

Π.Μ.Σ. ΤΕΧΝΗΤΗΣ ΝΟΗΜΟΣΥΝΗΣ

ΜΑΘΗΜΑ : ΠΡΟΧΩΡΗΜΕΝΑ ΘΕΜΑΤΑ ΜΗΧΑΝΙΚΗΣ ΜΑΘΗΣΗΣ

ΔΙΔΑΣΚΩΝ : ΤΣΟΥΜΑΚΑΣ ΓΡΗΓΟΡΙΟΣ

ΟΜΑΔΑ : ΠΑΠΑΤΖΕΛΟΣ ΣΠΥΡΙΔΩΝ Α.Μ. 43, ΚΙΛΗΣ ΝΙΚΟΛΑΟΣ Α.Μ. 34

# 1. ΟΡΙΣΜΟΣ ΠΡΟΒΛΗΜΑΤΟΣ

Στα πλαίσια αυτής της εργασίας, στόχος μας είναι να ασχοληθούμε με προβλήματα ταξινόμησης ροών δεδομένων σε πολλαπλές ετικέτες (“*multi-label*”), που ταυτόχρονα χαρακτηρίζονται και από ανομοιογένεια στο πλήθος των κλάσεων των δεδομένων τους (“*class imbalance*”). Επιπλέον, θα συζητηθεί και η περαιτέρω ανάλυση του συστήματος προς υλοποίηση, με μεθόδους ερμηνείας των αποτελεσμάτων του (“*explainable system*”).

# 2. ΠΕΡΙΓΡΑΦΗ ΣΥΝΟΛΟΥ ΔΕΔΟΜΕΝΩΝ

Το σύνολο δεδομένων που επιλέχθηκε ονομάζεται “*emotions*” και μπορεί να βρεθεί στον ακόλουθο σύνδεσμο: <http://mulan.sourceforge.net/datasets-mlc.html> της βιβλιοθήκης “*MULAN*” [1]. Το σύνολο αυτό αποτελείται από 100 μουσικά κομμάτια, τα οποία ανήκουν σε 7 διαφορετικά είδη μουσικής και πιο συγκεκριμένα στα “*Classical*”, “*Reggae*”, “*Rock*”, “*Pop*”, “*Hip-Hop*”, “*Techno*” και “*Jazz*”. Η συλλογή δημιουργήθηκε από 233 μουσικά άλμπουμ με επιλογή τριών τραγουδιών από το καθένα. Για κάθε κομμάτι επιλέχθηκε ένα τμήμα χρονικής διάρκειας 30 δευτερολέπτων, πάντα μετά των πρώτων 30 δευτερολέπτων του κομματιού. Τα τμήματα αυτά αποθηκεύτηκαν και μετατράπηκαν σε συχνότητα δειγματοληψίας 22050 Hz με 16 bit per sample και μονοφωνικά. Ακολούθως εξήχθησαν τα χαρακτηριστικά των κομματιών από το εργαλείο ανοικτού λογισμικού “*Marsyas tool*” [2], το οποίο λόγω της ταχύτητας και ευελιξίας του συνήθως επιλέγεται για κατηγοριοποίηση μουσικών συναισθημάτων.

Τα 72 χαρακτηριστικά του συνόλου δεδομένων μπορούν να ταξινομηθούν σε δύο κατηγορίες. Η πρώτη αφορά ρυθμικά χαρακτηριστικά (“*rhythmic*”) και η δεύτερη χαρακτηριστικά χροιάς (“*timbre*”). Αυτές οι κατηγορίες έχουν υψηλή συσχέτιση με το ανθρώπινο συναίσθημα. Τα ρυθμικά εξήχθησαν από τις περιοδικές αλλαγές του ιστογράμματος ρυθμού (“*beat histogram*”) και είναι 8 στον αριθμό τους. Τα χροιάς προήλθαν από ανάλυση με Short-time Fourier Transform (“*STFT”*) και Mel frequency cepstral coefficients (“*MFCCs”*) στα αρχεία ήχου. Συνολικά, κρατήθηκαν οι πρώτοι 13 MFCCs και 3 χαρακτηριστικά από την εφαρμογή STFT. Για καθένα από τα 16 αυτά χαρακτηριστικά, υπολογίστηκαν επιπλέον και η μέση τιμή, η τυπική απόκλιση, η μέση τιμή της τυπικής απόκλισης και η τυπική απόκλιση της τυπικής απόκλισης για κάθε πλαίσιο ανάλυσης, οπότε τελικά οδηγηθήκαμε σε 64 χαρακτηριστικά. Όσον αφορά τον αριθμό των παραδειγμάτων με αυτά τα χαρακτηριστικά, συνολικά απαριθμούν τα 593.

Για την τοποθέτηση των ετικετών στα παραδείγματα έγινε χρήση του μοντέλου Tellegen-Watson-Clark, το οποίο αποτελεί έναν ισχυρό τρόπο οργάνωσης μουσικών συναισθημάτων και έγινε περιορισμός στον αριθμό των ετικετών που παρήγαγε. Ειδικότερα, η τελική οργάνωση των ετικετών επιτεύχθηκε με 6 ζεύγη : “Amazed-surprised”, “Happy-pleased”, “Relaxing-calm”, “Quiet-still”,”Sad-lonely” και “Angry-fearful”.

# 3. ΜΕΘΟΔΟΙ ΠΡΟΣ ΔΙΕΡΕΥΝΗΣΗ

1. Class imbalance
   1. Μέθοδος : REsampling MultilabEl datasets by Decoupling highly ImbAlanced Labels (REMEDIAL) [3].
   2. Μέθοδος : Multilabel Synthetic Minority Over-sampling Technique (MLSMOTE) [4].
2. Multi-label learning
   1. Μέθοδος : Binary relevance (BR) [5].
   2. Μέθοδος : Multi-label KNN (MLKNN) [6].
3. Explainable system
   1. Μέθοδος : Local Interpretable Model-agnostic Explanations (LIME) [7].
   2. Μέθοδος : Multi-label Rule learning [8].

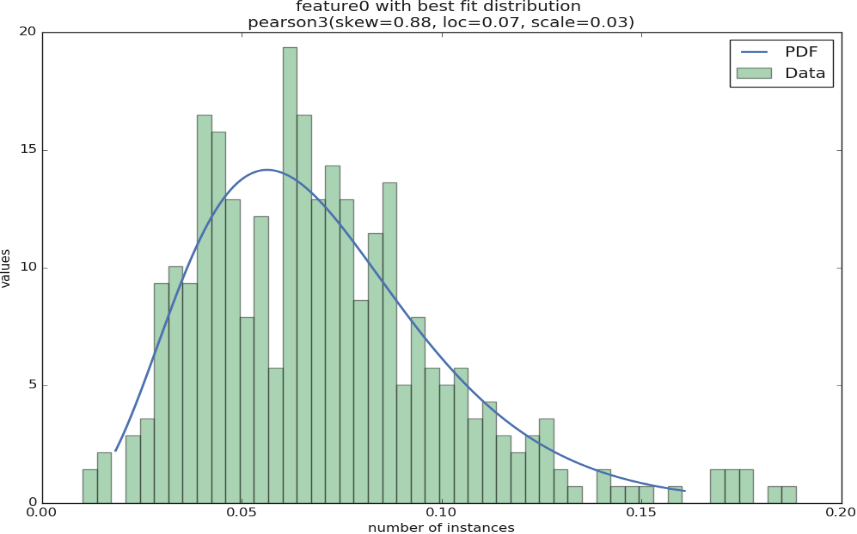
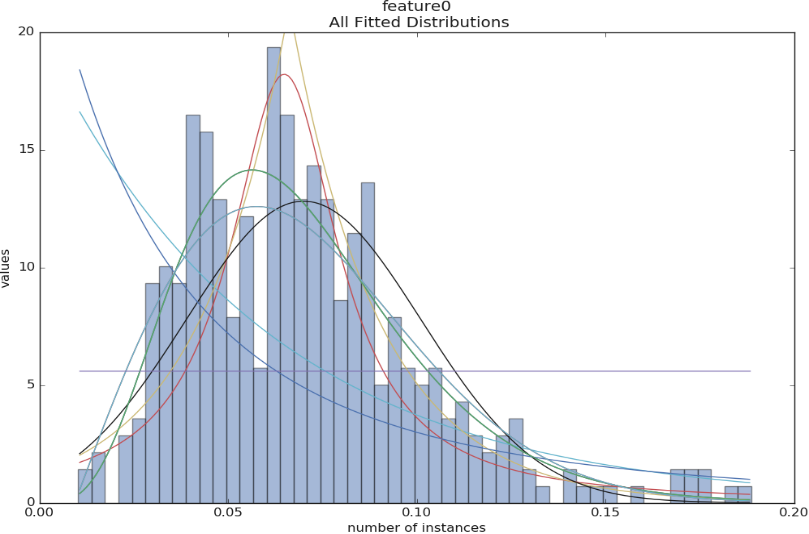
Ο κώδικας υλοποίησης των παραπάνω τεχνικών, μπορεί να βρεθεί στον ακόλουθο σύνδεσμο :

<https://github.com/NickKilis/AdvancedML-Project>

# 4. ΠΡΟΕΠΕΞΕΡΓΑΣΙΑ ΔΕΔΟΜΕΝΩΝ

## 4.1. Κατανομές χαρακτηριστικών

Αρχικά αναζητήσαμε την κατανομή των χαρακτηριστικών των παραδειγμάτων και για κάθε χαρακτηριστικό βρήκαμε την καταλληλότερη καμπύλη που περιγράφει την κατανομή του. Υπήρχαν 89 διαθέσιμες κατανομές, αλλά για λόγους ταχύτητας επιλέξαμε να γίνει η αναζήτηση στις 11 πιο συχνές. Το πρόγραμμα αποθηκεύει αυτόματα τις εικόνες στο φάκελο “*feature\_distributions*”. Ακολούθως φαίνονται τα αποτελέσματα για το πρώτο χαρακτηριστικό.



Εικόνα 1. Πιθανές κατανομές χαρακτηριστικού. Εικόνα 2. Καταλληλότερη κατανομή χαρακτηριστικού.

Μετά από τη συλλογή όλων των κατανομών καταλήγουμε σε 8 από τις 11 με συχνότητες εμφάνισης που φαίνονται στην παρακάτω εικόνα :



Εικόνα 3. Συχνότητες εμφάνισης κατανομών χαρακτηριστικών.

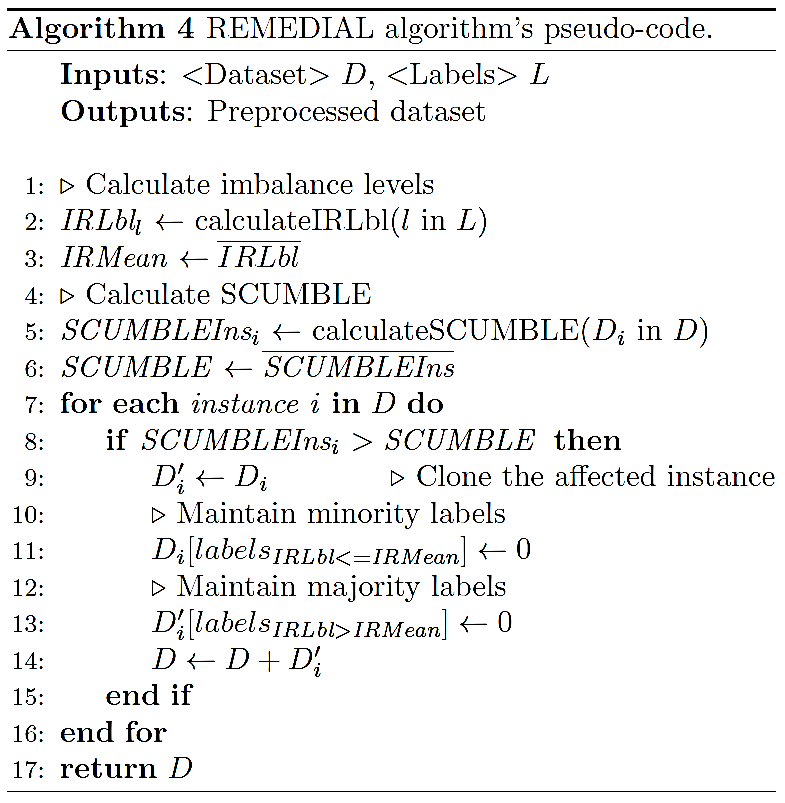
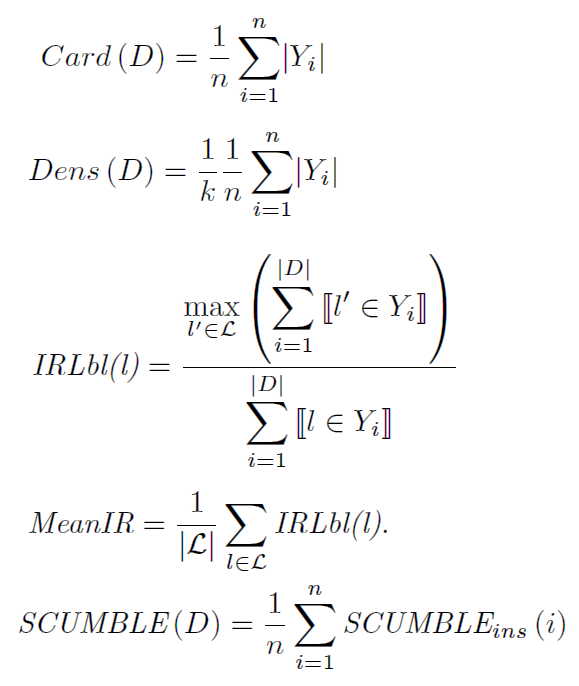
## 4.2. Κλιμάκωση

Πριν την εφαρμογή των δεδομένων στους κατηγοριοποιητές, εφαρμόζουμε την κατάλληλη κλιμάκωση (scaling) σε αυτά, ανάλογα με την κατανομή τους. Εδώ, μπορούμε να επιλέξουμε ανάμεσα από πέντε τρόπους κλιμάκωσης. Από τον “standard”, τον “minmax”, τον “robust”, τον “quantile transformer” και τον ”power transformer”. Ο “standard” υποθέτει πως τα δεδομένα σε κάθε χαρακτηριστικό βρίσκονται σε κανονική κατανομή και τα κλιμακώνει με τέτοιο τρόπο ώστε η κατανομή τους να κεντραριστεί γύρω από το 0 με τυπική απόκλιση ίση με τη μονάδα.

Αν τα δεδομένα δεν είναι αρχικά σε κανονική κατανομή, ο “standard” δεν είναι ο καταλληλότερος για κλιμάκωση. Αν συμβαίνει αυτό ή η τυπική απόκλιση των δεδομένων είναι πολύ μικρή, τότε επιλέγουμε τον “minmax”. O “minmax”, είναι ένας από τους πιο συχνά χρησιμοποιούμενους και συρρικνώνει με τέτοιο τρόπο την κλίμακα έτσι ώστε στο τέλος να έχουμε τιμές μεταξύ 0 και 1 (ή μεταξύ -1 και 1 αν λάβουμε υπόψη και αρνητικές τιμές). Το μειονέκτημά του είναι πως είναι ευαίσθητος σε έκτοπες τιμές (outliers). Ο “robust” λαμβάνει υπόψη του και τις έκτοπες τιμές χρησιμοποιώντας αντί για το μέγιστο και το ελάχιστο. Όσον αφορά την κλιμάκωση με τον “quantile transformer ”, χρησιμοποιήσαμε στην εσωτερική του παράμετρο “n\_quantiles”, τιμή ίση με το μέγεθος του συνόλου δεδομένων στο οποίο εφαρμοζόταν. Τέλος, για τον ”power transformer”, θέσαμε την εσωτερική του παράμετρο “method” ίση με “yeo-johnson”, διότι αυτή μας επιτρέπει να κλιμακώσουμε τόσο αρνητικά όσο και θετικά δεδομένα. Οι δύο τελευταίοι τρόποι κλιμάκωσης ωθούν τα δεδομένα προς μια “πιο κανονική” κατανομή (Gaussian-like distribution).

# 5. CLASS IMBALANCE: ΑΛΓΟΡΙΘΜΟΣ REMEDIAL

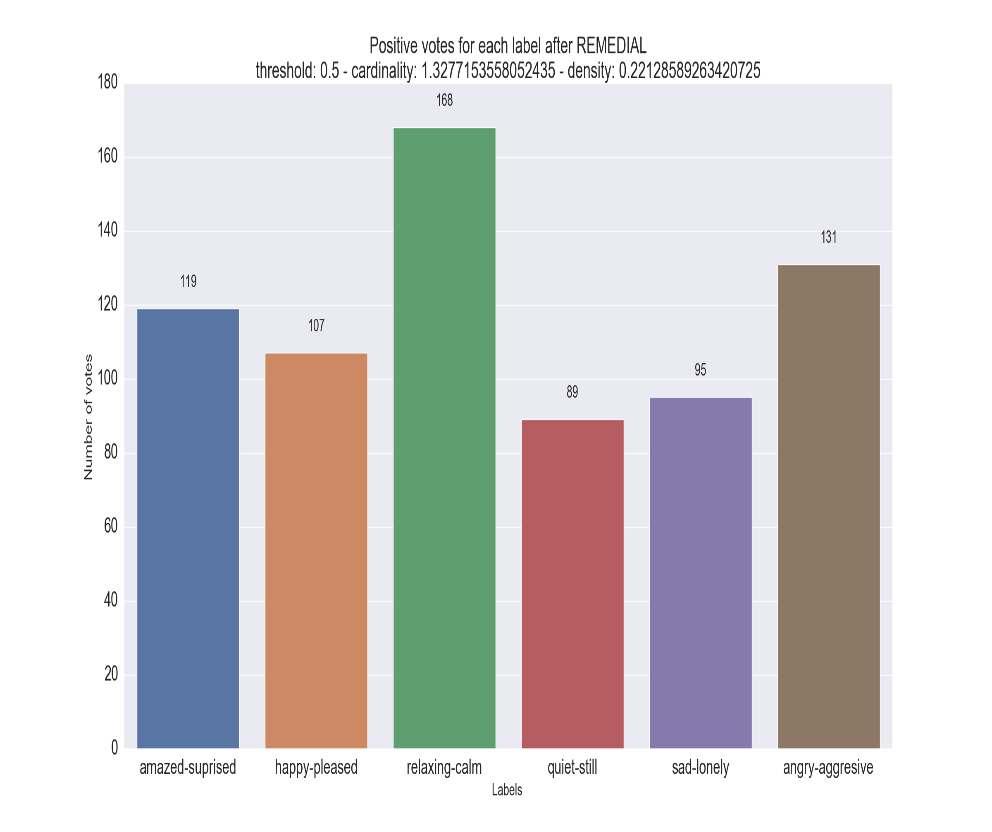
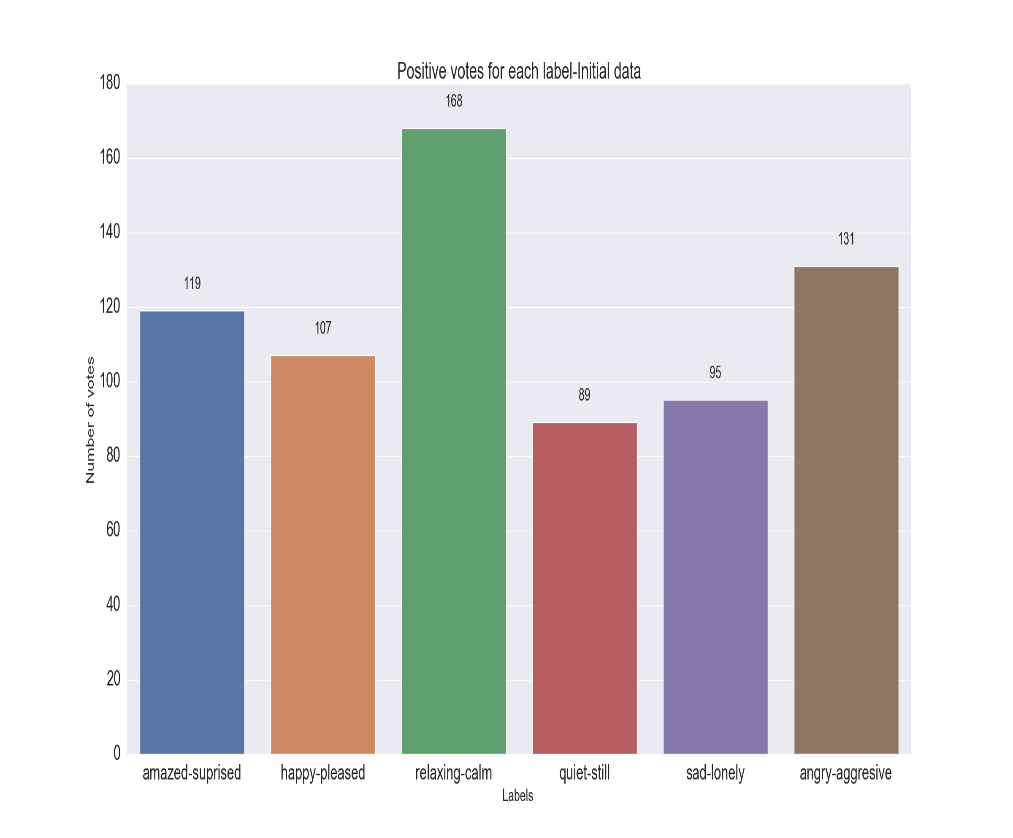
Έπειτα, δοκιμάσαμε να συγκρίνουμε την επίδραση της εφαρμογής του αλγορίθμου REMEDIAL (“*REsampling MultilabEl datasets byDecoupling highly ImbAlanced Labels”*) στα αρχικά δεδομένα με σκοπό την αποσύζευξη των “imbalanced” ετικετών. Όπως θα παρατηρήσουμε και από τις επόμενες εικόνες σύγκρισης, πετυχαίνουμε τη μείωση της πληθηκότητας των ετικετών (“cardinality”) και της μετρικής “scumble”, η οποία όταν είναι υψηλή έχουμε ένδειξη ανομοιογένειας.

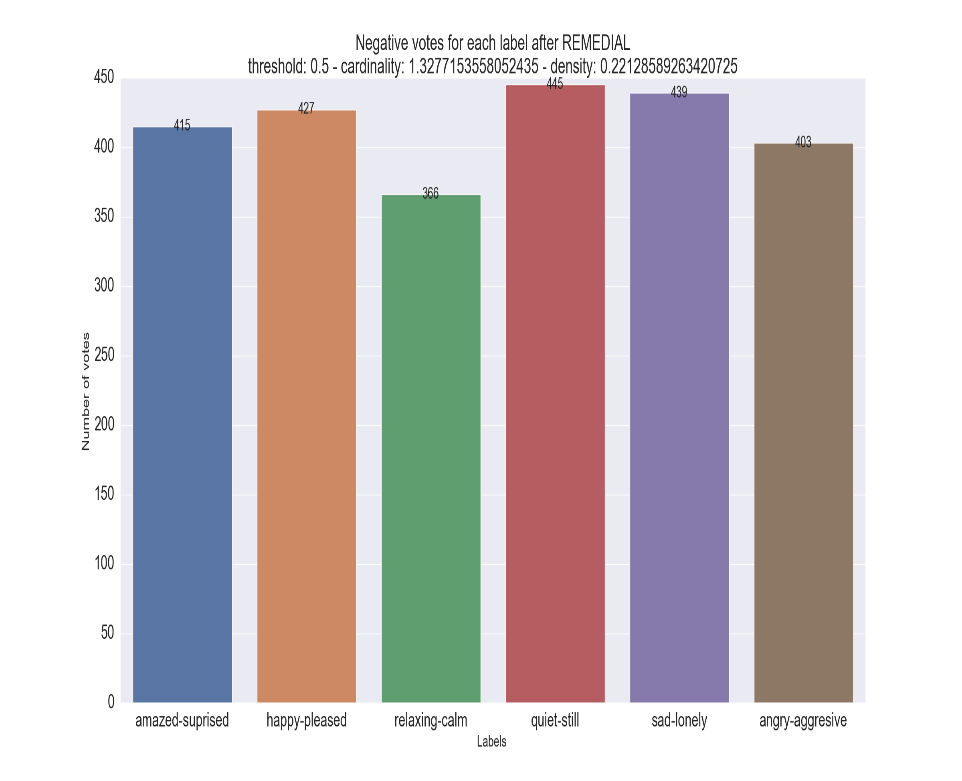
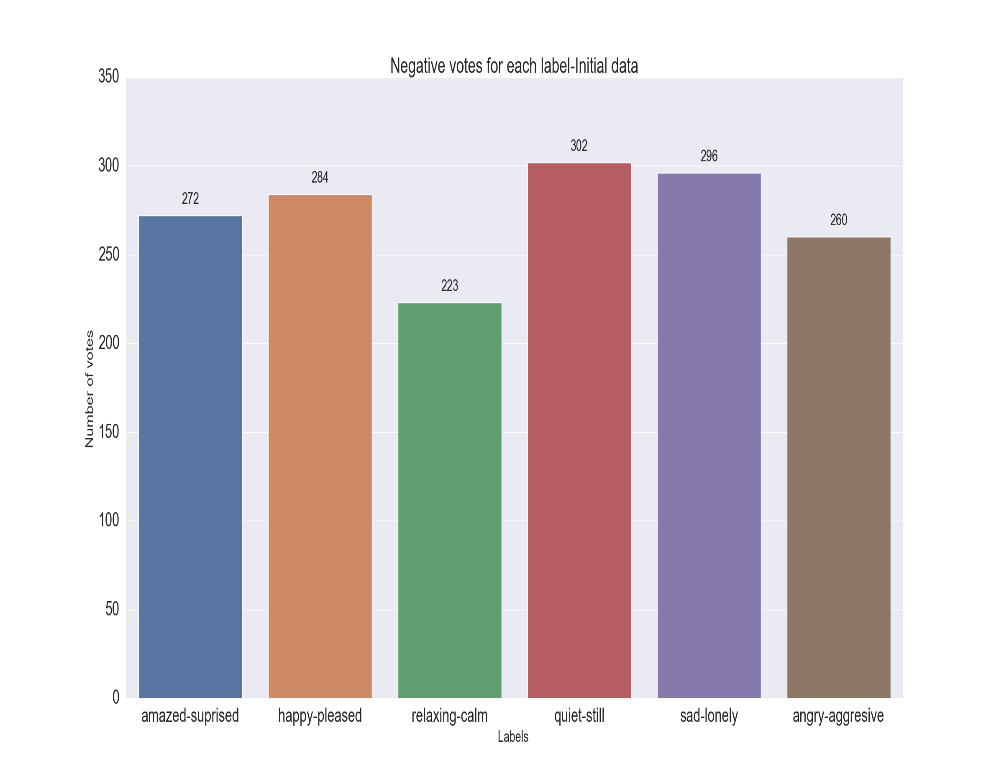
 

Εικόνα 4. Ψευδοκώδικας αλγορίθμου REMEDIAL. Εικόνα 5. Βασικές εξισώσεις αλγορίθμου REMEDIAL.

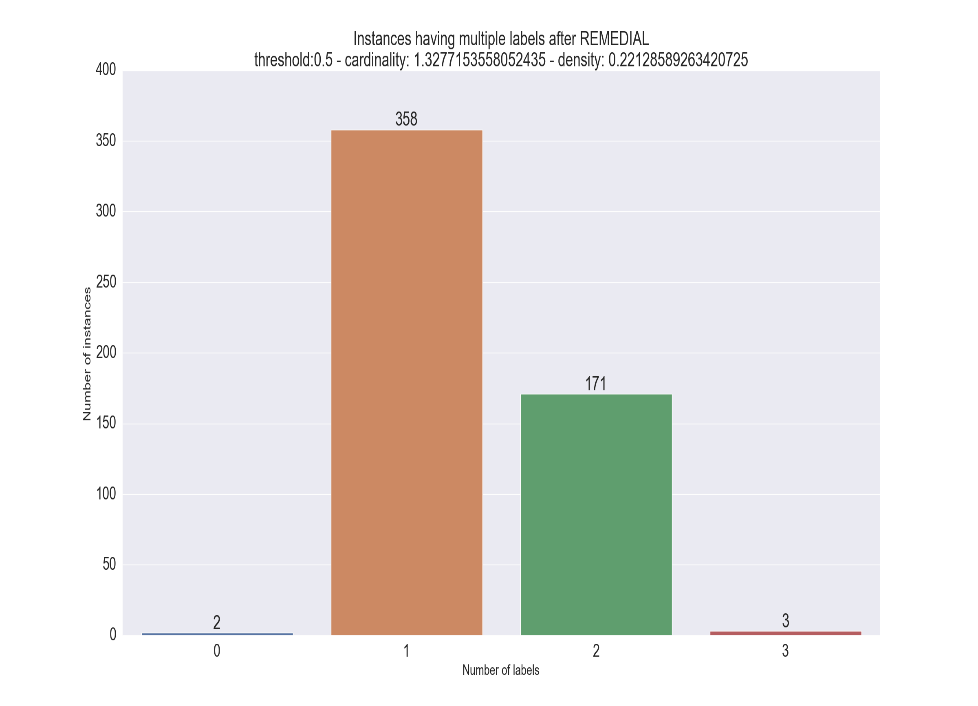
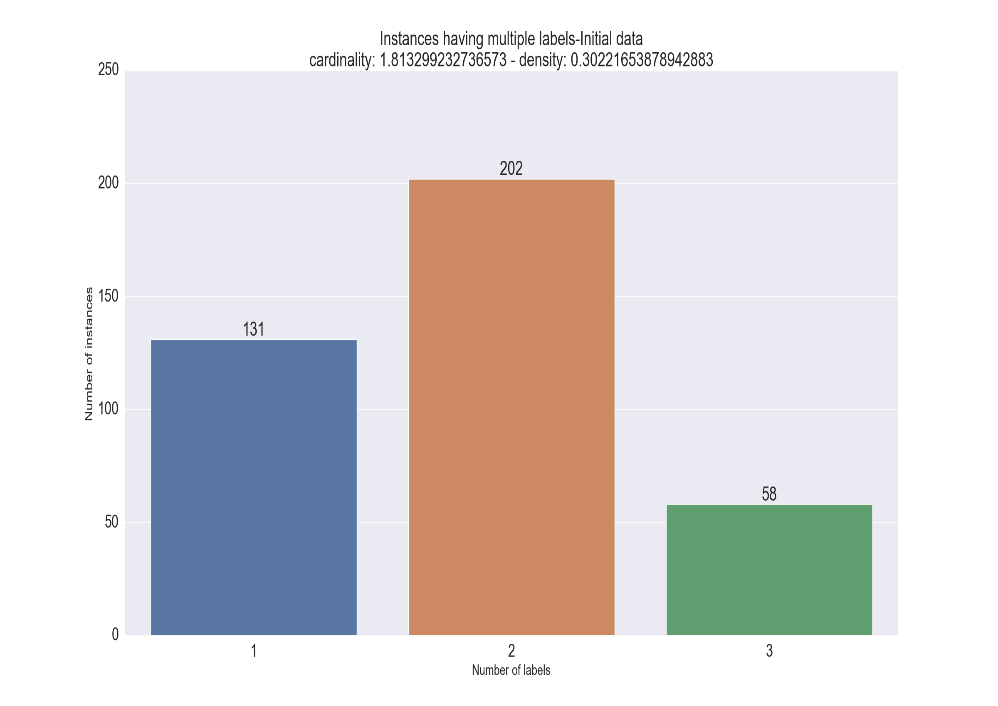
Η επιρροή του αλγορίθμου REMEDIAL, φαίνεται στις ακόλουθες συγκριτικές εικόνες, στις οποίες παρατηρούμε τα εξής :

* Διατήρηση των θετικών ψήφων στις ετικέτες των παραδειγμάτων.
* Προσθήκη αρνητικών ψήφων στις ετικέτες των παραδειγμάτων για την ισοστάθμιση της ανομοιογένειας.
* Μείωση πληθηκότητας (cardinality) από 1.81 στο 1.32.
* Συγκέντρωση των εμφανίσεων των ετικετών στην κύρια διαγώνιο του πίνακα συνεμφάνισης (co-occurrence matrix).
* Μείωση της μετρικής scumble από 0.01286 σε 0.00085.

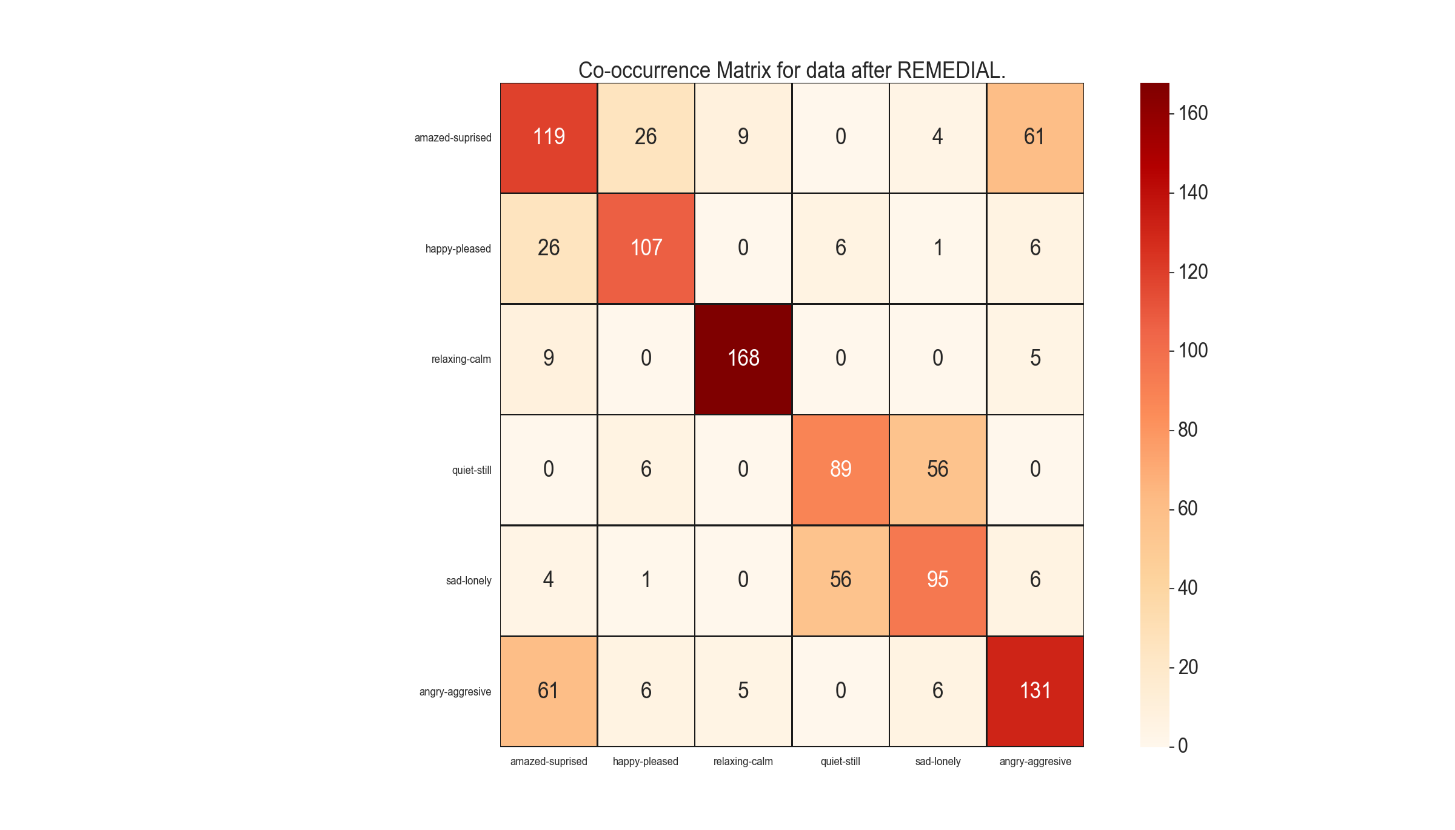
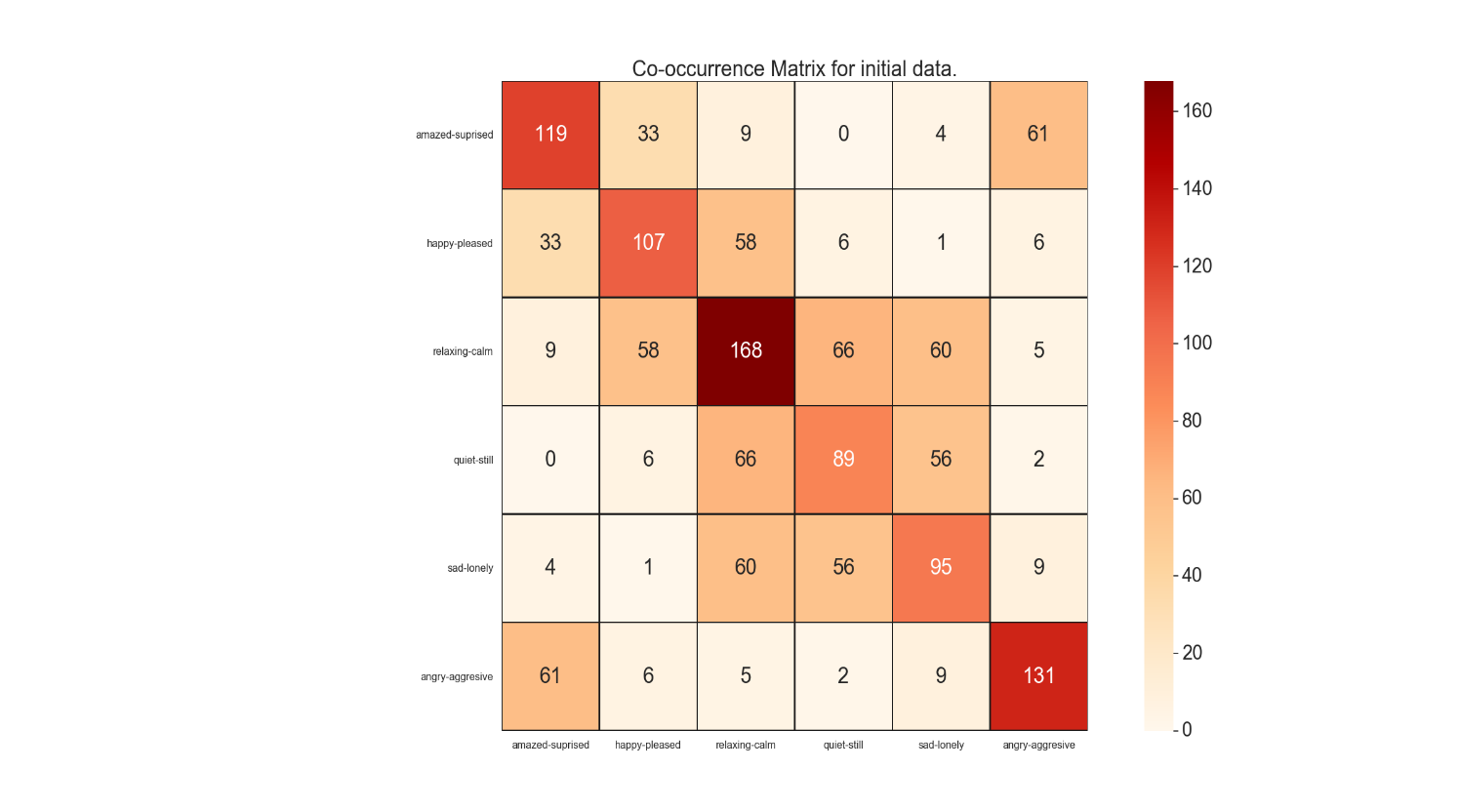
Εικόνα 6. Θετικές ψήφοι πριν την εφαρμογή του REMEDIAL. Εικόνα 7. Θετικές ψήφοι μετά την εφαρμογή του REMEDIAL.



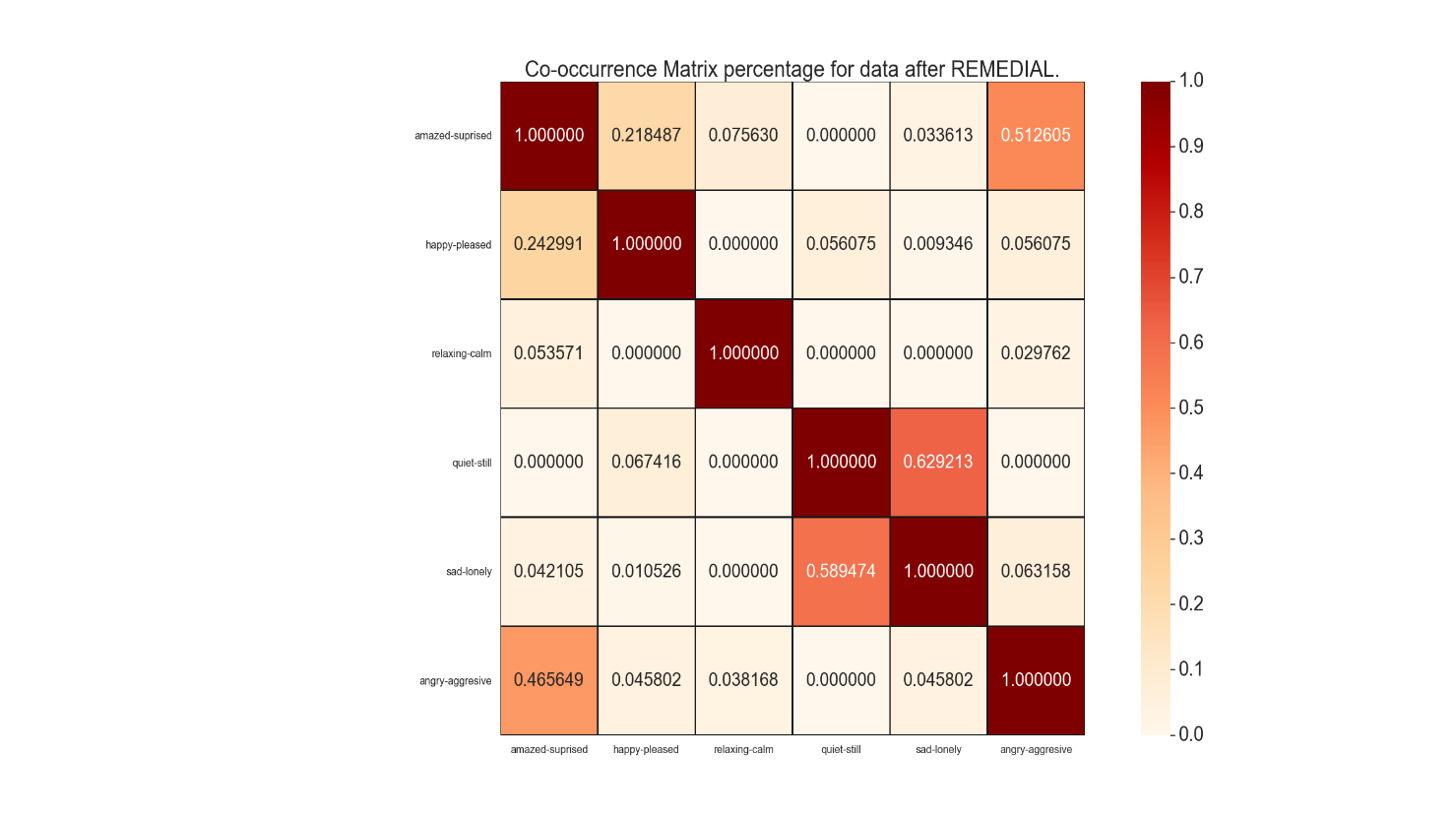
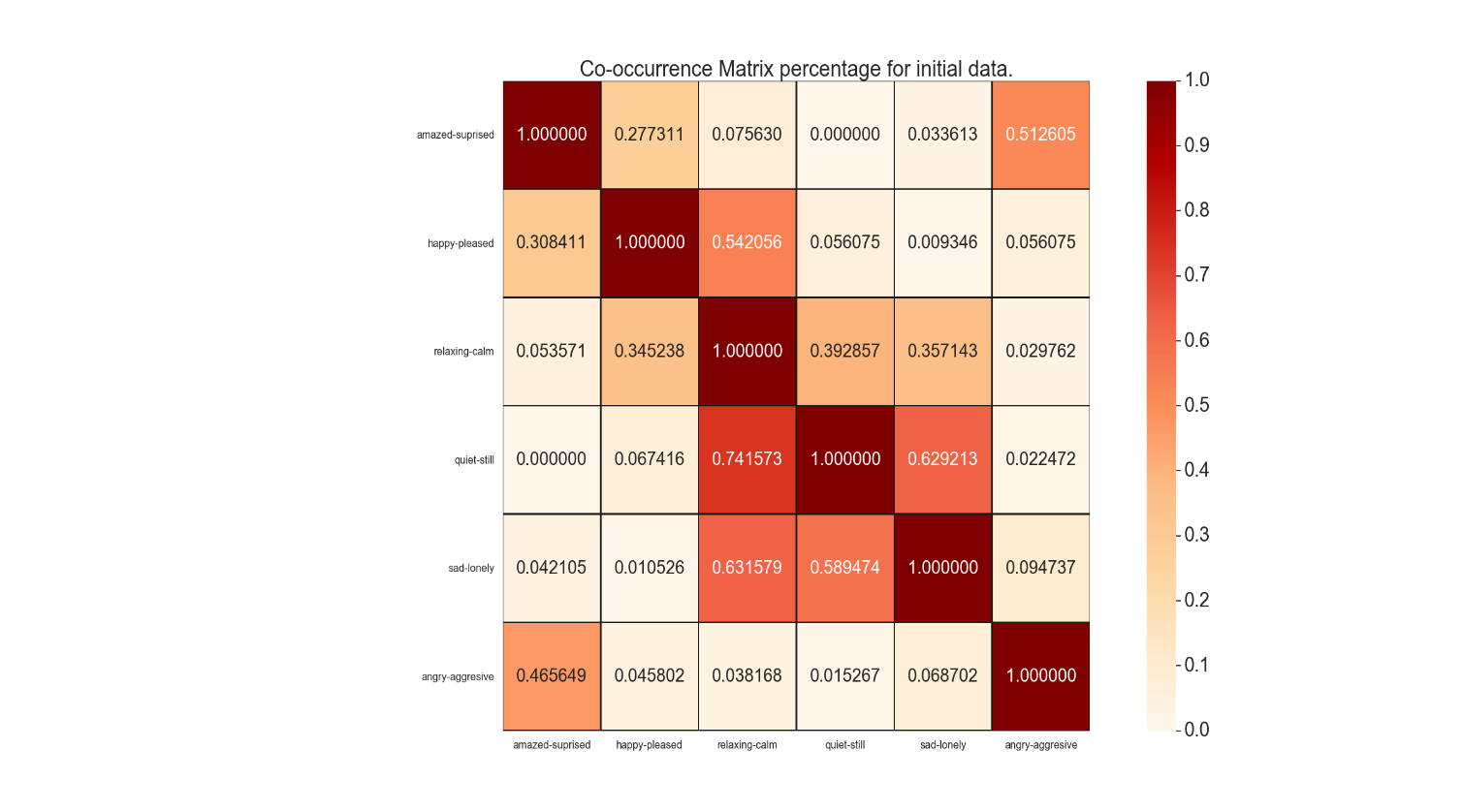
Εικόνα 8. Αρνητικές ψήφοι πριν τον REMEDIAL. Εικόνα 9. Αρνητικές ψήφοι μετά τον REMEDIAL.

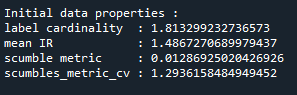
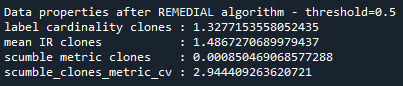


Εικόνα 10. Πληθηκότητα πριν τον REMEDIAL. Εικόνα 11. Πληθηκότητα μετά τον REMEDIAL.



Εικόνα 12. Συνεμφάνιση ετικετών πριν τον REMEDIAL. Εικόνα 13. Συνεμφάνιση ετικετών μετά τον REMEDIAL.

Εικόνα 14. Συνεμφάνιση (%) ετικετών πριν τον REMEDIAL. Εικόνα 15. Συνεμφάνιση (%) ετικετών μετά τον REMEDIAL.

Εικόνα 16. Μετρική “scumble” πριν τον REMEDIAL. Εικόνα 17. Μετρική “scumble” μετά τον REMEDIAL.

# 6. CLASS IMBALANCE: ΑΛΓΟΡΙΘΜΟΣ MLSMOTE

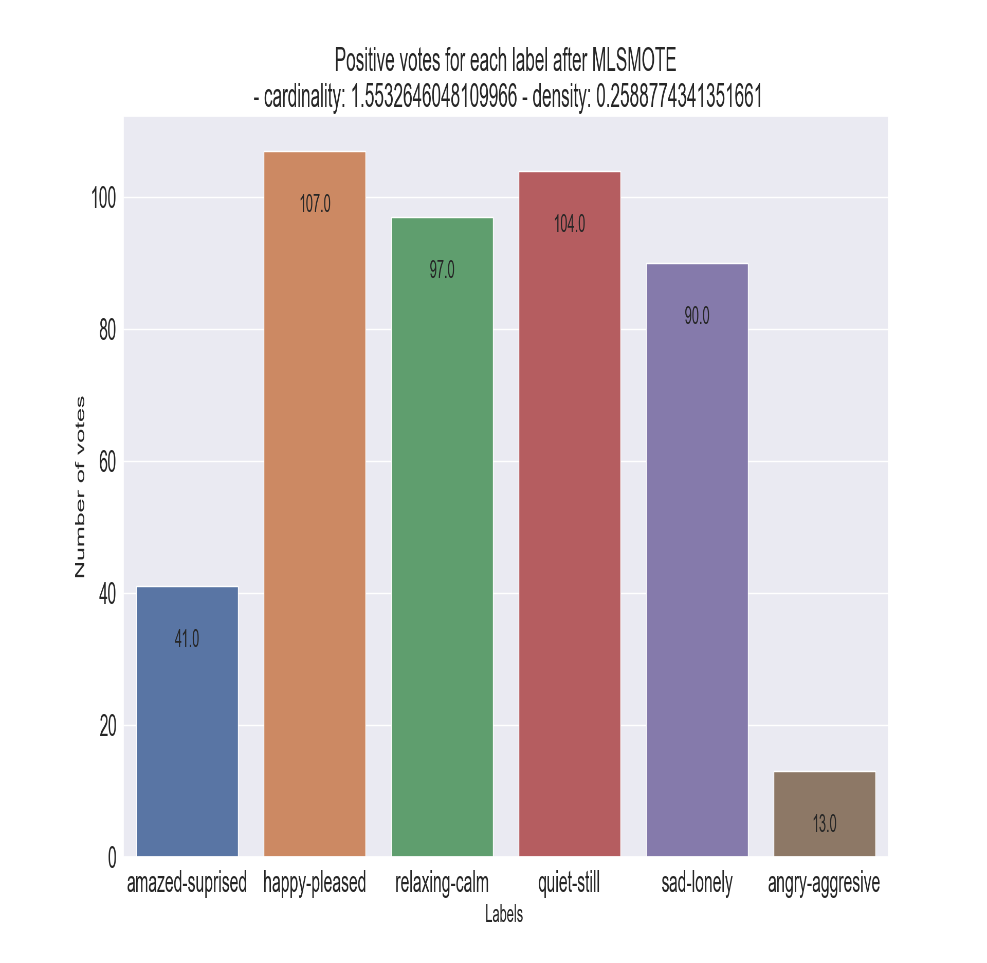
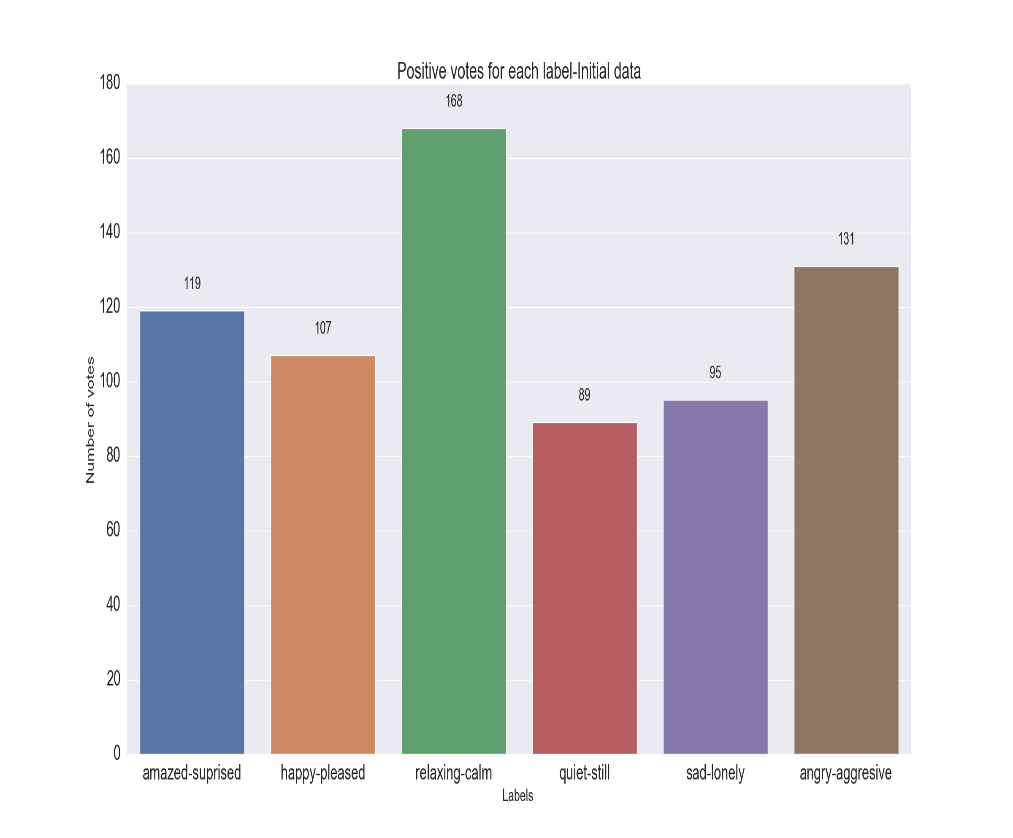
Στη συνέχεια, δοκιμάσαμε να συγκρίνουμε την επίδραση της εφαρμογής του αλγορίθμου MLSMOTE (“*MultiLabel Synthetic Minority Oversampling TEchnique”*), τόσο στα αρχικά δεδομένα όσο και στα δεδομένα μετά την εφαρμογή του αλγορίθμου REMEDIAL. Η δημιουργία του MLSMOTE από τους Francisco Charte et al. [4], ήταν φυσική απόρροια της επέκτασης του διάσημου αλγορίθμου SMOTE (“*Synthetic Minority Oversampling TEchnique”*) [9], η οποία θα εφαρμοζόταν σε δεδομένα με πολλαπλές ετικέτες. Ο κλασικός αλγόριθμος SMOTE, αφού εντοπίσει τα παραδείγματα του συνόλου δεδομένων, που ανήκουν σε μια κατηγορία μειονότητας (minority class), επιλέγει ένα τυχαίο γειτονικό παράδειγμα για κάθε ένα από αυτά και παράγει νέα δείγματα με την ετικέτα της μειονότητας. Για σύνολα δεδομένων με πολλαπλές ετικέτες, θα υπάρχουν πάντα περισσότερες από μια κατηγορίες μειονότητας. Συνεπώς, πρέπει να αναπτυχθεί μια στρατηγική για την επιλογή των καταλληλότερων παραδειγμάτων. Επιπρόσθετα, τα συνθετικά παραδείγματα χρειάζεται να ανήκουν σε πολλαπλές ετικέτες (labelsets), οπότε είναι αναγκαία και μια στρατηγική για αυτή τη διευθέτηση. Πιο συγκεκριμένα, η πρώτη στρατηγική κάνει χρήση μετρικών ανομοιογένειας συνόλων δεδομένων με πολλαπλές ετικέτες (*“IRLblis”*), για να αποφασίσει ποια παραδείγματα τελικά θεωρούνται μειονότητα. Όσον αφορά την υλοποίηση της δεύτερης στρατηγικής, μπορεί να γίνει με τρείς διαφορετικούς τρόπους :

* Intersection : τομή ετικετών που εμφανίζονται στους γείτονες.
* Union : ένωση ετικετών που εμφανίζονται στους γείτονες.
* Ranking : κατάταξη ετικετών ως προς τη συχνότητα εμφάνισής τους.

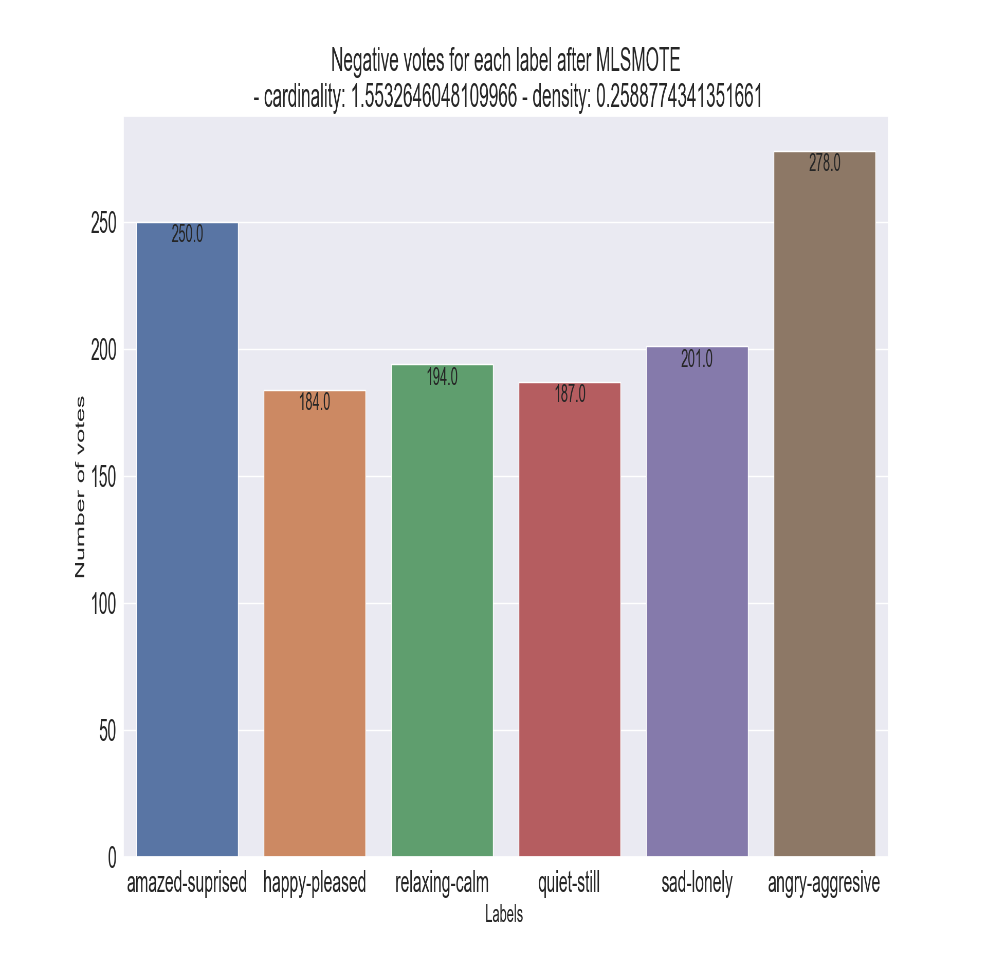
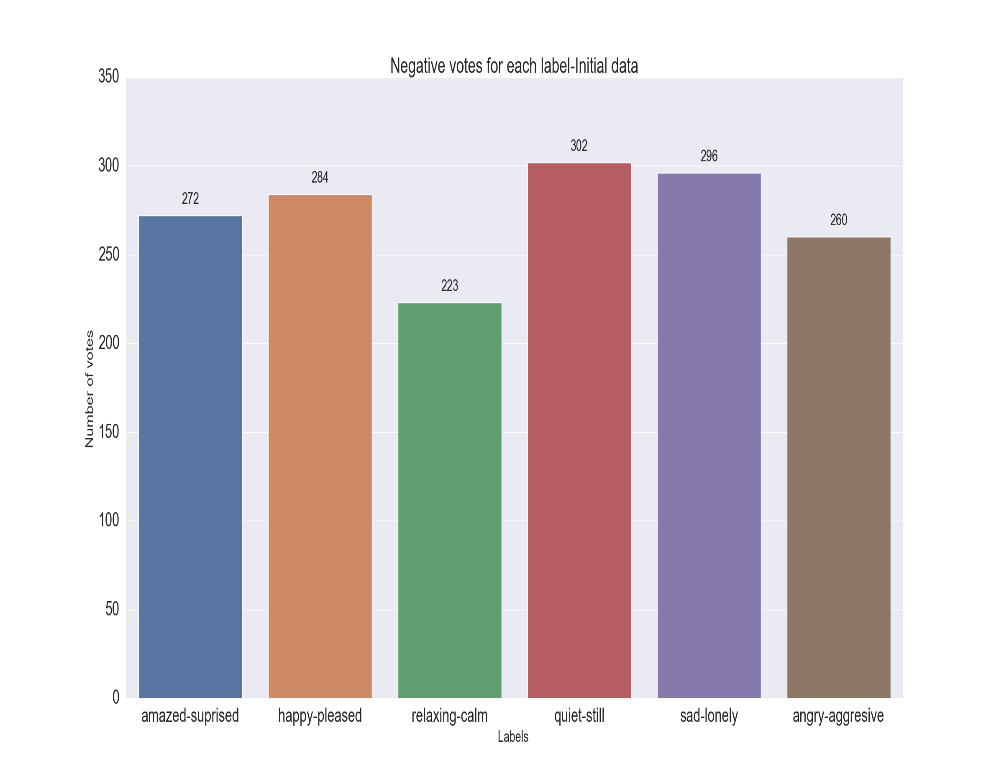
Η υλοποίησή του δεν υπήρχε σε γλώσσα προγραμματισμού Python, οπότε δημιουργήθηκε από την αρχή, με βάση τη δημοσίευση και τις στρατηγικές που αναφέρθηκαν.

Η επιρροή του αλγορίθμου MLSMOTE, φαίνεται στις ακόλουθες συγκριτικές εικόνες, στις οποίες παρατηρούμε τα εξής :

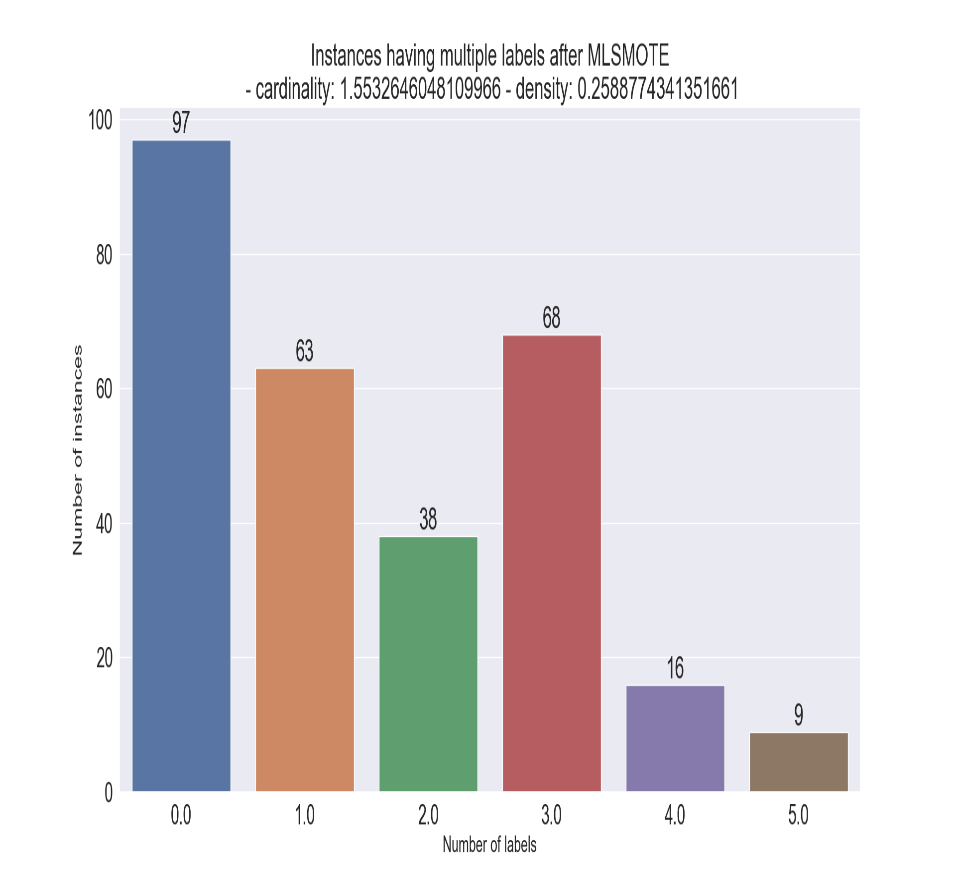
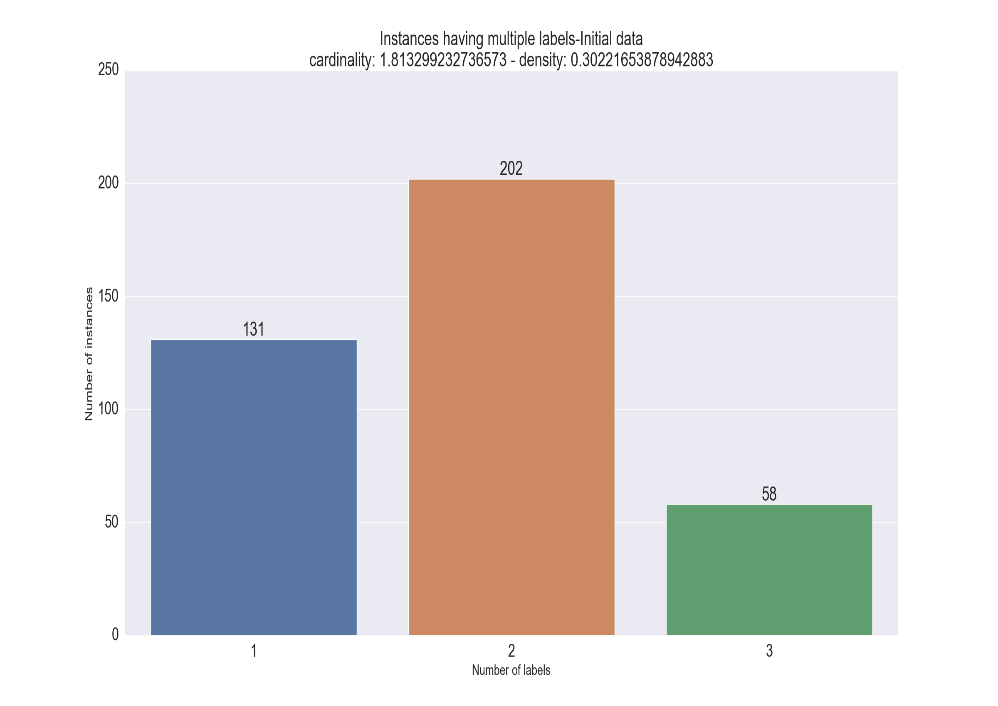
* Δημιουργία συνθετικών παραδειγμάτων με θετικές ψήφους, με βάση τη στρατηγική Ranking.
* Δημιουργία συνθετικών παραδειγμάτων με αρνητικές ψήφους, με βάση τη στρατηγική Ranking.
* Μείωση πληθηκότητας (cardinality) από 1.81 στο 1.55.
* Αύξηση της μετρικής scumble από 0.01286 σε 0.02880.



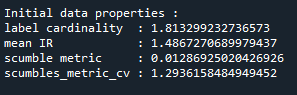
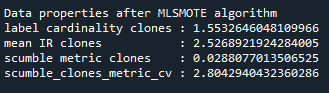
Εικόνα 18. Θετικές ψήφοι πριν τον MLSMOTE. Εικόνα 19. Συνθετικές θετικές ψήφοι μετά τον MLSMOTE.



Εικόνα 20. Αρνητικές ψήφοι πριν τον MLSMOTE. Εικόνα 21. Συνθετικές αρνητικές ψήφοι μετά τον MLSMOTE.



Εικόνα 22. Πληθηκότητα πριν τον MLSMOTE. Εικόνα 23. Πληθηκότητα συνθετικών μετά τον MLSMOTE.

Εικόνα 24. Μετρική “scumble” πριν τον MLSMOTE. Εικόνα 25. Μετρική “scumble” μετά τον MLSMOTE.

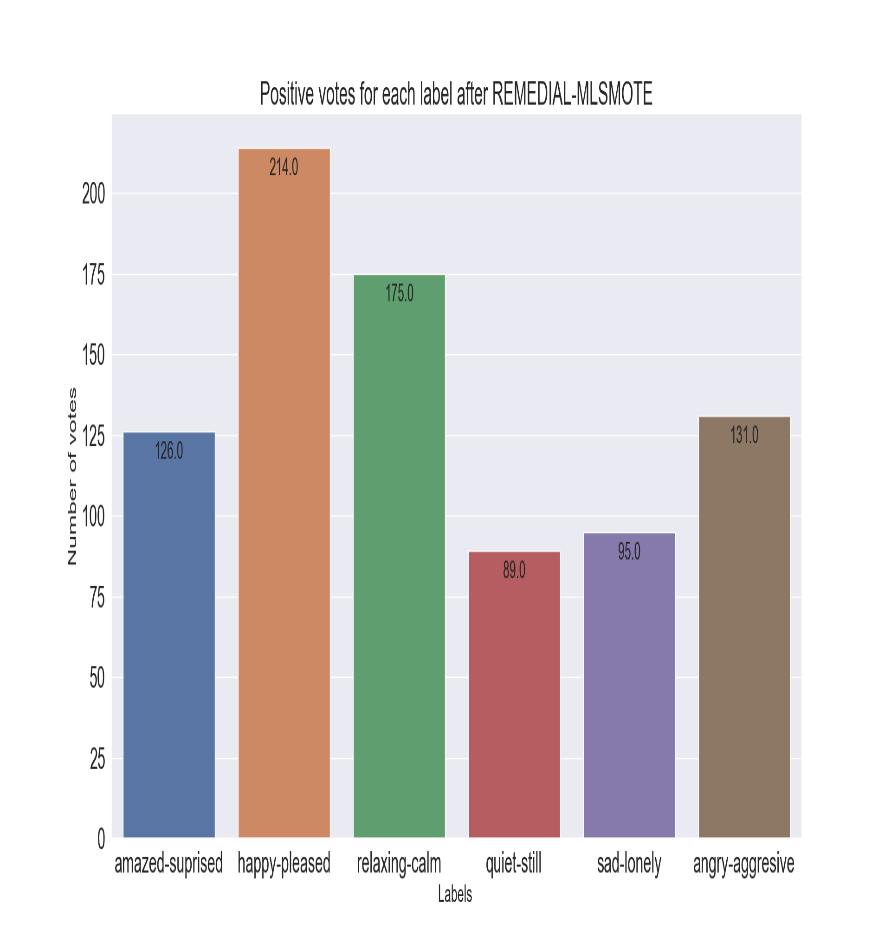
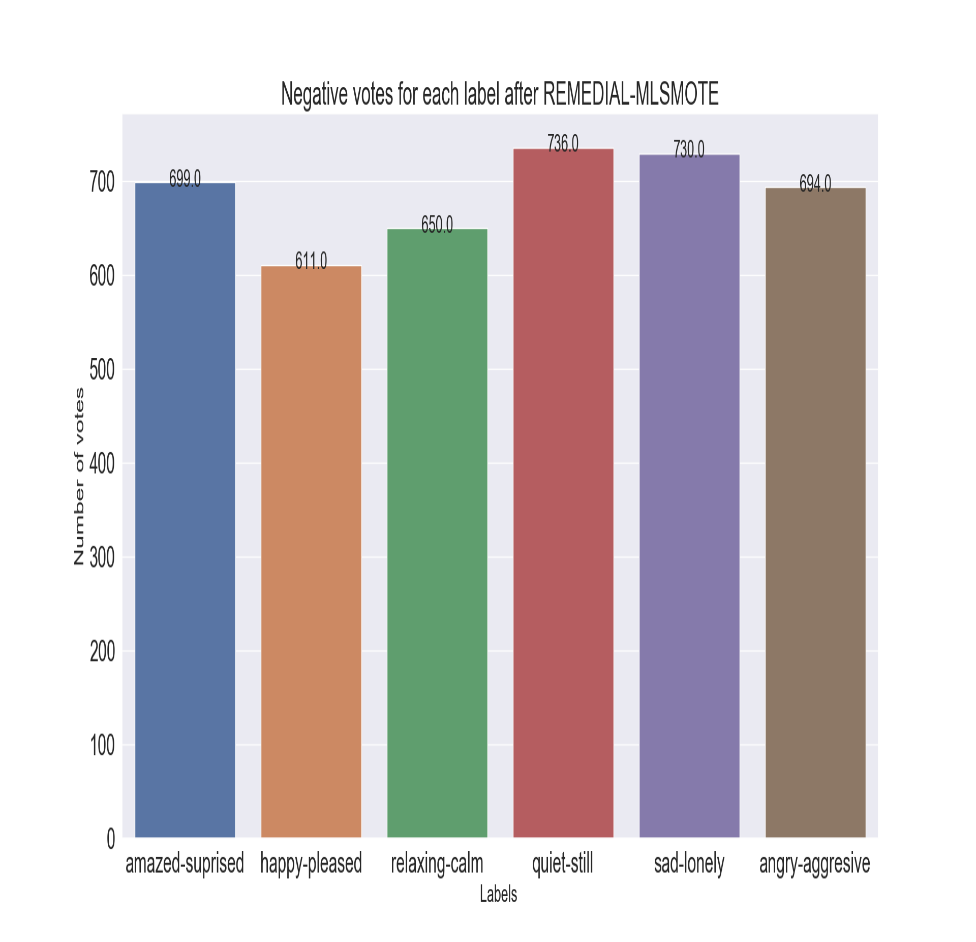
Από την εφαρμογή των προηγούμενων αλγορίθμων στο αρχικό σύνολο δεδομένων, προκύπτουν δύο νέα σύνολα. Επίσης, με εφαρμογή του αλγορίθμου MLSMOTE στο σύνολο μετά τον αλγόριθμο REMEDIAL, παίρνουμε ακόμη ένα σύνολο δεδομένων που εκφράζει το συνδυασμό των δύο αλγορίθμων. Συνολικά, καταλήγουμε σε τέσσερα σύνολα δεδομένων, τα οποία θα συγκριθούν στις επόμενες ενότητες με χρήση δύο διαφορετικών κατηγοριοποιητών :

* “original” : αρχικό σύνολο,
* “original-remedial” : σύνολο μετά την εφαρμογή του REMEDIAL,
* “original-mlsmote” : σύνολο μετά την εφαρμογή του MLSMOTE,
* “remedial-mlsmote” : σύνολο μετά την εφαρμογή του REMEDIAL και του MLSMOTE.

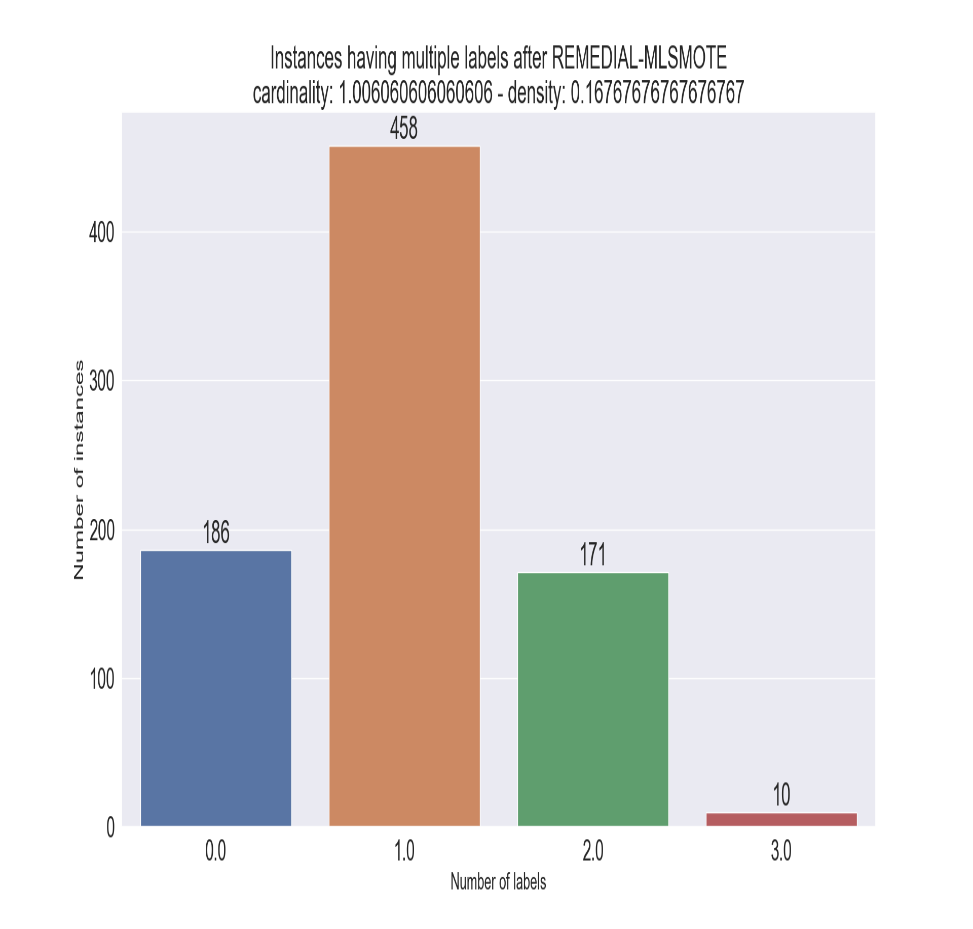
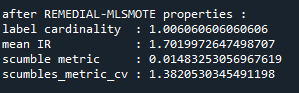
Ο αριθμός των παραδειγμάτων, των τεσσάρων συνόλων που προέκυψαν διαμορφώθηκαν ως εξής :

* “original” : 391 παραδείγματα,
* “original-remedial” : 534 παραδείγματα,
* “original-mlsmote” : 682 παραδείγματα,
* “remedial-mlsmote” : 825 παραδείγματα.

Ακολουθούν οι εικόνες θετικών, αρνητικών ψήφων και πληθηκότητας για το σύνολο δεδομένων “remedial-mlsmote”.

Εικόνα 26. Θετικές ψήφοι μετά τον REMEDIAL-MLSMOTE. Εικόνα 27. Αρνητικές ψήφοι μετά τον REMEDIAL-MLSMOTE.

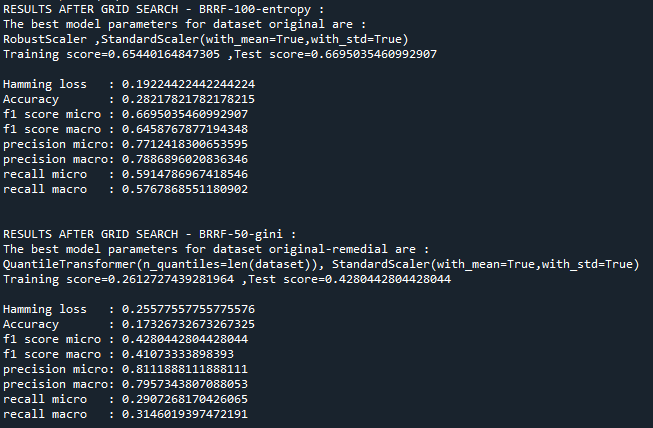
 

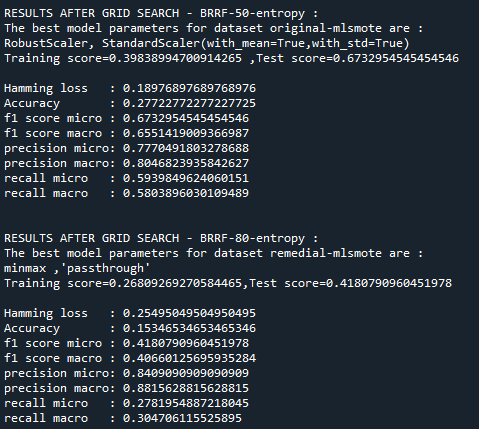
Εικόνα 28. Πληθηκότητα μετά τον REMEDIAL-MLSMOTE. Εικόνα 29. Μετρική “scumble” μετά τον REMEDIAL-MLSMOTE.

# 7. MULTI-LABEL LEARNING: BINARY RELEVANCE

Η μέθοδος Binary Relevance (*“BR”*), αποτελεί την πιο διάσημη μέθοδο μετασχηματισμού δεδομένων με πολλαπλές ετικέτες. Η λειτουργία της βασίζεται στην εκπαίδευση ανεξάρτητων δυαδικών κατηγοριοποιητών για κάθε κατηγορία. Έπειτα, συμψηφίζει τις επιμέρους προβλέψεις των κατηγοριοποιητών. Στα πλαίσια αυτής της εργασίας, η μέθοδος αυτή χρησιμοποιήθηκε με κατηγοριοποιητές Random Forest (*“RF”*).

Αφού γίνει αναζήτηση των βέλτιστων παραμέτρων, με κριτήριο μεγιστοποίησης τη μετρική F1-micro για τα τέσσερα σύνολα δεδομένων, παίρνουμε τα εξής αποτελέσματα :

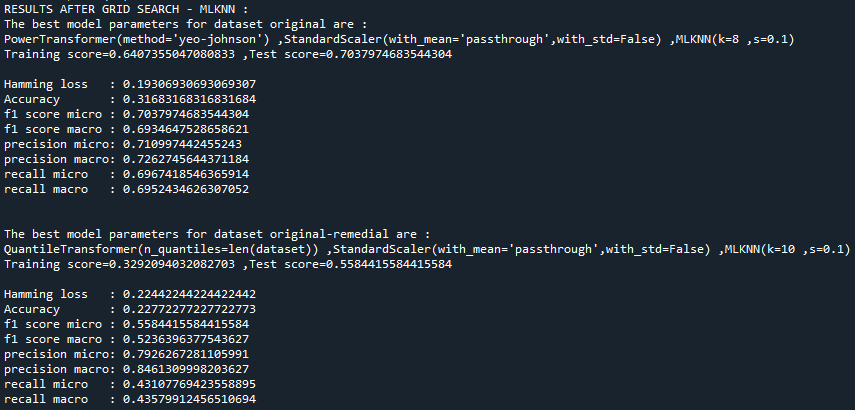


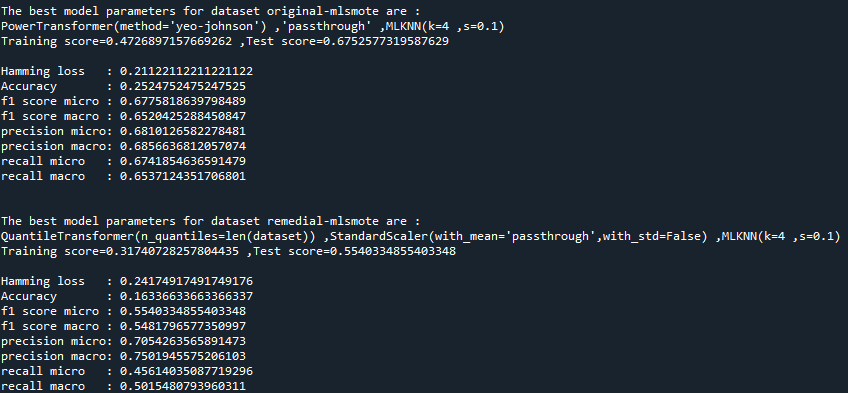


# 8. MULTI-LABEL LEARNING: MULTI-LABEL ΚΝΝ

Ακολούθως, μελετήσαμε την απόδοση των τεσσάρων συνόλων δεδομένων χρησιμοποιώντας τον αλγόριθμο Multi-label K-nearest neighbor *(“MLKNN”*). Ο αλγόριθμος αυτός είναι μια παραλλαγή του κλασικού αλγορίθμου K-nearest neighbor (*“KNN”*), για δεδομένα με πολλαπλές ετικέτες. Παρότι θεωρείται απλός στην υλοποίησή του, καταφέρνει να πετυχαίνει συχνά υψηλές αποδόσεις και έχει το επιπρόσθετο προνόμιο της ερμηνευσιμότητας του.

Αφού γίνει αναζήτηση των βέλτιστων παραμέτρων, με κριτήριο μεγιστοποίησης τη μετρική F1-micro για τα τέσσερα σύνολα δεδομένων, παίρνουμε τα εξής αποτελέσματα :





# 9. ΣΥΜΠΕΡΑΣΜΑΤΑ

## 9.1 Συμπεράσματα αποτελεσμάτων BINARY RELEVANCE

Παρατηρούμε, ότι το σύνολο δεδομένων “original-mlsmote” πετυχαίνει την καλύτερη απόδοση, με τη μετρική F1-micro να διαμορφώνεται στην τιμή 0.6732, έναντι του “original” με τιμή 0.6695, του “original-remedial” με 0.4280 και του “remedial-mlsmote” με 0.4180.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Όνομα συνόλου | original | original-remedial | original-mlsmote | remedial-mlsmote |
| F1-micro | 0.6695 | 0.4280 | 0.6732 | 0.4180 |

Σχετικά με τις υπόλοιπες μετρικές, παρατηρούμε επίσης καλύτερα αποτελέσματα στο σύνολο “original-mlsmote” με μοναδική εξαίρεση τη μετρική “accuracy”, όπου το αρχικό σύνολο δεδομένων έχει τιμή 0.2821 και το “original-mlsmote” έχει τιμή 0.2772.

Συνολικά, το βέλτιστο μοντέλο αποτελείται από τις εξής παραμέτρους :

* RobustScaler()
* StandardScaler(with\_mean=True,with\_std=True)
* BinaryRelevance(classifier=RandomForest(n\_estimators=50), criterion=”entropy”)

|  |  |
| --- | --- |
| Όνομα μετρικής | Τιμή |
| Hamming loss | 0.18976897689768976 |
| Accuracy | 0.27722772277227725 |
| F1-micro | 0.6732954545454546 |
| F1-macro | 0.6551419009366987 |
| Precision micro | 0.7770491803278688 |
| Precision macro | 0.8046823935842627 |
| Recall micro | 0.5939849624060151 |
| Recall macro | 0.5803896030109489 |

## 9.2 Συμπεράσματα αποτελεσμάτων MULTI-LABEL KNN

Και για την περίπτωση όπου ο κατηγοριοποιητής ήταν ο MLKNN, το καλύτερο μοντέλο που δημιουργήσαμε ήταν το “original-mlsmote”.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Όνομα συνόλου | original | original-remedial | original-mlsmote | remedial-mlsmote |
| F1-micro | 0.7037 | 0.5584 | 0.7225 | 0.5540 |

Συνολικά, το βέλτιστο μοντέλο αποτελείται από τις εξής παραμέτρους :

* PowerTransformer(method='yeo-johnson')
* MLKNN(k=9 , s=0.2)

# 10. EXPLAINABLE SYSTEM: LIME

Χρησιμοποιήθηκαν δυο τεχνικές για την ερμηνευσιμότητα των μοντέλων, οι τεχνικές lime και multi-label rule learning. Η μέθοδος lime αναζητά τον λόγο, για τον οποίο ένα μοντέλο πήρε απόφαση για ένα συγκεκριμένο παράδειγμα (τοπική ερμηνευσιμότητα). Από την άλλη μεριά, η μέθοδος multi-label rule learning προσπαθεί να ερμηνεύσει συνολικά το μοντέλο και να βρει τη λογική που το διέπει.

Σε πρώτη φάση, εφαρμόσαμε την μέθοδο Lime. Η μέθοδος Lime υπολογίζει τα χαρακτηριστικά που συνέβαλαν για την κατηγοριοποίηση ενός παραδείγματος σε μια συγκεκριμένη κατηγορία. Πρόκειται για τοπική τεχνική ερμηνευσιμότητας ενός μοντέλου.

Η μέθοδος lime μπορεί να εφαρμοστεί σε όλα τα μοντέλα εκπαίδευσης (model-agnostic) και σε κάθε τύπο δεδομένων (κείμενο, εικόνες, κλπ.). Η λειτουργία του βασίζεται στη δημιουργία ενός καινούριου χώρου καταστάσεων στον οποίο τα χαρακτηριστικά είναι ερμηνεύσιμα. Στη συνέχεια, μεταφέρει το παράδειγμα που θέλουμε να εξετάσουμε στον καινούριο χώρο καταστάσεων. Εκεί, δημιουργεί νέα συνθετικά δεδομένα εφαρμόζοντας μικρές αλλαγές σε ένα ή περισσότερα χαρακτηριστικά στις τιμές των χαρακτηριστικών του παραδείγματος που εξετάζουμε. Για τα νέα συνθετικά δεδομένα, υπολογίζει τις προβλέψεις τους από το μοντέλο (στον αρχικό χώρο καταστάσεων). Τελικά, συνδυάζοντας τα συνθετικά δεδομένα και τις προβλέψεις του μοντέλου εκπαιδεύεται τοπικά ένα γραμμικό μοντέλο. Με κατάλληλη παραμετροποίηση (πχ. L1 regularization) μπορεί να μειωθεί ο αριθμός των χαρακτηριστικών που λαμβάνονται υπόψιν. Στα συνθετικά δεδομένα γίνεται στάθμιση ανάλογα με την απόστασή τους από το παράδειγμα στο νέο χώρο καταστάσεων.

Εφαρμόσαμε την τεχνική lime στο μοντέλο με τα καλύτερα αποτελέσματα στην μετρική f1-score για κάποια τυχαία παραδείγματα από το test dataset. Καθώς, η μέθοδος lime δεν έχει αναπτυχθεί ακόμη για εφαρμογή σε multi-label δεδομένα, θεωρήσαμε το multi-label πρόβλημα ως multi-class πρόβλημα. Στον αλγόριθμο για τον υπολογισμό του lime χρησιμοποιήθηκαν οι πιθανότητες που έβγαλαν τα μοντέλα εκπαίδευσης αφού κανονικοποιήθηκαν ώστε να αθροίζουν στην μονάδα. Με τη συγκεκριμένη παραλλαγή, το lime υπολογίζει τα σημαντικότερα χαρακτηριστικά μόνο για το label με την μεγαλύτερη πιθανότητα. Ακολουθούν τα αποτελέσματα για κάποια τυχαία παραδείγματα.

Παράδειγμα #1

A screenshot of a cell phone

Description automatically generatedA picture containing orange

Description automatically generated

Το μοντέλο μας πρόβλεψε για το παράδειγμα τις τιμές 1 για labels 1 (“happy-pleased”) και 2 (“relaxing-calm”). Η πιθανότητα του label 2 ήταν μεγαλύτερη και επομένως το lime υπολόγισε τα σημαντικότερα χαρακτηριστικά μόνο για το label 2 (‘relaxing-clam’).

Στον πίνακα που ακολουθεί βλέπουμε σε τι ποσοστό επηρέασαν τα συγκεκριμένα χαρακτηριστικά ώστε να κατηγοριοποιηθεί το παράδειγμα με το label ”relaxing-calm”. Για κάθε χαρακτηριστικό που το lime το έλαβε υπόψιν αναφέρεται η τιμή του καθώς και οι συνθήκες για τις οποίες το χαρακτηριστικό επηρεάζει θετικά ή αρνητικά την συγκεκριμένη κατηγοριοποίηση.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | Χαρακτηριστικά | Τιμή | Τιμή απόφασης | Επίδραση | Ποσοστό |
| F14 | Mean\_Acc1298\_Mean\_Mem40\_MFCC\_11 | 1.58 | > 0.84 | Θετική | 0.04 |
| F5 | Mean\_Acc1298\_Mean\_Mem40\_MFCC\_2 | 1.26 | > 0.56 | Αρνητική | 0.04 |
| F7 | Mean\_Acc1298\_Mean\_Mem40\_MFCC\_4 | 0.98 | > 0.67 | Αρνητική | 0.03 |
| F58 | Std\_Acc1298\_Std\_Mem40\_MFCC\_7 | 1.67 | > 0.82 | Θετική | 0.03 |
| F31 | Mean\_Acc1298\_Std\_Mem40\_MFCC\_12 | 1.05 | > 0.84 | Αρνητική | 0.02 |
| F12 | Mean\_Acc1298\_Mean\_Mem40\_MFCC\_9 | 0.55 | > 0.06 &  <= 0.80 | Θετική | 0.02 |
| F63 | Std\_Acc1298\_Std\_Mem40\_MFCC\_12 | 1.76 | > 0.62 | Αρνητική | 0.02 |
| F57 | Std\_Acc1298\_Std\_Mem40\_MFCC\_6 | 0.52 | > 0.11 &  <= 0.83 | Θετική | 0.02 |
| F9 | Mean\_Acc1298\_Mean\_Mem40\_MFCC\_6 | 0.31 | > 0.00 &  <= 0.80 | Θετική | 0.02 |
| F47 | Std\_Acc1298\_Mean\_Mem40\_MFCC\_12 | 0.89 | > 0.72 | Αρνητική | 0.01 |

Παράδειγμα #2

A screenshot of a cell phone

Description automatically generatedA screenshot of a cell phone

Description automatically generated

Το μοντέλο μας πρόβλεψε για το παράδειγμα τις τιμές 1 για labels 1 (“happy-pleased”) και 0 (“amazed-suprised'”). Η πιθανότητα του label 1 ήταν μεγαλύτερη και επομένως το lime υπολόγισε τα σημαντικότερα χαρακτηριστικά μόνο για το label 1 (“happy-pleased”).

Στον πίνακα που ακολουθεί βλέπουμε σε τι ποσοστό επηρέασαν τα συγκεκριμένα χαρακτηριστικά ώστε να κατηγοριοποιηθεί το παράδειγμα με το label “happy-pleased”. Για κάθε χαρακτηριστικό που το lime το έλαβε υπόψιν αναφέρεται η τιμή του καθώς και οι συνθήκες για τις οποίες το χαρακτηριστικό επηρεάζει θετικά ή αρνητικά την συγκεκριμένη κατηγοριοποίηση.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | Χαρακτηριστικά | Τιμή | Τιμή απόφασης | Επίδραση | Ποσοστό |
| F31 | Mean\_Acc1298\_Std\_Mem40\_MFCC\_12 | 0.53 | > 0.47 &  <= 0.84 | Αρνητική | 0.02 |
| F40 | Std\_Acc1298\_Mean\_Mem40\_MFCC\_5 | 0.21 | > 0.10 &  <= 0.80 | Αρνητική | 0.02 |
| F8 | Mean\_Acc1298\_Mean\_Mem40\_MFCC\_5 | 0.81 | > 0.80 | Θετική | 0.02 |
| F16 | Mean\_Acc1298\_Std\_Mem40\_Centroid | 0.57 | > -0.09 &  <= 0.82 | Θετική | 0.02 |
| F38 | Std\_Acc1298\_Mean\_Mem40\_MFCC\_3 | 0.97 | > 0.81 | Θετική | 0.01 |
| F58 | Std\_Acc1298\_Std\_Mem40\_MFCC\_7 | 0.6 | > 0.15  <= 0.82 | Θετική | 0.01 |
| F44 | Std\_Acc1298\_Mean\_Mem40\_MFCC\_9 | -0.14 | > -0.97 &  <= 0.07 | Αρνητική | 0.01 |
| F9 | Mean\_Acc1298\_Mean\_Mem40\_MFCC\_6 | 0.75 | > 0.02 &  <= 0.82 | Θετική | 0.01 |
| F5 | Mean\_Acc1298\_Mean\_Mem40\_MFCC\_2 | -1.29 | <= -.53 | Θετική | 0.01 |
| F57 | Std\_Acc1298\_Std\_Mem40\_MFCC\_6 | 0.76 | > 0.10  <= 0.84 | Θετική | 0.01 |

# 11. EXPLAINABLE SYSTEM: MULTI-LABEL RULE LEARNING

Στη συνέχεια, χρησιμοποιήθηκε η μέθοδος multi-label rule learning για την καθολική ερμηνευσιμότητα του μοντέλου μας. Στη συγκεκριμένη περίπτωση, η μέθοδος εκπαιδεύει ένα μοντέλο μηχανικής μάθησης (μάθηση κανόνων) πάνω στις προβλέψεις ενός μοντέλου. Η μέθοδος μπορεί να εφαρμοστεί ανεξάρτητα από το μοντέλο εκπαίδευσης που χρησιμοποιήθηκε. Η λειτουργία του βασίζεται στη δημιουργία-εκπαίδευση ενός μοντέλου μάθησης κανόνων που προσπαθεί να προσεγγίσει το μοντέλο εκπαίδευσης. Για την εκπαίδευση του νέου μοντέλου χρησιμοποιείται ως είσοδος οι προβλέψεις του μοντέλου μας. Η αξιολόγηση γίνεται ανάλογα με το βαθμό που οι προβλέψεις του νέου μοντέλου συμφωνούν με τις προβλέψεις του μοντέλου μας, δηλαδή πόσο καλά μιμείται το μοντέλο μας.

Για την εφαρμογή του της μεθόδου multi-label rule learning χρησιμοποιήθηκε η εφαρμογή jar από το online repository <https://github.com/keelm/SeCo-MLC> το οποίο περιλαμβάνει έναν separate-and-conquer αλγόριθμο μάθησης κανόνων για μάθηση πάνω σε multi-label δεδομένα. Η εκμάθηση κανόνων έγινε με τις παρακάτω επιλογές:

* predictZeroRules=False: δεν δημιουργεί κανόνες για το πότε ένα label παίρνει μηδενική τιμή
* skipThresholdPercentage: το όριο για το οποίο για το οποίο θα δημιουργηθούν stopping rules.

Η τεχνική εφαρμόστηκε στο μοντέλο με τα καλύτερα αποτελέσματα της μετρικής f1-score. Στον παρακάτω πίνακα αναγράφονται τα αποτελέσματα:

|  |  |
| --- | --- |
| Αποτελέσματα | |
| Αριθμός κανόνων | 91 |
| Μέσο μήκος κανόνων | 1.2307692307692308 |
| Κανόνες που περιλαμβάνουν περισσότερες από μια ετικέτες  (multiHeadRules) | 44 |
| Μέσος αριθμός ετικετών | 1.5494505494505495 |
| Μέσος αριθμός ετικετών ανά multiHeadRules | 2.1363636363636362 |
| Αριθμός παραδειγμάτων που δεν καλύπτονται από τους κανόνες | 38 |
| Ποσοστό κάλυψης | 90.28% |

Ακολουθούν κάποιοι ενδεικτικοί κανόνες από την εφαρμογή της συγκεκριμένης τεχνικής.

1. [relaxing-calm = 1, quiet-still = 1] :-
   1. Std\_Acc1298\_Mean\_Mem40\_MFCC\_12 >= 0.572123,
   2. Mean\_Acc1298\_Mean\_Mem40\_MFCC\_10 <= 0.294459
2. [quiet-still = 1, sad-lonely = 1] :-
   1. Mean\_Acc1298\_Mean\_Mem40\_Rolloff <= 0.0551115
   2. Mean\_Acc1298\_Mean\_Mem40\_Flux <= 0.084511
3. [angry-aggresive = 1] :-
   1. Mean\_Acc1298\_Std\_Mem40\_MFCC\_3 >= 1.222218.
4. [amazed-suprised = 1, angry-aggresive = 1]
   1. Std\_Acc1298\_Std\_Mem40\_MFCC\_12 <= 0.0416915,
   2. Mean\_Acc1298\_Std\_Mem40\_MFCC\_10 <= 0.4246205

# ΒΙΒΛΙΟΓΡΑΦΙΑ

[1] G. Tsoumakas, E. Spyromitros-Xioufis, J. Vilcek and I. Vlahavas, "Mulan: A Java Library for Multi-Label Learning," The Journal of Machine Learning Research, vol. 12, pp. 2411-2414, 2011.

[2] Tzanetakis G, Cook P: Musical genre classification of audio signals. *IEEE Trans Speech Audio Process* 2002,10(5):293-302. 10.1109/TSA.2002.800560

[3] F. Charte, A. Rivera, M. J. del Jesus, F. Herrera, Resampling Multilabel Datasets by Decoupling Highly Imbalanced Labels, in: Hybrid Artificial Intelligent Systems, Vol. 9121 of Lecture Notes in Computer Science, Springer International Publishing, 2015, pp. 489–501. doi:10.1007/978-3-319-19644-2 41.

[4] F. Charte, A. J. Rivera, M. J. del Jesus, F. Herrera, MLSMOTE: Approaching imbalanced multilabel learning through synthetic instance generation, Knowledge-Based Systems 89 (2015) 385–397. doi:10.1016/j.knosys.2015.07.019.

[5] S. Godbole, S. Sarawagi, Discriminative Methods for MultiLabeled Classification, in: Advances in Knowl. Discovery and Data Mining, Vol. 3056, 2004, pp. 22–30. doi:10.1007/978-3- 540-24775-3 5

[6] Zhang, M.L.; Zhou, Z.H. (2007). "ML-KNN: A lazy learning approach to multi-label learning". Pattern Recognition. 40 (7): 2038–2048.  [doi](https://en.wikipedia.org/wiki/Doi_(identifier)):[10.1016/j.patcog.2006.12.019](https://doi.org/10.1016%2Fj.patcog.2006.12.019)

[7] Ribeiro, Marco Tulio, Sameer Singh, and Carlos Guestrin. 2016. “Why Should I Trust You?: Explaining the Predictions of Any Classifier.” In, 1135–44. ACM Press. https://doi.org/10.1145/2939672.2939778.

[8] Eneldo Loza Mencía, Johannes Fürnkranz, Eyke Hüllermeier, Michael Rapp. “Learning Interpretable Rules for Multi-label Classification”.

[9] N.V. Chawla, K.W. Bowyer, L.O. Hall, W.P. Kegelmeyer, SMOTE: syntheticminority over-sampling technique, J. Artif. Intell. Res. 16 (2002) 321–357, <http://dx.doi.org/10.1613/jair.953>.

# ΠΑΡΑΡΤΗΜΑ – ΟΝΟΜΑΤΑ ΧΑΡΑΚΤΗΡΙΣΤΙΚΩΝ

|  |  |
| --- | --- |
| F0 | 'Mean\_Acc1298\_Mean\_Mem40\_Centroid' |
| F1 | 'Mean\_Acc1298\_Mean\_Mem40\_Rolloff' |
| F2 | 'Mean\_Acc1298\_Mean\_Mem40\_Flux' |
| F3 | 'Mean\_Acc1298\_Mean\_Mem40\_MFCC\_0' |
| F4 | ‘Mean\_Acc1298\_Mean\_Mem40\_MFCC\_1' |
| F5 | 'Mean\_Acc1298\_Mean\_Mem40\_MFCC\_2' |
| F6 | ‘Mean\_Acc1298\_Mean\_Mem40\_MFCC\_3' |
| F7 | 'Mean\_Acc1298\_Mean\_Mem40\_MFCC\_4' |
| F8 | ‘Mean\_Acc1298\_Mean\_Mem40\_MFCC\_5' |
| F9 | 'Mean\_Acc1298\_Mean\_Mem40\_MFCC\_6' |
| F10 | ‘Mean\_Acc1298\_Mean\_Mem40\_MFCC\_7' |
| F11 | 'Mean\_Acc1298\_Mean\_Mem40\_MFCC\_8' |
| F12 | ‘Mean\_Acc1298\_Mean\_Mem40\_MFCC\_9' |
| F13 | 'Mean\_Acc1298\_Mean\_Mem40\_MFCC\_10' |
| F14 | ‘Mean\_Acc1298\_Mean\_Mem40\_MFCC\_11' |
| F15 | 'Mean\_Acc1298\_Mean\_Mem40\_MFCC\_12' |
| F16 | ‘Mean\_Acc1298\_Std\_Mem40\_Centroid' |
| F17 | 'Mean\_Acc1298\_Std\_Mem40\_Rolloff' |
| F18 | ‘Mean\_Acc1298\_Std\_Mem40\_Flux' |
| F19 | 'Mean\_Acc1298\_Std\_Mem40\_MFCC\_0' |
| F20 | ‘Mean\_Acc1298\_Std\_Mem40\_MFCC\_1' |
| F21 | 'Mean\_Acc1298\_Std\_Mem40\_MFCC\_2' |
| F22 | ‘Mean\_Acc1298\_Std\_Mem40\_MFCC\_3' |
| F23 | 'Mean\_Acc1298\_Std\_Mem40\_MFCC\_4' |
| F24 | ‘Mean\_Acc1298\_Std\_Mem40\_MFCC\_5' |
| F25 | 'Mean\_Acc1298\_Std\_Mem40\_MFCC\_6' |
| F26 | ‘Mean\_Acc1298\_Std\_Mem40\_MFCC\_7' |
| F27 | 'Mean\_Acc1298\_Std\_Mem40\_MFCC\_8' |
| F28 | ‘Mean\_Acc1298\_Std\_Mem40\_MFCC\_9' |
| F29 | 'Mean\_Acc1298\_Std\_Mem40\_MFCC\_10', |
| F30 | ‘Mean\_Acc1298\_Std\_Mem40\_MFCC\_11' |
| F31 | 'Mean\_Acc1298\_Std\_Mem40\_MFCC\_12' |
| F32 | ‘Std\_Acc1298\_Mean\_Mem40\_Centroid' |
| F33 | 'Std\_Acc1298\_Mean\_Mem40\_Rolloff' |
| F34 | ‘Std\_Acc1298\_Mean\_Mem40\_Flux' |
| F35 | 'Std\_Acc1298\_Mean\_Mem40\_MFCC\_0' |
| F36 | ‘Std\_Acc1298\_Mean\_Mem40\_MFCC\_1' |
| F37 | 'Std\_Acc1298\_Mean\_Mem40\_MFCC\_2' |
| F38 | ‘Std\_Acc1298\_Mean\_Mem40\_MFCC\_3' |
| F39 | 'Std\_Acc1298\_Mean\_Mem40\_MFCC\_4' |
| F40 | ‘Std\_Acc1298\_Mean\_Mem40\_MFCC\_5' |
| F41 | 'Std\_Acc1298\_Mean\_Mem40\_MFCC\_6', |
| F42 | ‘Std\_Acc1298\_Mean\_Mem40\_MFCC\_7' |
| F43 | 'Std\_Acc1298\_Mean\_Mem40\_MFCC\_8' |
| F44 | ‘Std\_Acc1298\_Mean\_Mem40\_MFCC\_9' |
| F45 | 'Std\_Acc1298\_Mean\_Mem40\_MFCC\_10', |
| F46 | ‘Std\_Acc1298\_Mean\_Mem40\_MFCC\_11' |
| F47 | 'Std\_Acc1298\_Mean\_Mem40\_MFCC\_12' |
| F48 | ‘Std\_Acc1298\_Std\_Mem40\_Centroid' |
| F49 | 'Std\_Acc1298\_Std\_Mem40\_Rolloff' |
| F50 | ‘Std\_Acc1298\_Std\_Mem40\_Flux' |
| F51 | 'Std\_Acc1298\_Std\_Mem40\_MFCC\_0', |
| F52 | ‘Std\_Acc1298\_Std\_Mem40\_MFCC\_1' |
| F53 | 'Std\_Acc1298\_Std\_Mem40\_MFCC\_2' |
| F54 | ‘Std\_Acc1298\_Std\_Mem40\_MFCC\_3' |
| F55 | 'Std\_Acc1298\_Std\_Mem40\_MFCC\_4' |
| F56 | ‘Std\_Acc1298\_Std\_Mem40\_MFCC\_5' |
| F57 | 'Std\_Acc1298\_Std\_Mem40\_MFCC\_6' |
| F58 | ‘Std\_Acc1298\_Std\_Mem40\_MFCC\_7' |
| F59 | 'Std\_Acc1298\_Std\_Mem40\_MFCC\_8' |
| F60 | ‘Std\_Acc1298\_Std\_Mem40\_MFCC\_9' |
| F61 | 'Std\_Acc1298\_Std\_Mem40\_MFCC\_10' |
| F62 | ‘Std\_Acc1298\_Std\_Mem40\_MFCC\_11' |
| F63 | 'Std\_Acc1298\_Std\_Mem40\_MFCC\_12' |
| F64 | ‘BH\_LowPeakAmp' |
| F65 | 'BH\_LowPeakBPM' |
| F66 | 'BH\_HighPeakAmp' |
| F67 | 'BH\_HighPeakBPM' |
| F68 | ‘BH\_HighLowRatio' |
| F69 | 'BHSUM1' |
| F70 | 'BHSUM2' |
| F71 | 'BHSUM3' |