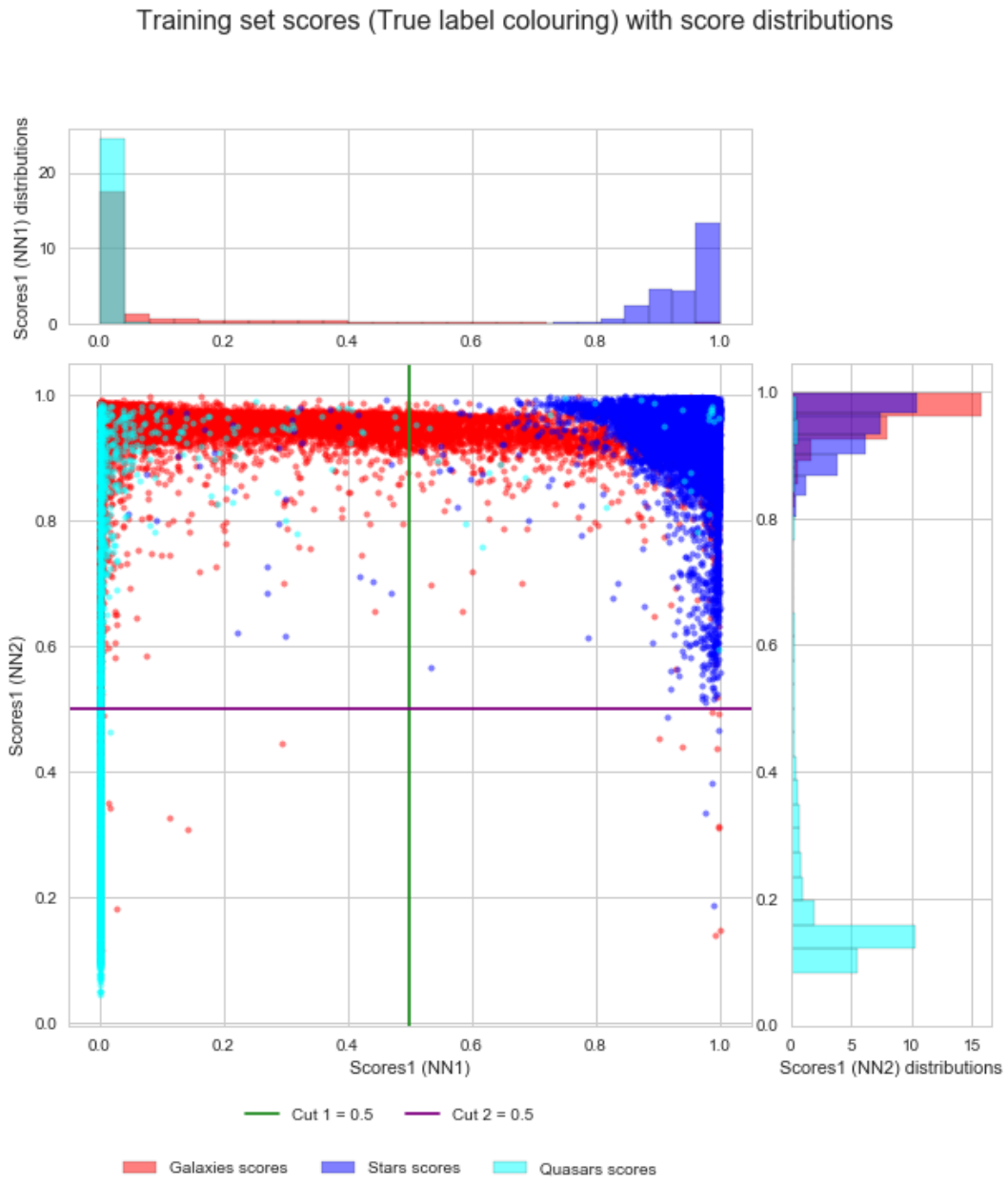
Από την καμπύλη της συνάρτησης κόστους παρατηρούμε ότι δεν εμφανίζεται υπερκεπαίδευση αφού το κόστος του συνόλου αξιολόγησης μειώνεται σταθερά.

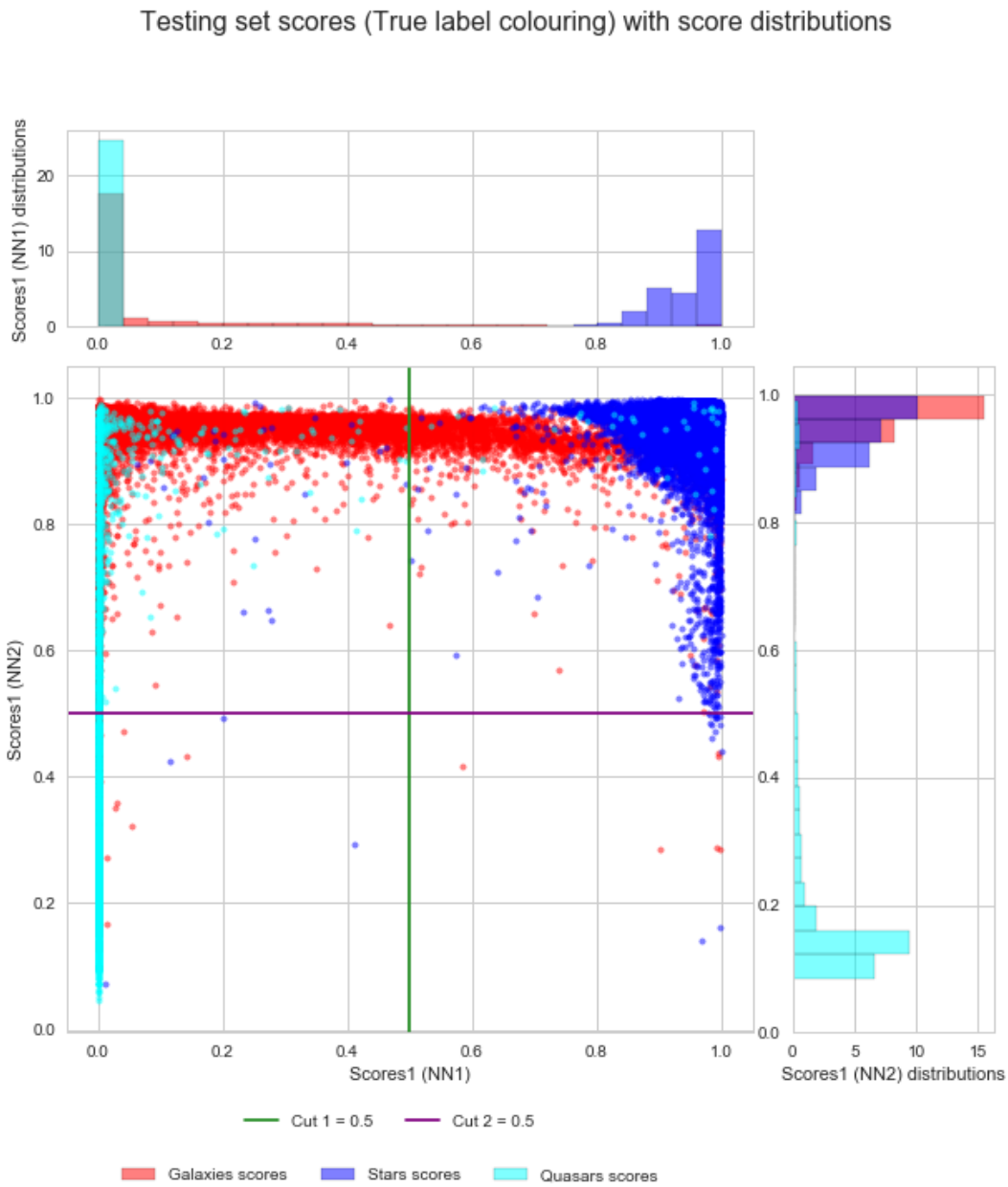
Το 2ο Νευρωνικό εκπαιδεύτηκε με ρυθμό εκπαίδευσης = 0.3 για 300 κύκλους εκπαίδευσης. Η καμπύλη της συνάρτησης κόστους παρουσιάζεται πιο κάτω:



Ούτε στο 2ο Νευρωνικό δίκτυο παρουσιάζεται υπερεκπαίδευση, οπότε μπορούμε να προχωρήσουμε με την αξιολόγηση του αλγορίθμου.

Για την ευκολότερη μελέτη των αποτελεσμάτων σχεδιάσαμε ένα διάγραμμα με σημεία που έχουν ως x-συντεταγμένη τα *σκορ 1* και ως y-συντεταγμένη τα *σκορ 2*. Σε κάθε άξονα προσαρμόστηκε το αντίστοιχο ιστόγραμμα για κάθε σκορ. Υπενθυμίζουμε ότι με τη βοήθεια του δέντρου απόφασης που υλοποιήσαμε, χωρίσαμε το διάγραμμα σε 3 περιοχές.





Παρατηρούμε ότι τα περισσότερα σκορς των Αστέρων μαζεύονται στην περιοχή που ορίσαμε για αυτά ($\sigma\kappa o\rho\ 1 > cut1$). Ωστόσο, τα σκορς των Γαλαξιών δεν περιορίζονται στην περιοχή που ορίσαμε ($\sigma\kappa o\rho\ 1 < cut1$ και $\sigma\kappa o\rho\ 2 > cut2$) αφού κάποιοι ταξινομήθηκαν στο χωρίο των Αστέρων. Τέλος, τα περισσότερα σκορς των Κβαζαρς ταξινομούνται στην περιοχή που ορίσαμε ($\sigma\kappa o\rho\ 1 < cut1$ και $\sigma\kappa o\rho\ 2 < cut2$). Για τον ακριβή αριθμό των ταξινομήσεων θα μελετήσουμε τα confusion matrices που ακολουθούν.

Αξιολόγηση

Με τη βοήθεια των 2 σκορς υπολογίζουμε βάσει των thresholds (cut 1, cut 2) τα confusion matrices, με την βοήθεια των οποίων θα υπολογίζουμε κάθε φορά το σφάλμα ταξινόμησης που θα προκύπτει για κάθε διαφορετικό συνδυασμό thresholds, με σκοπό να βρούμε τον συνδυασμό που δίνει το ελάχιστο σφάλμα ταξινόμησης.

Confusion Matrices

Παρακάτω παρουσιάζονται τα Confusion matrices και τα αντίστοιχα κανονικοποιημένα Confusion matrices των συνόλων εκπαίδευσης και αξιολόγησης. Τα διαγώνια στοιχεία εκπροσωπούν τα ορθώς ταξινομημένα διανύσματα, ενώ τα μη-διαγώνια στοιχεία τα διανύσματα που ταξινομήθηκαν εσφαλμένα. Ενδεικτικά, τα confusion matrices για $cut1 = cut2 = 0.5$ είναι τα ακόλουθα:

Σύνολο εκπαίδευσης

Πραγματική κλάση		ΑΣΤΕΡΙΑ	ΓΑΛΑΞΙΕΣ	ΚΒΑΖΑΡΣ
	ΑΣΤΕΡΙΑ	12207	30	0
	ΓΑΛΑΞΙΕΣ	2265	30695	405
	ΚΒΑΖΑΡΣ	31	1176	7225
		ΑΣΤΕΡΙΑ	ΓΑΛΑΞΙΕΣ	ΚΒΑΖΑΡΣ
		Προβλεπόμενη κλάση		

Πραγματική κλάση		ΑΣΤΕΡΙΑ	ΓΑΛΑΞΙΕΣ	ΚΒΑΖΑΡΣ
	ΑΣΤΕΡΙΑ	$\frac{12207}{12237} \simeq 1$	$\frac{30}{12237} \simeq 0$	$\frac{0}{12237} = 0$
	ΓΑΛΑΞΙΕΣ	$\frac{2265}{33365} = 0.07$	$\frac{30695}{33365} = 0.92$	$\frac{405}{33365} = 0.01$
	ΚΒΑΖΑΡΣ	$\frac{31}{8432} \simeq 0$	$\frac{1176}{8432} = 0.14$	$\frac{7225}{8432} = 0.86$
		ΑΣΤΕΡΙΑ	ΓΑΛΑΞΙΕΣ	ΚΒΑΖΑΡΣ
		Προβλεπόμενη κλάση		

Σφάλμα ταξινόμησης:

$$\frac{30 + 2265 + 405 + 31 + 1176}{54034} = \frac{3907}{54034} = 0.07$$

Σύνολο αξιολόγησης

Πραγματική κλάση				
	ΑΣΤΕΡΙΑ	12154	29	4
	ΓΑΛΑΞΙΕΣ	2244	30835	403
	ΚΒΑΖΑΡΣ	37	1134	7195
		ΑΣΤΕΡΙΑ	ΓΑΛΑΞΙΕΣ	ΚΒΑΖΑΡΣ
		Προβλεπόμενη κλάση		

|

Πραγματική κλάση				
	ΑΣΤΕΡΙΑ	$\frac{12154}{12237} = 0.99$	$\frac{29}{12237} \simeq 0$	$\frac{4}{12237} \simeq 0$
	ΓΑΛΑΞΙΕΣ	$\frac{2244}{33365} = 0.07$	$\frac{30835}{33365} = 0.92$	$\frac{403}{33365} = 0.01$
	ΚΒΑΖΑΡΣ	$\frac{37}{8432} \simeq 0$	$\frac{1134}{8432} = 0.13$	$\frac{7195}{8432} = 0.85$
		ΑΣΤΕΡΙΑ	ΓΑΛΑΞΙΕΣ	ΚΒΑΖΑΡΣ
		Προβλεπόμενη κλάση		

Σφάλμα ταξινόμησης:

$$\frac{29 + 4 + 2244 + 403 + 37 + 1134}{54035} \frac{3851}{54035} = 0.07$$

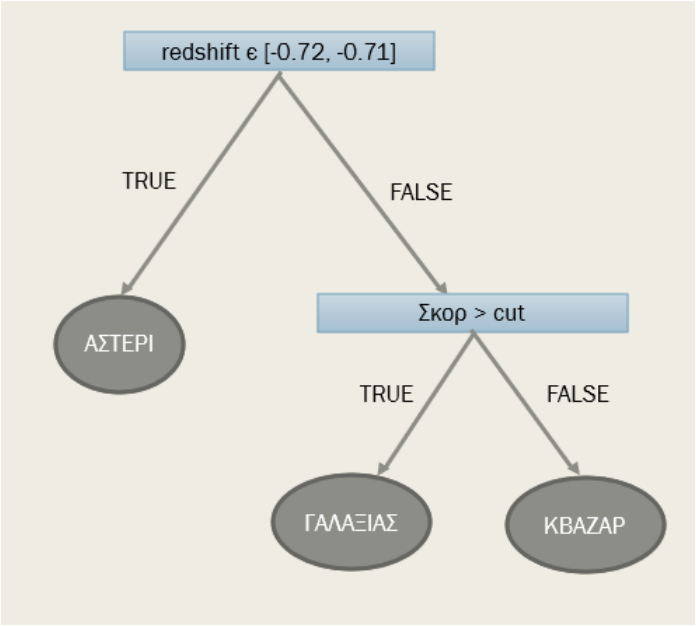
Στον παρακάτω πίνακα συνοψίζονται οι συνδυασμοί των thresholds που παρουσίασαν τα χαμηλότερα σφάλματα ταξινόμησης:

CUT OPTIMIZATION			
Cut 1	Cut 2	Συνολικό σφάλμα ταξινόμησης	
		Σύνολο Εκπαίδευσης	Σύνολο Αξιολόγησης
0.5	0.5	0.07	0.07
0.7	0.6	0.05	0.05
0.7	0.7	0.05	0.05

4.3.3 Συνδυασμός ενός Νευρωνικού δικτύου με ένα κρυφό στρώμα 6 κόμβων και συνθήκης που βασίζεται στην τιμή του redshift

Στη συνέχεια, στην προσπάθειά μας να βελτιώσουμε την απόδοση του ταξινομητή, παρατηρήσαμε από το ιστόγραμμα της μεταβλητής redshift ότι οι τιμές της που αντιστοιχούν στις παρατηρήσεις των Αστέρων παίρνουν τιμές στο διάστημα $[-0.72, -0.71]$ (βλ. υποκεφάλαιο 1.6).

Βασιζόμενοι στην παρατήρηση αυτή επιχειρήσαμε να αντικαταστήσουμε την 1η συνθήκη που αφορά στα σκορ 1 που μας έδινε το 1ο Νευρωνικό δίκτυο, με μια συνθήκη που αφορά στην μεταβλητή redshift. Συγκεκριμένα, ο αλγόριθμος διαμορφώνεται ως εξής:



Θεωρούμε $cut = 0.5$

Αξιολόγηση

Confusion Matrices

Σύνολο Εκπαίδευσης

Πραγματική κλάση		ΑΣΤΕΡΕΣ	ΓΑΛΑΞΙΕΣ	ΚΒΑΖΑΡΣ
	ΑΣΤΕΡΕΣ	12174	63	0
	ΓΑΛΑΞΙΕΣ	216	32642	507
	ΚΒΑΖΑΡΣ	1	1137	7294
		ΑΣΤΕΡΕΣ	ΓΑΛΑΞΙΕΣ	ΚΒΑΖΑΡΣ
		Προβλεπόμενη κλάση		

Πραγματική κλάση		ΑΣΤΕΡΕΣ	ΓΑΛΑΞΙΕΣ	ΚΒΑΖΑΡΣ
	ΑΣΤΕΡΕΣ	$\frac{12174}{12237} = 0.99$	$\frac{63}{12237} = 0.01$	$\frac{0}{12237} = 0$
	ΓΑΛΑΞΙΕΣ	$\frac{216}{33365} = 0.01$	$\frac{32642}{33365} = 0.98$	$\frac{507}{33365} = 0.02$
	ΚΒΑΖΑΡΣ	$\frac{1}{8432} \simeq 0$	$\frac{1137}{8432} = 0.13$	$\frac{7294}{8432} = 0.86$
		ΑΣΤΕΡΕΣ	ΓΑΛΑΞΙΕΣ	ΚΒΑΖΑΡΣ
		Προβλεπόμενη κλάση		

Σφάλμα ταξινόμησης:

$$\frac{63 + 216 + 507 + 1 + 1137}{54034} = \frac{1924}{54034} = 0.036$$

Σύνολο Αξιολόγησης

Πραγματική κλάση		ΑΣΤΕΡΕΣ	ΓΑΛΑΞΙΕΣ	ΚΒΑΖΑΡΣ
	ΑΣΤΕΡΕΣ	12109	77	1
	ΓΑΛΑΞΙΕΣ	199	32768	515
	ΚΒΑΖΑΡΣ	3	1107	7256
		Προβλεπόμενη κλάση		

Πραγματική κλάση		ΑΣΤΕΡΕΣ	ΓΑΛΑΞΙΕΣ	ΚΒΑΖΑΡΣ
	ΑΣΤΕΡΕΣ	$\frac{12109}{12187} = 0.99$	$\frac{77}{12187} = 0.01$	$\frac{1}{12187} \simeq 0$
	ΓΑΛΑΞΙΕΣ	$\frac{199}{33482} = 0.01$	$\frac{32768}{33482} = 0.98$	$\frac{515}{33482} = 0.02$
	ΚΒΑΖΑΡΣ	$\frac{3}{8366} \simeq 0$	$\frac{1107}{8366} = 0.13$	$\frac{7256}{8366} = 0.87$
		Προβλεπόμενη κλάση		

Σφάλμα ταξινόμησης:

$$\frac{77 + 1 + 199 + 515 + 3 + 1107}{54035} = \frac{1902}{54035} = 0.035$$

Τα συνολικά σφάλματα ταξινόμησης είναι της τάξης του 3.5%-3.6%, οπότε το τελικό αποτέλεσμα είναι ικανοποιητικό. Η αλλαγή στην 1η συνθήκη που κάναμε έχει ελαφρώς καλύτερα αποτελέσματα από την πρώτη δοκιμή αφού το σφάλμα ταξινόμησης μειώθηκε από 5% σε 3.5% - 3.6%.

4.4 Σύγκριση αποτελεσμάτων αλγορίθμων

Τα συνολικά σφάλματα ταξινόμησης συνοψίζονται στον παρακάτω πίνακα:

Σύγκριση Αλγορίθμων		
Αλγόριθμος	Σφάλμα ταξινόμησης (Σύνολο εκπαίδευσης)	Σφάλμα ταξινόμησης (Σύνολο αξιολόγησης)
Ένα Νευρωνικό Δίκτυο με 1 κρυφό στρώμα 6 κόμβων	0.27	0.26
Συνδυασμός δύο Νευρωνικών Δικτύων με 1 κρυφό στρώμα 6 κόμβων το καθένα	0.05	0.05
Συνδυασμός ενός Νευρωνικού Δικτύου με 1 κρυφό στρώμα 6 κόμβων και συνθήκης redshift	0.036≃0.04	0.035≃0.04

Παρατηρούμε ότι η απόδοση του ενός Νευρωνικού Δικτύου σε σύγκριση με τους αλγορίθμους που συνδυάζουν δύο Νευρωνικά Δίκτυα, ή ένα Νευρωνικό Δίκτυο και τη συνθήκη του redshift δεν συγκρίνεται, αφού το πρώτο Νευρωνικό ταξινόμησε το 26%-27% των παρατηρήσεων σε λανθασμένη κλάση, ενώ οι αλγόριθμοι που αναπτύξαμε στη συνέχεια, σημείωσαν σφάλματα της τάξης του 4%-5%. Ωστόσο, σε όλες τις περιπτώσεις, η κλάση που είχε τα περισσότερα σφάλματα ταξινόμησης ήταν αυτή των Κβάζαρς.

Κεφάλαιο 5

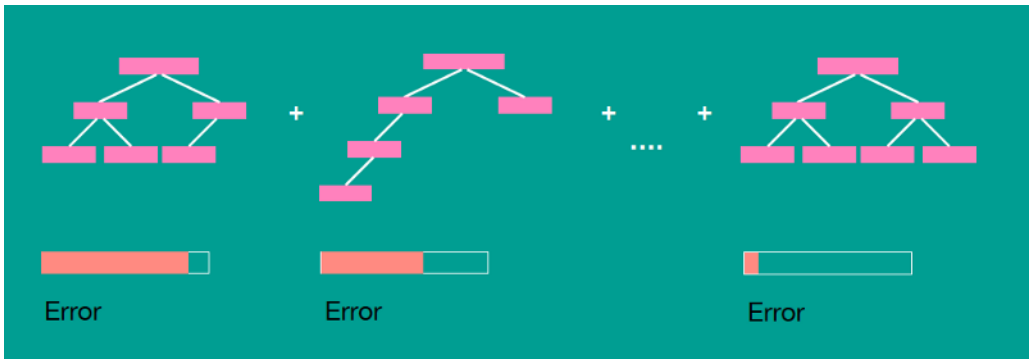
Ενδυναμωμένα Δέντρα Απόφασης

Έχει αποδειχθεί ότι εκπαιδεύοντας ένα συνδυασμό από μοντέλα, επιτυγχάνεται υψηλότερη απόδοση ταξινόμησης, συγκριτικά με αυτή που θα είχε ένα μοντέλο μόνο. Μια κατηγορία τέτοιων μεθόδων είναι γνωστή ως Ενδυνάμωση **(Boosting)** και αφορά στην διαδοχική εκπαίδευση πολλών μοντέλων, όπου η τιμή της συνάρτησης κόστους που χρησιμοποιείται για να εκπαιδευτεί ένα συγκεκριμένο μοντέλο, εξαρτάται από την απόδοση που είχαν τα προηγούμενα. Με αυτό τον τρόπο επιτυγχάνεται ουσιαστική βελτίωση στην τελική απόδοση του ταξινομητή, συγκριτικά με την απόδοση που θα είχε ένα μόνο εκπαιδευόμενο μοντέλο. Υπάρχουν διάφορα είδη αλγορίθμων ενδυνάμωσης: κάποιοι επιστρέφουν τις τελικές προβλέψεις ως το βεβαρυμένο άθροισμα των προβλέψεων των επιμέρους μοντέλων, ενώ άλλοι επιστρέφουν σαν τελική πρόβλεψη αυτή που έκανε ένα συγκεκριμένο μοντέλο, η επιλογή του οποίου γίνεται συναρτήσει των μεταβλητών που εισάγουμε.

Τα επιμέρους μοντέλα που χρησιμοποιούνται συνήθως σε τέτοιου είδους αλγορίθμους είναι τα **δέντρα απόφασης**. Τα δέντρα απόφασης διαιρούν το χώρο των χαρακτηριστικών σε άνισα ορθογώνια με πλευρές παράλληλες προς τους άξονες, οπότε οι τελικές περιοχές απόφασης που προκύπτουν είναι υπερορθογώνια. Η διαδικασία για την επιλογή της τελικής περιοχής απόφασης (δηλαδή για την κατάταξη των παρατηρήσεων σε κάποια κλάση) περιλαμβάνει μια ακολουθία διαδοχικών διαχωρισμών του αρχικού συνόλου σε δύο υποσύνολα μέχρι οι παρατηρήσεις να καταλήξουν στα φύλλα του δέντρου.

Υπάρχουν διάφοροι αλγόριθμοι Ενδυναμωμένων Δέντρων Απόφασης. Η μέθοδος που χρησιμοποιήθηκε για την εργασία είναι γνωστή ως **Gradient Boosting Machine (GBM)**. Η Gradient Boost έχει σκοπό να ελαχιστοποιήσει το κόστος του τελικού ταξινομητή, προσθέτοντας σε κάθε βήμα έναν *ασθενή ταξινομητή (δηλ. ένα δέντρο)*, κρατώντας τους προηγούμενους αμετάβλητους, χρησιμοποιώντας τη μέθοδο απότομης καθόδου (Gradient Descent). Συγκεκριμένα, κτίζει δέντρα συγκεκριμένου βάθους βάσει του κόστους που είχαν τα προηγούμενα δέντρα. Η GBM δίνει το ίδιο βάρος (γνωστό ως *ρυθμός εκπαίδευσης*) στα αποτελέσματα όλων των δέντρων και συνεχίζει να κατασκευάζει δέντρα μέχρι τον επιθυμητό αριθμό που θέτει ο χρήστης. Οι τελικές προβλέψεις του αλγορίθμου αποτελούν το βεβαρυμένο άθροισμα των απο-

τελεσμάτων που είχαν τα επιμέρους μοντέλα. Ακολουθεί μια γενική σχηματική αναπαράσταση του αλγορίθμου:



Πλεονεκτήματα:

- Οι αλγόριθμοι ενδυνάμωσης έχουν υψηλή απόδοση σε προβλήματα ταξινόμησης και έχουν πιο σταθερή συμπεριφορά συγκριτικά με τη χρήση ενός μόνο μοντέλου, το οποίο επηρεάζεται από τις στατιστικές διακυμάνσεις του συνόλου εκπαίδευσης.
- Η Gradient Boost συνήθως είναι ασυναγώνιστη στην ακρίβεια πρόβλεψης, συγκριτικά με τους υπόλοιπους αλγορίθμους Μηχανικής Μάθησης. Είναι πολύ αποδοτική σε σύνολα δεδομένων μέτριας κλίμακας, δηλαδή της τάξης μερικών εκατομμυρίων διανυσμάτων.

Μειονεκτήματα:

- Τα GBMs επιχειρούν να ελαχιστοποιήσουν όλα τα σφάλματα που προκύπτουν από το σύνολο εκπαίδευσης. Αυτό έχει ως αποτέλεσμα να δίνεται μεγάλη έμφαση στα ακραία σημεία (outliers), το οποίο έχει ως επακόλουθο το φαινόμενο της υπερεκπαίδευσης (overfitting).
- Τα GBMs απαιτούν μεγάλη υπολογιστική ισχύ λόγω του ότι πολλά προβλήματα απαιτούν την κατασκευή πολλών δέντρων (100 και πάνω) για γίνει αποδοτική η ταξινόμηση, οπότε δαπανείται πολύς υπολογιστικός χρόνος και μνήμη.

5.1 Αλγόριθμος

Παρά να υλοποιηθεί ένας αλγόριθμος που θα εκτελεί multiclass ταξινόμηση (δηλαδή ταξινόμηση σε περισσότερες από 2 κλάσεις), υλοποιήθηκαν δύο αλγόριθμοι που εκτελούν ταξινόμηση σε 2 κλάσεις έκαστος, ακολουθώντας την ίδια λογική με αυτή που εφαρμόστηκε στα

Νευρωνικά Δίκτυα. Με αυτό τον τρόπο, η απόδοση του ταξινομητή βελτιώνεται.

ΑΛΓΟΡΙΘΜΟΣ ΤΗΣ GRADIENT BOOST ΓΙΑ ΤΑΞΙΝΟΜΗΣΗ ΣΕ ΔΥΟ ΚΛΑΣΕΙΣ

Υποθέτουμε ότι έχουμε στη διάθεσή μας n παρατηρήσεις και ο αλγόριθμος θα κατασκευάσει M δέντρα.

Εισαγωγη δεδομενων:

Εισάγονται οι παρατηρήσεις (x_i) και οι αντίστοιχες πραγματικές ετικέτες (true labels) (y_i) και μία διαφορίσιμη συνάρτηση κόστους $L(y_i, F(x))$. Η συνάρτηση κόστους που χρησιμοποιείται συνήθως για ταξινόμηση σε δύο κλάσεις είναι η *λογιστική συνάρτηση*:

$$log(loss) \quad = \quad -\sum_{i=1}^{n} (\, y_i log(p(y_i)) + (1 - y_i) log(1 - p(y_i))\,)$$

Κάθε προσθεταίος όρος μπορεί επίσης να εκφραστεί ως:

$$\begin{aligned} -(ylogp + (1-y)log(1-p)) &= -ylogp - (1-y)log(1-p) \\ &= ... \\ &= -ylog(odds) + log(1 + e^{log(odds)})\ (*) \end{aligned}$$

όπου $odds = \frac{p}{1-p}$ και $log(odds) = log(\frac{p}{1-p}) = \gamma$

Παραγωγίζοντας τη συνάρτηση κόστους προκύπτει:

$$\frac{dL}{d(log(odds))} = \frac{dL}{d(\gamma)} \quad = \quad -y + \frac{e^{log(odds)}}{1 + e^{log(odds)}}$$

Αφού

$$\frac{e^{log(odds)}}{1 + e^{log(odds)}} = p$$

τότε:

$$\frac{dL}{d(log(odds))} = \frac{dL}{d(\gamma)} \quad = \quad -\mathbf{y} + \mathbf{p}$$

Βηκα 1:

Το μοντέλο αρχικοποιείται με μια σταθερή τιμή $F_0(x)$. Το $F_0(x)$ είναι η τιμή $log(odds) = \gamma$ η οποία ελαχιστοποιεί την συνάρτηση κόστους, δηλαδή:

$$F_0(x) = \underset{\gamma}{\operatorname{argmin}} \sum_{i=1}^{n} L(y_i, \gamma)$$

Το $F_0(x)$ είναι η τιμή του πρώτου δέντρου (το οποίο αποτελείται από ένα μόνο φύλλο) και αποτελεί την αρχική πρόβλεψη που επιστρέφει ο αλγόριθμος.

Βηκα 2:

Για $m = 1$ μέχρι M:

1. Στο m-οστό δέντρο η κατεύθυνση της μεθόδου απότομης καθόδου δίνεται από την αρνητική παράγωγο της συνάρτησης κόστους. Προχωρώντας προς αυτή την κατεύθυνση το κόστος μειώνεται. Οπότε για κάθε παρατήρηση $i = 1, ..., n$ υπολογίζονται τα υπόλοιπα r_{i_m}:

$$r_{i_m} = -\left[\frac{\partial L(y_i, F(x_i))}{\partial F(x_i)}\right]_{F(x_i)=F_{m-1}(x_i)} = y_i - p(y_i)$$

2. Κατασκευάζεται ένα δέντρο απόφασης με τέτοιο τρόπο ούτως ώστε κάθε υπόλοιπο r_{i_m} να αντιστοιχεί σε κάποια περιοχή απόφασης R_{j_m} για $j = 1, ..., J_m$.

3. Για κάθε φύλλο $j = 1, ..., J_m$ υπολογίζεται το μήκος βήματος γ_{j_m} της μεθόδου απότομης καθόδου:

$$\begin{aligned} \gamma_{j_m} &= \underset{\gamma}{\operatorname{argmin}} \sum_{x_i \in R_{j_m}} L(y_i, F_{m-1}(x_i) + \gamma) \\ L(y_i, F_{m-1}(x_i) + \gamma) &\overset{(*)}{=} -y_i[F_{m-1}(x_i) + \gamma] + log(1 + e^{F_{m-1}(x_i)+\gamma}) \end{aligned}$$

Λόγω του ότι ειναι δύσκολο να υπολογιστεί η παράγωγος της συνάρτησης κόστους ως προς γ_{j_m} και στη συνέχεια να γίνει επίλυση ως προς γ_{j_m}, είναι προτιμότερο να προσεγγιστεί η συνάρτηση κόστους χρησιμοποιώντας πολυώνυμο Taylor 2ης τάξης. Για κάθε παρατήρηση x_i με πραγματική τιμή y_i προκύπτει:

$$\begin{aligned}
L(y_i, F_{m-1}(x_i) + \gamma) &\approx L(y_i, F_{m-1}(x_i)) + \frac{dL(y_i, F_{m-1}(x_i))}{dF()}.\gamma + \frac{1}{2}\frac{d^2L(y_i, F_{m-1}(x_i))}{dF()^2}.\gamma^2 \\
\frac{dL(y_i, F_{m-1}(x_i) + \gamma)}{d\gamma} &\approx \frac{dL(y_i, F_{m-1}(x_i))}{dF()} + \frac{d^2L(y_i, F_{m-1}(x_i))}{dF()^2}.\gamma \\
&= 0
\end{aligned}$$

Λύνοντας ως προς γ_{j_m}:

$$\begin{aligned}
\frac{d^2L(y_i, F_{m-1}(x_i)}{dF()^2}.\gamma_{j_m} &= -\frac{dL(y_i, F_{m-1}(x_i))}{dF()} \\
\gamma_{j_m} &= \frac{-\frac{dL(y_i, F_{m-1}(x_i))}{dF()}}{\frac{d^2L(y_i, F_{m-1}(x_i))}{dF()^2}} \\
\gamma_{j_m} &= \frac{(y_i - p(y_i))}{(p(y_i).(1 - p(y_i)))}
\end{aligned}$$

Οπότε αθροίζοντας για όλες τις παρατηρήσεις x_i που ταξινομήθηκαν στην περιοχή R_{j_m} προκύπτει:

$$\gamma_{j_m} = \frac{\sum_{x_i \in R_{j_m}} (y_i - p(y_i))}{\sum_{x_i \in R_{j_m}} (p(y_i).(1 - p(y_i)))}$$

Τα γ_{j_m} υπολογίζονται για κάθε φύλλο του m-οστού δέντρου.

4. Για κάθε παρατήρηση x_i οι αντίστοιχες προβλέψεις $F(x_i)$ ενημερώνονται με τη βοήθεια της παρακάτω σχέσης:

$$F_m(x_i) = F_{m-1}(x_i) + \nu \sum_{j=1}^{J_m} \gamma_{j_m}.I(x_i \in R_{j_m})$$

όπου το ν είναι ο ρυθμός εκπαίδευσης.

Το βηκα 2 επαναλαμβάνεται μέχρι να ολοκληρωθεί ο αριθμός των δέντρων (M) που έχει τεθεί.

Βηκα 3:

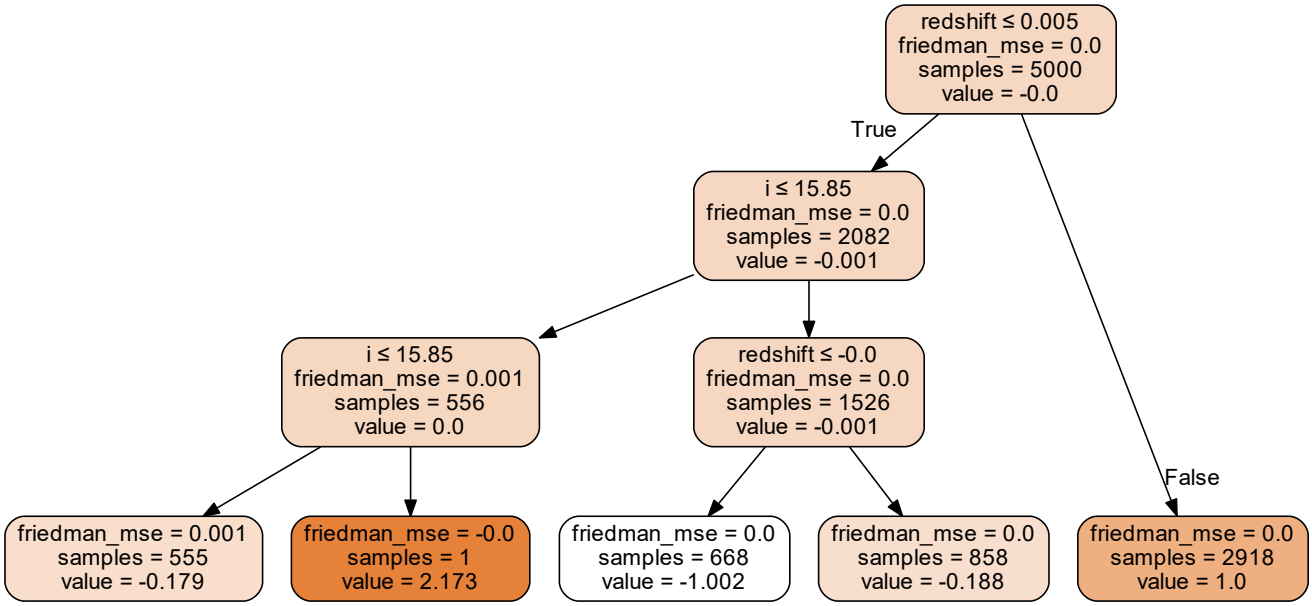
Ο αλγόριθμος επιστρέφει τις τελικές τιμές $F_M(x_i)$ για κάθε παρατήρηση x_i, οι οποίες αποτελούν τις τελικές προβλέψεις της Gradient Boost.

$$F_M(x_i) = log(odds) = log\left(\frac{p(y_i)}{1 - p(y_i)}\right)$$

Λύνοντας ως προς $p(y_i)$ υπολογίζεται η πιθανότητα η παρατήρηση x_i να ανήκει στην θετική κλάση:

$$p(y_i) = \frac{e^{log(odds)}}{1 + e^{log(odds)}} \ (**)$$

Παρακάτω απεικονίζεται ένα παράδειγμα ασθενούς ταξινομητή με βάθος 3:



Η τιμή value στα φύλλα του δέντρου αποτελεί την τιμή F_M (output value), που με τη βοήθεια του τύπου (**) μετατρέπεται στην πιθανότητα η παρατήρηση x_i να ανήκει στην θετική κλάση.

5.2 Υλοποίηση

Το μοντέλο Gradient Boosting Machine υλοποιήθηκε με τη βοήθεια της βιβλιοθήκης *scikit-learn* (ή *sklearn)* που διαθέτει η Python. Ένα μεγάλο πλεονέκτημα της μεθόδου GBM είναι η ευελιξία της αφού διαθέτει αρκετές δυνατότητες όσο αφορά στην επιλογή παραμέτρων, οπότε δίνεται η ευκαιρία στον χρήστη να πειραματιστεί με διαφορετικούς συνδυασμούς προκειμένου να βρεθούν αυτοί που δίνουν τα μικρότερα σφάλματα ταξινόμησης. Λόγω του ότι η μέθοδος είναι ευάλωτη στην υπερεκπαίδευση, οι παράμετροι πρέπει να επιλεχθούν με προσοχή, ούτως ώστε να αποφευχθούν τέτοια φαινόμενα. Πιο συγκεκριμένα, μπορεί να χρησιμοποιηθεί κάθε διαφορίσιμη συνάρτηση κόστους (για παράδειγμα συνηθίζεται για προβλήματα παλινδρόμησης να χρησιμοποιείται η συνάρτηση ελαχίστων τετραγώνων ενώ για προβλήματα ταξινόμησης η λογιστική συνάρτηση).

Ωστόσο, επειδή οι αλληλεπιδράσεις που έχουν οι παράμετροι μεταξύ τους είναι περίπλοκες, είναι δύσκολο να εντοπιστούν οι βέλτιστες τιμές των παραμέτρων βάσει μόνο της θεωρίας. Επομένως, πρέπει να δοκιμαστούν διάφοροι διαφορετικοί συνδυασμοί παραμέτρων για να εντοπιστούν οι βέλτιστοι συνδυασμοί, δηλαδή αυτοί που δίνουν τα χαμηλότερα σφάλματα τόσο στο σύνολο εκπαίδευσης, όσο και στο σύνολο αξιολόγησης.

Επιλογές παραμέτρων που διαθέτει η βιβλιοθήκη *sklearn*:

1. **Συνάρτηση κόστους**
   Όπως αναφέρθηκε προηγουμένως, η συνάρτηση κόστους που επιλέχθηκε για το μοντέλο είναι η λογιστική συνάρτηση, η οποία χρησιμοποιείται συνήθως για προβλήματα ταξινόμησης δύο κλάσεων.

2. **Βάθος δέντρων**
   Δοκιμάστηκαν διαφορετικά βάθη ούτως ώστε να εντοπιστεί το καλύτερο μοντέλο που να ταιριάζει καλύτερα στο πρόβλημα. Όσο μεγαλύτερο βάθος έχουν τα δέντρα, τόσο πιο περίπλοκες τάσεις (patterns) μπορούν να περιγράψουν, όμως είναι πιο ευάλωτα στην υπερεκπαίδευση. Ωστόσο, εάν χρησιμοποιούνται δέντρα μικρού βάθους (έστω 2 με 5) τότε το πρόβλημα της υπερεκπαίδευσης μπορεί να αποφευχθεί.

3. **Ρυθμός εκπαίδευσης**
   Η *ανθεκτικότητα* (robustness) του μοντέλου μπορεί να ενισχυθεί μειώνοντας τον ρυθμό εκπαίδευσης, ο οποίος ελέγχει το βάρος που αποδίδεται στο κάθε δέντρο. Θέτοντας ένα

μικρό βάρος στο αποτέλεσμα που προκύπτει από κάθε δέντρο, έχει ως αποτέλεσμα σε κάθε επανάληψη του αλγορίθμου (δηλαδή με τη δημιουργία ενός νέου δέντρου) η μέθοδος απότομης καθόδου να προχωράει ένα βήμα προς την σωστή κατεύθυνση (δηλαδή προς την κατεύθυνση που η συνάρτηση κόστους μειώνεται με τον μεγαλύτερο ρυθμό). Έχει αποδειχθεί ότι εκτελώντας πολλά μικρά βήματα προς την ορθή κατεύθυνση, προκύπτουν μικρότερα σφάλματα ταξινόμησης στο σύνολο αξιολόγησης. Θέτοντας έναν μικρό ρυθμό εκπαίδευσης (0.1-0.3) μειώνεται η επιρροή που έχει κάθε δέντρο στην τελική πρόβλεψη, και αυτό βελτιώνει την απόδοση του αλγορίθμου μετά από αρκετές επαναλήψεις. Ορίζοντας μικρό ρυθμό εκπαίδευσης απαιτείται η κατασκευή περισσότερων δέντρων, όμως αυτό βελτιώνει την ακρίβεια των προβλέψεων. Είναι σημαντικό να βρεθεί ο βέλτιστος συνδυασμός ρυθμού εκπαίδευσης και αριθμού των δέντρων. Για αυτό το λόγο, δοκιμάστηκαν διάφοροι συνδυασμοί τιμών αυτών των δύο παραμέτρων. Τελικά, ορίστηκε ο ρυθμός εκπαίδευσης να είναι 0.1 και αναζητούσαμε τον βέλτιστο αριθμό δέντρων.

4. **Αριθμός δέντρων**
   Εάν οριστεί μεγάλος αριθμός δέντρων τότε υπάρχει πιθανότητα να παρουσιαστεί υπερεκπαίδευση. Για να ελέγχεται εάν παρουσιάζεται αυτό το φαινόμενο, σχεδιάζεται η καμπύλη των τιμών της συνάρτησης κόστους για το σύνολο εκπαίδευσης και για το σύνολο αξιολόγησης (κάθε σημείο της καμπύλης αντιστοιχεί στο m-οστό δέντρο), προκειμένου να ελέγχουμε σε κάθε στάδιο του μοντέλου εάν παρουσιάζεται υπερεκπαίδευση, και σε ποια επανάληψη παρουσιάζεται.

5. **Κριτήριο διαχωρισμού**
   Στο βήμα 2.2 του αλγόριθμου, κατασκευάζεται ένα νέο δέντρο με τέτοιο τρόπο ούτως ώστε κάθε υπόλοιπο r_{i_m} να αντιστοιχεί σε κάποια περιοχή απόφασης R_{j_m}. Το δέντρο κτίζεται με τη βοήθεια του κριτήριου διαχωρισμού.

   Κάθε κόμβος του δέντρου περιέχει μια συνθήκη, βάσει της οποίας το αρχικό σύνολο χωρίζεται σε δύο υποσύνολα, κάθε φορά. Κάθε συνθήκη πρέπει να αυξάνει την ομοιογένεια των υποσυνόλων που προκύπτουν (συγκριτικά με την ομοιογένεια που είχαν τα προγενέστερα σύνολα). Αυτό σημαίνει ότι τα διανύσματα του συνόλου εκπαίδευσης σε κάθε ένα από τα νέα υποσύνολα δείχνουν υψηλότερη προτίμηση για συγκεκριμένη κλάση.
   Το κριτήριο διαχωρισμού που χρησιμοποιείται για να αξιολογηθούν οι πιθανοί διαχωρισμοί της περιοχής απόφασης R σε δύο υποπεριοχές (R_{left}, R_{right}) είναι γνωστό ως μέσο τετραγωνικό σφάλμα του Friedman (Friedman's Mean Squared Error), δηλαδή το μέσο τετραγωνικό σφάλμα με τη διόρθωση Friedman:

$$i^2(R_l, R_r) = \frac{w_l.w_r}{w_l + w_r}.(\overline{y}_l - \overline{y}_r)^2$$

   όπου $\overline{y}_l, \overline{y}_r$ είναι η μέση τιμή του αποτελέσματος που έδωσε ο αριστερός και δεξιός απόγο-

νος κόμβος αντίστοιχα, ενώ w_l, w_r είναι τα αντίστοιχα αθροίσματα μοναδιαίων βαρών. Τα μοναδιαία βάρη ευνοούν τους διαχωρισμούς υποσυνόλων από τους οποίους προκύπτουν συμμετρικά υποσύνολα σε κάθε απόγονο κόμβο.

5.3 Εκτέλεση και Αποτελέσματα

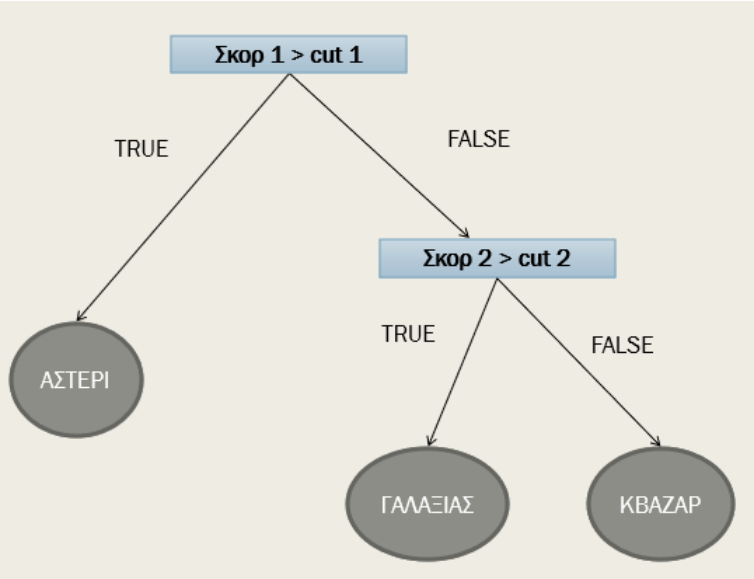
Οι μέθοδοι που χρησιμοποιούν δέντρα απόφασης τείνουν να αποδίδουν καλύτερα σε μη-επεξεργασμένα δεδομένα, οπότε για την εκτέλεση του αλγορίθμου Gradient Boosting Machine δεν τυποποιήθηκαν τα δεδομένα, όπως χρειάστηκε στα προηγούμενα μοντέλα Μηχανικής Μάθησης.

Για την εκτέλεση των Ενδυναμωμένων Δέντρων Απόφασης εφαρμόστηκε η ίδια λογική όπως αυτή που εφαρμόστηκε στα Νευρωνικά Δίκτυα. Κατασκευάστηκε ένα μοντέλο Gradient Boost για να διαχωρίσει τους Αστέρες (κλάση 0) από τους Γαλαξίες και τα Κβάζαρς (και τα δύο θεωρούνται ότι ανήκουν στην κλάση 1) και ένα ξεχωριστό δεύτερο μοντέλο Gradient Boost για να διαχωρίζει τους Γαλαξίες (κλάση 0) από τα Κβάζαρς (κλάση 1).

Το πρώτο μοντέλο εκπαιδεύτηκε με όλες τις παρατηρήσεις. Στο τέλος της εκπαίδευσης, το μοντέλο έδινε προβλέψεις τόσο για το σύνολο εκπαίδευσης όσο και για το σύνολο αξιολόγησης. Για κάθε παρατήρηση δινόταν η πιθανότητα να ανήκει στην κλάση των Αστέρων, η οποία ονομάστηκε **σκορ 1**.

Το δεύτερο μοντέλο εκπαιδεύτηκε με τις παρατηρήσεις των Γαλαξιών και των Κβαζαρς. Στο τέλος της εκπαίδευσης το μοντέλο έδινε προβλέψεις για το σύνολο εκπαίδευσης και για το σύνολο αξιολόγησης. Για κάθε παρατήρηση δινόταν η πιθανότητα να ανήκει στην κλάση των Γαλαξιών, η οποία ονομάστηκε **σκορ 2**.

Παρακάτω απεικονίζεται το δέντρο απόφασης που υλοποιεί ο αλγόριθμος, προκειμένου να ταξινομήσει τις παρατηρήσεις με τη βοήθεια και των δύο σκορς:



Θέτουμε $cut1 = cut2 = 0.5$.

Μετά από αρκετές δοκιμές αποδεικνύεται ότι είναι καλύτερο να μείνει σταθερός ο ρυθμός εκπαίδευσης στο 0.1 και να δοκιμαστούν διάφοροι συνδυασμοί των υπόλοιπων παραμέτρων. Κάποιες από τις δοκιμές παρουσιάζονται παρακάτω:

1ος ταξινομητής Gradient Boost					
Δοκιμή	**Αριθμός δέντρων**	**Ρυθμός εκπαίδευσης**	**Μέγιστο βάθος**	**Κριτήριο διαχωρισμού**	**Υπερεκπαίδευση**
1	100	0.1	5	Friedman's MSE	Στο 60° δέντρο
2	60	0.1	5	Friedman's MSE	----
3	100	0.1	4	Friedman's MSE	-----
4	100	0.1	3	Friedman's MSE	----
5	100	0.1	2	Friedman's MSE	----
6	150	0.1	2	Friedman's MSE	-----

2ος ταξινομητής Gradient Boost					
Δοκιμή	**Αριθμός δέντρων**	**Ρυθμός εκπαίδευσης**	**Μέγιστο βάθος**	**Κριτήριο διαχωρισμού**	**Υπερεκπαίδευση**
1	100	0.1	5	Friedman's MSE	Στο 51° δέντρο
2	51	0.1	5	Friedman's MSE	----
3	100	0.1	4	Friedman's MSE	Στο 75° δέντρο
4	100	0.1	3	Friedman's MSE	----
5	100	0.1	2	Friedman's MSE	----
6	150	0.1	2	Friedman's MSE	-----

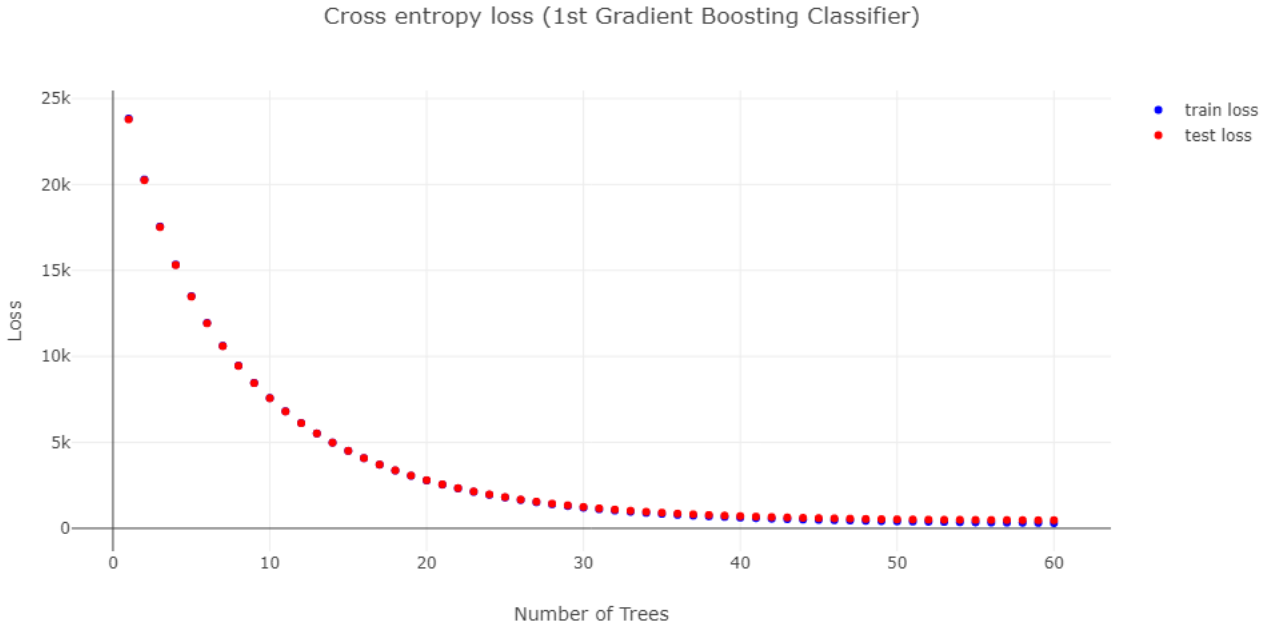
Δοκιμή	**Cut 1**	**Cut 2**	**Συνολικό σφάλμα ταξινόμησης**	
			Σύνολο εκπαίδευσης	**Σύνολο αξιολόγησης**
2	0.5	0.5	$\frac{974}{54034} = 0.018$	$\frac{950}{54035} = 0.018$
4	0.5	0.5	$\frac{1050}{54034} = 0.019$	$\frac{1016}{54035} = 0.019$
5	0.5	0.5	$\frac{1107}{54034} = 0.020$	$\frac{1090}{54035} = 0.020$
6	0.5	0.5	$\frac{1065}{54034} = 0.020$	$\frac{1069}{54035} = 0.020$

Το συνολικό σφάλμα ταξινόμησης ισούται με το πηλίκο του αριθμού των διανυσμάτων που ταξινομήθηκαν σε λανθασμένη κλάση προς τον συνολικό αριθμό παρατηρήσεων. Τα σφάλματα ταξινόμησης τόσο στο σύνολο εκπαίδευσης όσο και στο σύνολο αξιολόγησης είναι πολύ χαμηλά, αφού κυμαίνονται μεταξύ 1.8% - 2%, οπότε η συνολική απόδοση του αλγορίθμου της Gradient Boost είναι πολύ καλή.

Παρακάτω παρουσιάζονται με λεπτομέρεια τα αποτελέσματα της *2ης δοκιμής*:

Εκπαιδευση του 1ου ταξινομητη

Πιο κάτω φαίνονται οι καμπύλες της συνάρτησης κόστους για τα σύνολα εκπαίδευσης και αξιολόγησης:



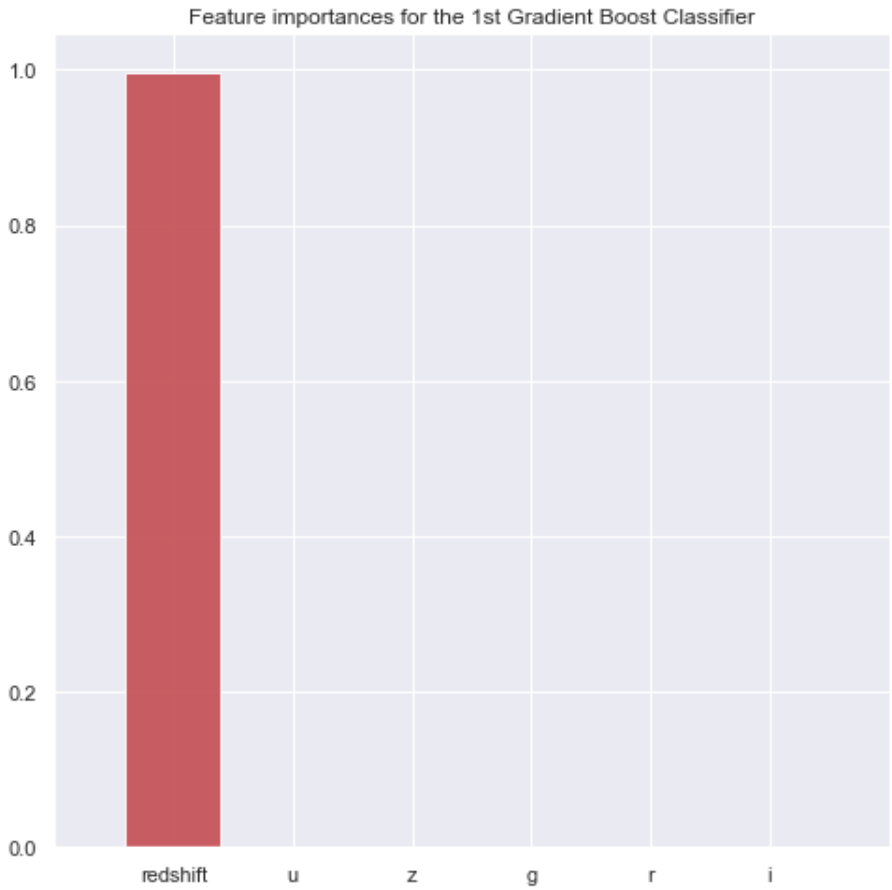
Κάθε σημείο αναπαριστά το κόστος ενός ασθενούς ταξινομητή. Το κόστος του συνόλου αξιολόγησης μειώνεται σταθερά, οπότε δεν υπάρχει κάποια ένδειξη για υπερεκπαίδευση.

Το ακόλουθο ιστόγραμμα αναπαριστά τη σημαντικότητα των χαρακτηριστικών, δηλαδή υποδηλώνει πόσο συνέβαλε η κάθε μεταβλητή στην εκπαίδευση του μοντέλου. Η σημαντικότητα

των μεταβλητών μετρήθηκε βάσει του πόσες φορές χρησιμοποιήθηκε η εκάστοτε μεταβλητή στις συνθήκες των κόμβων των δέντρων, βεβαρυμένη με την διόρθωση Friedman και διαιρεμένη με τον συνολικό αριθμό δέντρων.

Κατάταξη των χαρακτηριστικών:

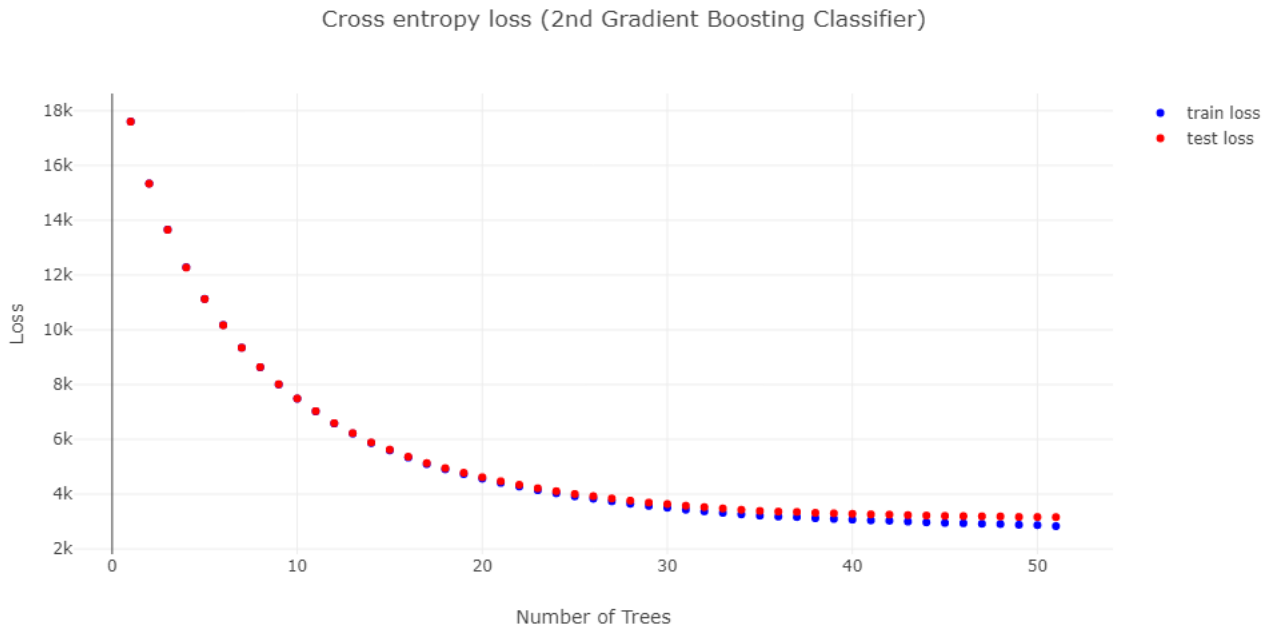
1. *Μετατόπιση στο ερυθρό (redshift)* (0.995948)
2. u (0.001419)
3. z (0.001094)
4. g (0.000584)
5. r (0.000541)
6. i (0.000415)



Είναι ξεκάθαρο ότι η μεταβλητή $redshift$ είναι η πιο σημαντική από όλες. Αυτό είναι αναμενόμενο αφού στο διάγραμμα της κατανομής του $redshift$ (βλ. υποκεφάλαιο 1.6) παρατηρείται ότι οι Αστέρες παίρνουν τιμές $redshift$ στο διάστημα $[-0.715, -0.710]$, οπότε μπορούν εύκολα να διαχωριστούν από τις άλλες δύο κλάσεις, βάσει της συγκεκριμένης μεταβλητής.

Εκπαιδευση του 2ου ταξινομητη:

Παρακάτω απεικονίζονται οι καμπύλες κόστους των συνόλων εκπαίδευσης και αξιολόγησης:



Κάθε σημείο αναπαριστά το κόστος ενός ασθενούς ταξινομητή. Το κόστος του συνόλου αξιολόγησης μειώνεται σταθερά, οπότε δεν υπάρχει κάποια ένδειξη για υπερεκπαίδευση.

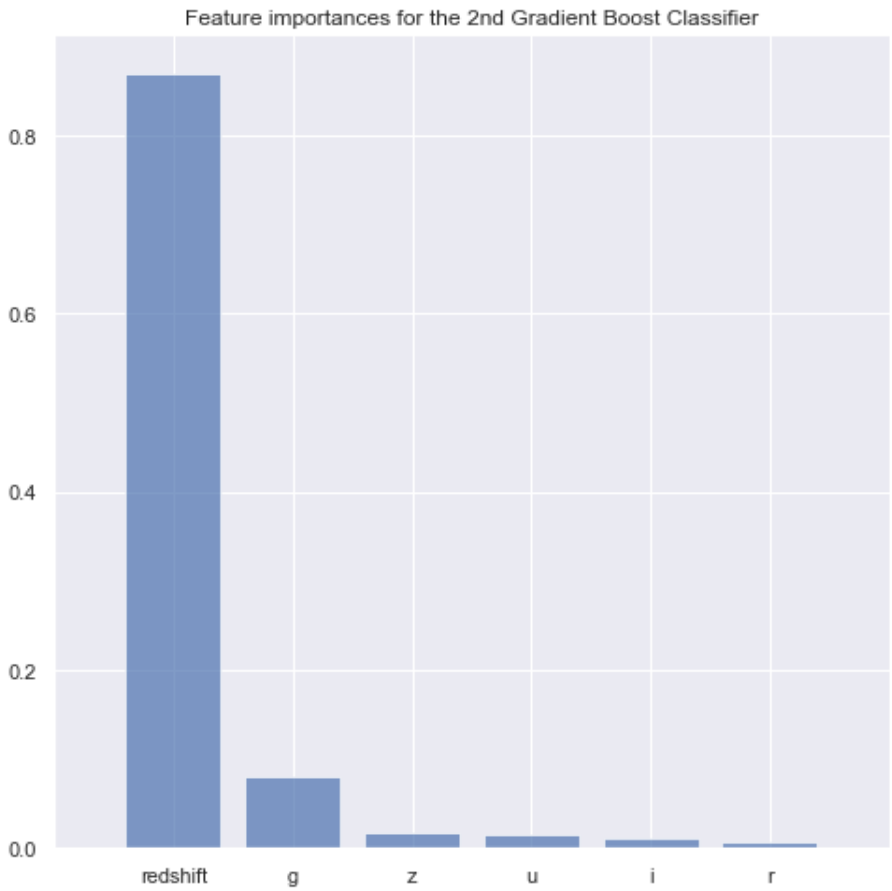
Το ακόλουθο ιστόγραμμα αναπαριστά τη σημαντικότητα των χαρακτηριστικών:

Κατάταξη των χαρακτηριστικών:

1. *Μετατόπιση στο ερυθρό (redshift)* (0.868653)
2. g (0.079312)
3. z (0.017803)
4. u (0.016343)

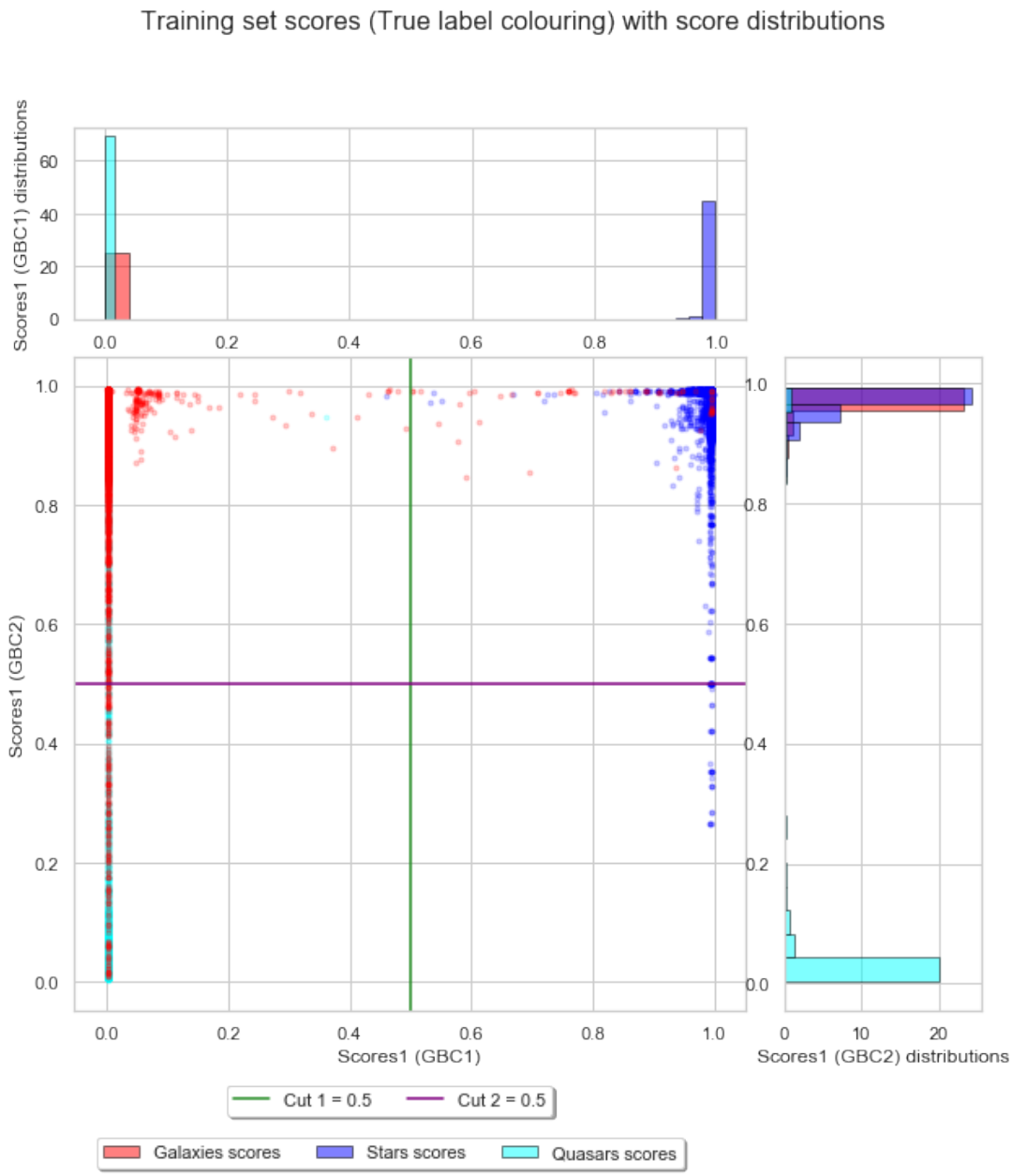
5. i (0.012010)

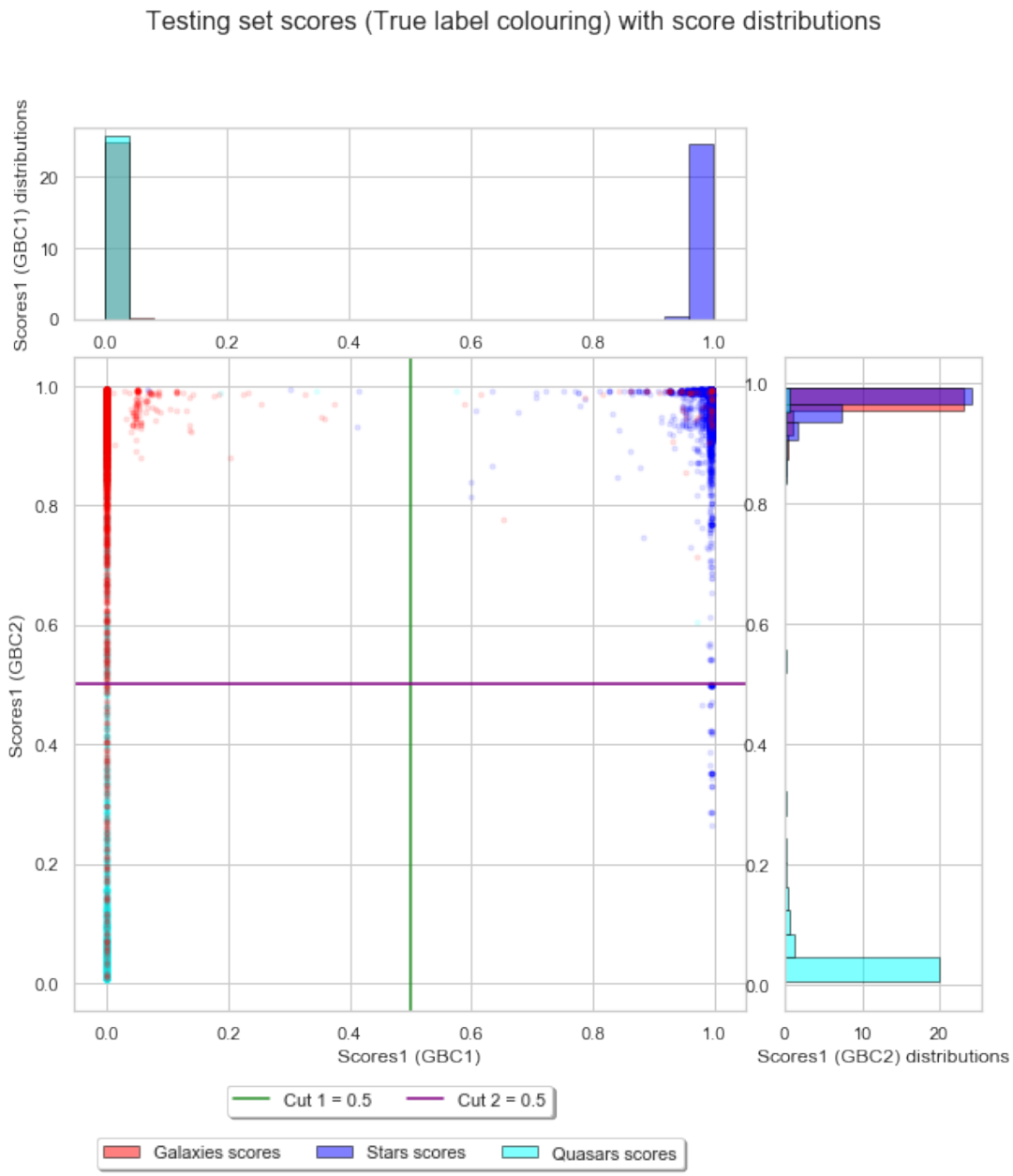
6. r (0.005879)



Το $redshift$ είναι η πιο σημαντική μεταβλητή και σε αυτόν τον ταξινομητή, ενώ οι υπόλοιπες μεταβλητές παίζουν λιγότερο σημαντικό ρόλο στην διαδικασία.

Επιπλέον κατασκευάστηκε ένα διάγραμμα με σημεία τα οποία έχουν ως x-συντεταγμένη το **σκορ 1** και ως y-συντεταγμένη το **σκορ 2**. Σε κάθε άξονα προσαρμόστηκε το αντίστοιχο ιστόγραμμα για κάθε σκορ, ούτως ώστε να μελετούνται εύκολα οι κατανομές τους.


Training set scores (True label colouring) with score distributions
Scores1 (GBC1) distributions
Scores1 (GBC2)
Scores1 (GBC1)
Scores1 (GBC2) distributions
Cut 1 = 0.5
Cut 2 = 0.5
Galaxies scores
Stars scores
Quasars scores



Τόσο στο σύνολο εκπαίδευσης όσο και στο σύνολο αξιολόγησης, τα **σκορς 1** και **σκορς 2** των Αστέρων συσσωρεύονται κοντά στο 1. Οι παρατηρήσεις των Γαλαξιών έχουν **σκορς 1** κοντά στο 0 και υψηλά **σκορς 2** (προσεγγίζουν το 1). Τέλος, οι παρατηρήσεις των Κβάζαρς έχουν και τα δύο σκορς κοντά στο 0. Οπότε οι κατανομές των σκορς είναι οι αναμενόμενες.

5.4 Αξιολόγηση Αλγορίθμου

Για να αξιολογηθεί η απόδοση της Gradient Boosting Machine συνοψίζονται τα αποτελέσματα στα confusion matrices και στη συνέχεια υπολογίζεται το συνολικό σφάλμα ταξινόμησης.

Confusion matrices

Παρακάτω παρουσιάζονται τα Confusion matrices και τα αντίστοιχα κανονικοποιημένα Confusion matrices των συνόλων εκπαίδευσης και αξιολόγησης.

Σύνολο εκπαίδευσης				
Πραγματικές ετικέτες	ΑΣΤΕΡΙΑ	12236	1	0
	ΓΑΛΑΞΙΕΣ	63	33115	187
	ΚΒΑΖΑΡΣ	0	723	7709
		ΑΣΤΕΡΙΑ	ΓΑΛΑΞΙΕΣ	ΚΒΑΖΑΡΣ
		Προβλεπόμενες ετικέτες		

Σύνολο εκπαίδευσης				
Πραγματικές ετικέτες	ΑΣΤΕΡΙΑ	$\frac{12236}{12237} \simeq 1$	$\frac{1}{12237} \simeq 0$	$\frac{0}{12237} = 0$
	ΓΑΛΑΞΙΕΣ	$\frac{63}{33365} \simeq 0$	$\frac{33115}{33365} = 0.99$	$\frac{187}{33365} = 0.01$
	ΚΒΑΖΑΡΣ	$\frac{0}{8432} = 0$	$\frac{723}{8432} = 0.09$	$\frac{7709}{8432} = 0.91$
		ΑΣΤΕΡΙΑ	ΓΑΛΑΞΙΕΣ	ΚΒΑΖΑΡΣ
		Προβλεπόμενες ετικέτες		

σφάλμα ταξινόμησης $= \frac{1+63+187+723}{54034} = \frac{974}{54034} = 0.018$

Πραγματικές ετικέτες	Σύνολο αξιολόγησης		
ΑΣΤΕΡΙΑ	12180	7	0
ΓΑΛΑΞΙΕΣ	82	33202	198
ΚΒΑΖΑΡΣ	2	661	7703
	ΑΣΤΕΡΙΑ	ΓΑΛΑΞΙΕΣ	ΚΒΑΖΑΡΣ
	Προβλεπόμενες ετικέτες		

Πραγματικές ετικέτες	Σύνολο αξιολόγησης		
ΑΣΤΕΡΙΑ	$\frac{12180}{12187} \simeq 1$	$\frac{7}{12187} \simeq 0$	$\frac{0}{12187} = 0$
ΓΑΛΑΞΙΕΣ	$\frac{82}{33482} \simeq 0$	$\frac{33202}{33482} = 0.99$	$\frac{198}{33482} = 0.01$
ΚΒΑΖΑΡΣ	$\frac{2}{8366} \simeq 0$	$\frac{661}{8366} = 0.08$	$\frac{7703}{8366} = 0.92$
	ΑΣΤΕΡΙΑ	ΓΑΛΑΞΙΕΣ	ΚΒΑΖΑΡΣ
	Προβλεπόμενες ετικέτες		

σφάλμα ταξινόμησης $= \frac{7+82+198+2+661}{54035} = \frac{950}{54035} = 0.018$

Αν μετακινηθεί το $cut1 = 0.5$ δεν παρατηρείται κάποια σημαντική διαφορά όσο αφορά στην απόδοση του αλγορίθμου. Ωστόσο, μετακινώντας το $cut2 = 0.5$ κάποιες παρατηρήσεις των Κβαζαρς ταξινομούνται ως Γαλαξίες και το αντίστροφο. Οπότε έτσι το σφάλμα ταξινόμησης αυξάνεται, αντί να μειώνεται (όπως παρατηρήθηκε στον αλγόριθμο με τα δύο Νευρωνικά Δίκτυα). Συνεπώς σταθεροποιούνται και τα δύο $cuts$ στο 0.5.

Κεφάλαιο 6

Σύγκριση αποτελεσμάτων

Αφού μελετήσαμε και αξιολογήσαμε την κάθε μέθοδο ξεχωριστά, θα συνεχίσουμε συγκρίνοντας τα αποτελέσματα όλων των μεθόδων, ούτως ώστε να καταλήξουμε σε αυτήν που επιλύει το πρόβλημα της ταξινόμησης του διαθέσιμου συνόλου δεδομένων με μεγαλύτερη επιτυχία.

Θεωρούμε ότι το καλύτερο μοντέλο μηχανικής μάθησης είναι αυτό που κατά την ταξινόμηση των δεδομένων, παρουσιάζει τα χαμηλότερα σφάλματα ταξινόμησης. Στον παρακάτω πίνακα συνοψίζουμε τα σφάλματα ταξινόμησης που είχαμε σε κάθε μέθοδο, για το σύνολο εκπαίδευσης και για το σύνολο αξιολόγησης:

Σύγκριση Αποτελεσμάτων		
Μέθοδος	**Σφάλμα ταξινόμησης (Σύνολο εκπαίδευσης)**	**Σφάλμα ταξινόμησης (Σύνολο αξιολόγησης)**
Γραμμικοί ταξινομητές		
Fisher [διαχωρισμός φασικού χώρου]	0.17	0.17
Fisher [όρια απόφασης sklearn]	0.29	0.29
LDA+SVD [διαχωρισμός φασικού χώρου]	0.18	0.19
LDA+SVD [όρια απόφασης sklearn]	0.16	0.16

Μη γραμμικοί ταξινομητές		
Αλγόριθμος με 2 Νευρωνικά Δίκτυα [2 Binary ταξινομήσεις]	0.05	0.05
Αλγόριθμος με 2 συνθήκες (βάσει 'redshift' και αποτελέσματος ενός Νευρωνικού Δικτύου)	$0.036 \simeq 0.04$	$0.035 \simeq 0.04$
Ενδυναμωμένα Δέντρα Απόφασης (2 Gradient Boost Machines)	$0.018 \simeq 0.02$	$0.018 \simeq 0.02$

Η απόδοση των γραμμικών ταξινομητών δεν συγκρίνεται με αυτή των μη γραμμικών, αφού οι πρώτοι παρουσιάζουν υψηλά σφάλματα ταξινόμησης, από 16%-29%. Η προσοχή μας, λοιπόν, εστιάζεται στην σύγκριση των μη γραμμικών ταξινομητών, οι οποίοι ανταγωνίζονται ο ένας τον άλλο, αφού τα σφάλματα ταξινόμησης τους έχουν μικρές διαφορές.

Συγκεκριμένα, όσο αφορά στους αλγορίθμους των Νευρωνικών Δικτύων τα χαμηλότερα σφάλματα που παρουσιάστηκαν υπολογίστηκαν στο 4% του συνόλου δεδομένων, εκ των οποίων τα περισσότερα λάθος ταξινομημένα αντικείμενα ήταν τα Κβάζαρς (το 13% των Κβάζαρς ταξινομήθηκε εσφαλμένα στους Γαλαξίες).

Στη συνέχεια, με την υλοποίηση των Ενδυναμωμένων Δέντρων Απόφασης, παρατηρήσαμε ότι το σφάλμα ταξινόμησης είναι ακόμα χαμηλότερο, αφού μόνο το 2% των δεδομένων ταξινομήθηκε σε λανθασμένη κατηγορία, εκ των οποίων και πάλι τα περισσότερα λάθος ταξινομημένα αντικείμενα ήταν τα Κβάζαρς (8%-9% εκ των οποίων ταξινομήθηκε στους Γαλαξίες). Αυτό πολύ πιθανό να οφείλεται στο γεγονός ότι τα Κβάζαρς αποτελούν το μικρότερο μέρος του συνόλου δεδομένων, οπότε ο ταξινομητής δεν διαθέτει όσο μεγάλο αριθμό διανυσμάτων διαθέτει από τις άλλες δύο κλάσεις, για να εκπαιδευτεί στον καλύτερο δυνατό βαθμό.

Συνεπώς, από τα πιο πάνω, συμπεραίνουμε ότι για το συγκεκριμένο σύνολο δεδομένων, ο συνδυασμός δύο Gradient Boost Machines έχει τα χαμηλότερα σφάλματα ταξινόμησης, οπότε κρίνεται ως ο καταλληλότερος ταξινομητής για το δεδομένο πρόβλημα.

Συντομογραφίες

SDSS	Sloan Digital Sky Survey
CCD	Charge-coupled device
X train	Σύνολο εκπαίδευσης
X test	Σύνολο αξιολόγησης
LDA	Linear Discriminant Analysis
SVD	Singular Value Decomposition
MLP	Multi-Layer perceptron
BGD	Batch Gradient Descent

Αναφορές

[1]: *"The Master Algorithm: How the Quest for he ultimate learning machine will remake our world"*, P. Domingos, 2017, Penguin Books

[2]: (Paper) *"Using Discriminant Analysis for Multi-Class Classification: An Experimental Investigation"*, Tao Li, Shenghuo Zhu, and Mitsunori Ogihara, Knowledge and Information Systems 10, no. 4 (2006): 453–72

Πηγές

Βιβλιογραφία

1. *"Pattern Recognition and Machine Learning"*, C. M. Bishop, 2009, Springer
2. *"Pattern Recognition"*, S. Theodoridis, K. Koutroumbas, 2009, Berlington: Elsevier
3. (Paper) *"Greedy Function Approximation: A Gradient Boosing Machine"*, J. Friedman, 2001, IMS 1999 Reitz Lecture
4. (Paper) *"Scikit-learn: Machine Learning in Python"*, Pedregosa et al., JMLR 12, pp. 2825-2830, 2011
5. (Paper) *"The Elements of Statistical Learning (Data Mining, Inference and Prediction)"*, J. Friedman, T. Hastie, R. Tibshirani, 2008

Ιστοσελίδες

1. `https://www.sdss.org`
2. `https://www.sdss.org/press-releases/the-sloan-digital-sky-survey-opens-a-new-public-view-of-the-sky/`
3. `https://www.kaggle.com/ashishsaxena2209/sloan-digital-sky-survey-dr12-server-data`
4. `https://www.eso.org/public/teles-instr/alma/receiver-bands/`
5. `http://skyserver.sdss.org/dr7/en/proj/advanced/color/sdssfilters.asp`
6. `http://www.sdss3.org/dr8/imaging/imaging_basics.php`
7. `https://www.loc.gov/rr/scitech/mysteries/universe.html`

8. https://www.britannica.com/science/redshift

9. https://www.esa.int/Our_Activities/Space_Science/What_is_red_shift

10. https://en.wikipedia.org/wiki/Photometry_(astronomy)

11. https://en.wikipedia.org/wiki/Photometric_system

12. https://scikit-learn.org/stable/modules/lda_qda.html#estimation-algorithms

13. https://www.apsl.net/blog/2017/07/18/using-linear-discriminant-analysis-lda-data-explore-step-step/

14. https://python-for-multivariate-analysis.readthedocs.io/a_little_book_of_python_for_multivariate_analysis.html

15. https://en.wikipedia.org/wiki/Whitening_transformation

16. https://stats.stackexchange.com/questions/109747/why-is-pythons-scikit-learn-lda-not-working-correctly-and-how-does-it-compute-l

17. https://towardsdatascience.com/when-to-standardize-your-data-in-4-minutes-f9282190707e

18. https://stats.stackexchange.com/questions/275692/how-do-pca-svd-decorrelate-the-variables

19. https://towardsdatascience.com/metrics-to-evaluate-your-machine-learning-algorithm-f10ba6e38234

20. https://scikit-learn.org/stable/modules/lda_qda.html#lda-qda

21. https://www.python-course.eu/linear_discriminant_analysis.php

22. https://sebastianraschka.com/Articles/2014_python_lda.html

23. https://towardsdatascience.com/curse-of-dimensionality-2092410f3d27

24. https://medium.com/diogo-menezes-borges/give-me-the-antidote-for-the-curse-of-dimensionality-b14bce4bf4d2

25. https://medium.com/@AI_with_Kain/understanding-of-multilayer-perceptron-mlp-8f179c4a135f

26. https://gombru.github.io/2018/05/23/cross_entropy_loss/

27. https://stackoverflow.com/questions/34240703/what-is-logits-softmax-and-softmax-cross-entropy-with-logits

28. https://medium.com/data-science-bootcamp/understand-the-softmax-function-in-minutes-f3a59641e86d

29. http://uc-r.github.io/gbm_regression

30. https://towardsdatascience.com/activation-functions-neural-networks-1cbd9f8d91d6

31. https://medium.com/data-science-bootcamp/understand-the-softmax-function-in-minutes-f3a59641e86d

32. https://www.ibm.com/analytics/hadoop/big-data-analytics

33. https://repository.kallipos.gr/bitstream/11419/3382/1/02_chapter_04.pdf

34. https://www.youtube.com/watch?v=3CC4N4z3GJc, (Parts 1,2,3,4)

35. https://towardsdatascience.com/boosting-algorithm-gbm-97737c63daa3

36. https://machinelearningmastery.com/gentle-introduction-gradient-boosting-algorithm-machine-learning/

37. https://towardsdatascience.com/boosting-algorithm-gbm-97737c63daa3

38. https://medium.com/all-things-ai/in-depth-parameter-tuning-for-gradient-boosting-3363992e9bae

39. https://scikit-learn.org/stable/modules/ensemble.html#gradient-boosting

40. Tensorflow: https://www.tensorflow.org/

41. Scikit-learn: http://scikit-learn.org/stable/