

ΙΟΝΙΟ ΠΑΝΕΠΙΣΤΗΜΙΟ

ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ



Πτυχιακή Εργασία

ΣΥΣΤΗΜΑ ΣΤΛΛΟΓΗΣ ΚΑΙ ΕΠΑΝΑΔΙΑΘΕΣΗΣ
ΔΕΔΟΜΕΝΩΝ ΔΗΜΟΤΙΚΟΤΗΤΑΣ ΜΟΥΣΙΚΩΝ ΚΟΜΜΑΤΙΩΝ
ΚΑΙ ΠΡΟΒΛΕΨΗ ΤΟΤΣ ΜΕΣΩ ΑΚΟΥΣΤΙΚΩΝ ΧΑΡΑΚΤΗΡΙΣΤΙΚΩΝ

ΘΕΟΔΩΡΟΠΟΥΛΟΣ ΣΠΥΡΙΔΩΝ

Επιβλέπων: – ΙΩΑΝΝΗΣ ΚΑΡΥΔΗΣ –

15 Ιουνίου 2021

Επιβλέπων

ΙΩΑΝΝΗΣ ΚΑΡΥΔΗΣ, *Επίκουρος καθηγητής,*
Ιόνιο Πανεπιστήμιο

Τριμελής Επιτροπή

ΙΩΑΝΝΗΣ ΚΑΡΥΔΗΣ, *Επίκουρος καθηγητής,*
Ιόνιο Πανεπιστήμιο

ΚΑΤΙΑ-ΛΗΔΑ ΚΕΡΜΑΝΙΔΟΥ, *Αναπληρώτρια καθηγήτρια,*
Ιόνιο Πανεπιστήμιο

ΑΓΓΕΛΙΚΗ ΤΣΩΧΟΥ, *Επίκουρη καθηγήτρια,*
Ιόνιο Πανεπιστήμιο

Περίληψη

Στην παρούσα εργασία ασχοληθήκαμε με την δημιουργία ενός συστήματος συλλογής και επαναδιάθεσης δεδομένων δημοτικότητας μουσικών κομματιών και στην συνέχεια με την πρόβλεψη των δημοφιλών κομματιών. Η εργασία χωρίστηκε σε δύο μέρη. Στο πρώτο μέρος μελετήσαμε διάφορες πηγές λιστών δημοφιλών μουσικών κομματιών(λίστες επιτυχιών) ή όπως συχνά αναφέρονται στα αγγλικά "Music charts". Καταλήξαμε σε τρεις πηγές από τις οποίες αντλήσαμε δεδομένα τα οποία αποθηκεύτηκαν σε μια απλή βάση δεδομένων. Συγκεκριμένα αντλήθηκαν δεδομένα δημοτικότητας για 2.844 λίστες , 22.600 μουσικά κομμάτια και 8.329 καλλιτέχνες. Συνολικά η βάση περιείχε 337.981 εγγραφές και δεδομένα που χρονολογούνται από το 1958 μέχρι σήμερα. Στην συνέχεια δημιουργήθηκε μια ιστοσελίδα η οποία εκτός από διάφορες πληροφορίες για το σύστημα παρείχε και μερικές διαγραμματικές απεικονίσεις των δεδομένων που αποθηκεύσαμε προηγουμένως στην βάση δεδομένων. Μέσω των διαγραμματικών απεικονίσεων ο χρήστης μπορεί να πάρει μια πολύ καλή γύση των δεδομένων από μια διαφορετική οπτική σε σχέση με το πως παρουσιάζονται στις αρχικές πηγές. Στο δεύτερο μέρος της εργασίας έγινε μια προσπάθεια να προβλέψουμε τα δεδομένα χρησιμοποιώντας διάφορες τεχνικές μηχανικής μάθησης. Συγκεκριμένα αφού χρησιμοποιήσαμε την βάση δεδομένων που κατασκευάσαμε στο πρώτο μέρος δημιουργήσαμε αρχικά ένα σετ δεδομένων που περιείχε 17425 διάσημα κομμάτια μαζί με τα μουσικά χαρακτηριστικά που παράγει το spotify για καθένα από αυτά. Έπειτα κατασκευάσαμε άλλο ένα σετ δεδομένων με 17315 μη-διάσημα μουσικά κομμάτια, δηλαδή κομμάτια που δεν υπήρξαν ποτέ σε κάποια λίστα επιτυχιών, και έπειτα προσθέσαμε και σε αυτά τα μουσικά χαρακτηριστικά που παράγει το spotify για καθένα από αυτά. Ενώνοντας τα δύο σετ δεδομένων καταλήξαμε σε ένα σετ δεδομένων 34740 γραμμών και 17 στηλών. Έπειτα αφού καθαρίσαμε, αναλύσαμε και κανονικοποιήσαμε το τελικό σετ δεδομένων προχωρήσαμε στην πρόβλεψη των δεδομένων. Κατά την ανάλυση παρατηρήσαμε μια σχέση της ποσότητας των φωνητικών σε ένα κομμάτι αλλά και της έντασης του με τον δείκτη που δήλωνε αν το κομμάτι ήταν διάσημο ή όχι. Έπειτα κατά την πρόβλεψη των δεδο-

μένων το μεγαλύτερο ποσοστό επιτυχίας ήταν 78.71%. Το αποτέλεσμα αυτό επετεύχθη με χρήση της τεχνικής min - max για κανονικοποίηση, του αλγόριθμου Νευρωνικών δικτύων για την δημιουργία του μοντέλου πρόβλεψης και της τεχνικής k-folds cross validation για την αξιολόγηση του μοντέλου.

Extended synopsis

In this work, we attempt to solve the "Hit Song Science" problem which aims to the prediction of commercially successful songs before their distribution using their audio features and characteristics. Predicting hit songs would have a huge impact on the development of the music industry. Artists would be able to focus their efforts on more promising and maybe even better songs and therefore increase their profits. Also further analyzing data from popularity sources would lead to a better understanding of the history and trends of music. Several works in the past have tried to address the issue. Such works exist since 2005. Our first part of the work was based on a previous work that was published in 2016. In the first part of this work, we constructed a database consisting of 2844 Music charts, 22600 hit songs, and 8329 artists, dating back to 1985 till today. Three popularity sources were used to collect the data namely billboard, Spotify, and shazam. Also, an automated data collection system was constructed. Attention was given to its error-proofing and speed in an attempt to make a fully automated weekly update system for our database. Later a website was made to present our work. The database was used on our website to provide a user-friendly way to explore the data through various types of diagrams. For example, a user can explore the presence an artist had on music charts or explore all of his songs that made it to the charts. In the second part of this work, we constructed a dataset in an attempt to solve the Hit Song Science problem. First using the database we mentioned earlier we constructed a dataset with approximately 17430 hit songs. Later we constructed a second dataset with approximately 17315 nonhit songs using a third party dataset. We then combined the two datasets and extracted the audio features for every track using the Spotify API. The column "On_chart" was also added to indicate if the song is a hit or not. The final dataset contained approximately 34740 tracks and 17 features for each of the songs. To predict whether a song will be a hit or not we tested four models on our dataset. Before testing the models we analyzed the dataset to understand

our data and check for missing or invalid values that may existed. The analysis of the dataset showed a strong influence of the "instrumentalness" and "valence" features on the "On_chart" indicator. This led us to the assumption that the amount of vocals in the song and the musical positiveness conveyed by the song may have a serious impact in defining a song as hit or non-hit. Then we normalized our data using the min-max normalization technique to make it more suitable for testing. The four models we used were:

- Support Vector Machine
- Neural Network
- Random Forest
- k-NN

The best model was Neural Network, which was able to predict hit songs with 78.71% accuracy. The four models had the following results.

Accuracy			
k-NN	SVM	Neural Networks	Random forest
75.68	75.43	78.71	76.03

It should also be considered that according to the confusion matrix we constructed later, Random Forest algorithm had fewer False Positive classifications than any other algorithm. Finally, we believe that in the future more chart sources should be considered. Also, further attention should be given to "instrumentalness" and "valence" audio features which seemed to have a strong influence on the class.

Πρόλογος και Ευχαριστίες

Η παρούσα πτυχιακή εργασία με τίτλο "Σύστημα συλλογής και επαναδιάθεσης δεδομένων δημοτικότητας μουσικών κομματιών, και πρόβλεψη τους μέσω ακουστικών χαρακτηριστικών" εκπονήθηκε στα πλαίσια της εκπλήρωσης των προϋποθέσεων για την απόκτηση πτυχίου από το τμήμα πληροφορικής του Ιονίου Πανεπιστημίου με έδρα την πόλη της Κέρκυρας. Το πρακτικό κομμάτι της εργασίας ξεκίνησε στα μέσα του 2019 και ολοκληρώθηκε στα τέλη του 2020 με υπεύθυνο τον καθηγητή Ιωάννη Καρύδη. Σκοπός της εργασίας ήταν αρχικά η δημιουργία μιας βάσης δεδομένων και ενός ολοκληρωμένου πληροφοριακού συστήματος, συλλογής και επαναδιάθεσης δεδομένων δημοτικότητας μουσικών κομματιών. Έπειτα πραγματοποιήθηκε μελέτη και πειραματισμός πάνω στο θέμα της πρόβλεψης των δεδομένων. Θα ήθελα να ευχαριστήσω θερμά τον καθηγητή μου κύριο Ιωάννη Καρύδη για την καθοδήγηση και τις πολύτιμες συμβουλές που μου προσέφερε κατά την τέλεση της εργασίας. Επίσης θα ήθελα να εκφράσω τις ευχαριστίες μου προς τους διδάσκοντες και τους συμφοιτητές μου στο Ιόνιο Πανεπιστήμιο με τους οποίους συμπορεύτηκα. Τέλος θα ήθελα να ευχαριστώ την οικογένεια μου για την στήριξη που μου παρείχαν κατά την διάρκεια των σπουδών μου.

Κέρκυρα, 15 Ιουνίου 2021

ΘΕΟΔΩΡΟΠΟΥΛΟΣ ΣΠΥΡΟΣ

Περιεχόμενα

A'	Εισαγωγή	1
A'.1	ΤΟ ΠΡΟΒΛΗΜΑ ΚΑΙ ΤΑ ΚΙΝΗΤΡΑ ΕΠΙΛΕΞΗΣ ΤΟΥ	2
A'.2	ΣΥΝΕΙΣΦΟΡΑ	2
A'.3	ΣΥΝΟΠΤΙΚΗ ΠΑΡΟΥΣΙΑΣΗ ΚΕΦΑΛΑΙΩΝ.	3
B'	Σχετικές έρευνες.	4
B'.1	ΠΡΟΪΠΑΡΧΟΥΣΕΣ ΕΡΕΥΝΕΣ	4
B'.2	MIR TRACK POPULARITY PREDICTION	5
Γ'	Αποθήκευση και επαναδιάθεση των δεδομένων	7
Γ'.1	ΠΗΓΕΣ	8
Γ'.2	ΣΧΕΔΙΑΣΜΟΣ ΚΑΙ ΜΟΡΦΗ	9
Γ'.3	ΑΡΧΙΚΗ ΣΥΛΛΟΓΗ ΔΕΔΟΜΕΝΩΝ	12
Γ'.4	ΑΝΑΝΕΩΣΗ ΚΑΙ ΣΥΝΤΗΡΗΣΗ	13
Γ'.5	ΠΕΡΙΕΧΟΜΕΝΟ	13
Γ'.6	ΕΠΑΝΑΔΙΑΘΕΣΗ ΔΕΔΟΜΕΝΩΝ	14
Γ'.7	ΜΕΛΛΟΝΤΙΚΗ ΧΡΗΣΗ ΣΥΣΤΗΜΑΤΟΣ	15
Δ'	Ανάλυση και πρόβλεψη των δεδομένων	19
Δ'.1	PYTHON ΚΑΙ JUPYTER NOTEBOOK	20
Δ'.2	NUMPY, PANDAS ΚΑΙ SCIKIT-LEARN	20
Δ'.3	ΑΡΧΙΚΟ ΣΕΤ ΔΕΔΟΜΕΝΩΝ.	22
Δ'.4	SPOTIFY AUDIO FEATURES.	23
Δ'.5	ΣΕΤ ΔΕΔΟΜΕΝΩΝ	24

Περιεχόμενα	vii
Δ'.6 ΚΑΘΑΡΙΣΜΟΣ ΚΑΙ ΑΝΑΛΥΣΗ ΔΕΔΟΜΕΝΩΝ	26
Δ'.7 ΚΑΘΑΡΙΣΜΟΣ ΔΕΔΟΜΕΝΩΝ	27
Δ'.8 ΑΝΑΛΥΣΗ ΔΕΔΟΜΕΝΩΝ	27
Δ'.9 ΕΠΕΞΕΡΓΑΣΙΑ / ΜΕΤΑΣΧΗΜΑΤΙΣΜΟΣ ΤΩΝ ΔΕΔΟΜΕΝΩΝ	31
Δ'.10 ΠΡΟΒΛΕΨΗ ΤΩΝ ΔΕΔΟΜΕΝΩΝ	33
Δ'.11 ΑΠΟΤΕΛΕΣΜΑΤΑ	38
Ε' Συμπεράσματα και μελλοντική έρευνα	42
Ε'.1 ΣΥΜΠΕΡΑΣΜΑΤΑ ΚΑΙ ΠΑΡΑΤΗΡΗΣΕΙΣ	42
Ε'.2 ΜΕΛΛΟΝΤΙΚΗ ΕΡΕΥΝΑ	43
Βιβλιογραφία	45

Κατάλογος Σχημάτων

B'.1 dataset schema (karydis,2016)	5
Γ'.2 Διάγραμμα οντοτήτων - συσχετίσεων	9
Γ'.3 Σχεσιακό διάγραμμα	10
Γ'.4 Αρχική σελίδα και καρτέλες.	14
Γ'.5 Διάγραμμα δημοτικότητας κομματιού.	15
Γ'.6 Αναζήτηση μεμονωμένου καλλιτέχνη.	16
Γ'.7 Αναζήτηση συνεργασιών.	16
Γ'.8 Αποτέλεσμα αναζήτησης για τα κομμάτια του καλλιτέχνη.	17
Γ'.9 Αναζήτηση καλλιτέχνη.	17
Γ'.10 Αποτέλεσμα αναζήτησης για τον καλλιτέχνη.	17
Γ'.11 Αποτέλεσμα αναζήτησης λίστας δημοτικότητας.	18
Γ'.12 Φόρμα Επικοινωνίας.	18
Δ'.13Ιστογράμματα των στηλών του σετ δεδομένων, Μέρος 1ο	27
Δ'.14Ιστογράμματα των στηλών του σετ δεδομένων, Μέρος 2ο.	28
Δ'.15Ιστογράμματα των στηλών του σετ δεδομένων, Μέρος 3ο.	28
Δ'.16Ιστογράμματα των στηλών του σετ δεδομένων, Μέρος 4ο.	29
Δ'.17Ιστογράμματα των στηλών του σετ δεδομένων, Μέρος 5ο.	29
Δ'.18Ιστογράμματα των στηλών του σετ δεδομένων, Μέρος 6ο.	29
Δ'.19Ιστογράμματα των στηλών του σετ δεδομένων, Μέρος 7ο.	29
Δ'.20Correlation heatmap.	30
Δ'.21Οι πιο συχνά εμφανιζόμενοι καλλιτέχνες.	30
Δ'.22SVM	34
Δ'.23k-NN	34
Δ'.24Decision trees.	36

Δ'.25 Παράδειγμα Decision tree.	36
Δ'.26 Πίνακας σύγχυσης «Confusion matrix».	39
Δ'.27 Πίνακας σύγχυσης «Knn».	40
Δ'.28 Πίνακας σύγχυσης «Svm».	40
Δ'.29 Πίνακας σύγχυσης «Neural network».	41
Δ'.30 Πίνακας σύγχυσης «Random forest».	41

Κατάλογος Πινάκων

Κεφάλαιο Α΄

Εισαγωγή

Οι λίστες δημοφιλών μουσικών κομματιών ή λίστες επιτυχιών ή όπως αναφέρονται στα αγγλικά "Music charts" είναι τα πιο διάσημα μουσικά κομμάτια για ένα συγκεκριμένο χρονικό διάστημα, ταξινομημένα με βάση την δημοτικότητα που είχαν σε αυτό το διάστημα. Συνήθως αυτές οι λίστες εκτός από το χρονικό διάστημα περιλαμβάνουν περισσότερα κριτήρια όπως είναι κάποιο συγκεκριμένο είδος μουσικής ή κάποιο συγκεκριμένο γεωγραφικό εύρος. Πάντα οι λίστες έχουν κάποιο όριο ως προς το πόσα μουσικά κομμάτια θα περιλαμβάνουν. Οι πιο διάσημες λίστες έχουν όριο τα σαράντα(40) κομμάτια αλλά υπάρχουν και λίστες με δέκα(10) εκατό(100) ή ακόμα και διακόσια(200) κομμάτια (Record chart, 2021). Αν για παράδειγμα μια λίστα έχει όριο τα εκατό(100) κομμάτια αυτό σημαίνει ότι περιλαμβάνει τα εκατό πιο διάσημα κομμάτια για το χρονικό διάστημα και τα κριτήρια στα οποία αναφέρεται και ούτω καθεξής. Την πρώτη θέση καταλαμβάνει πάντα το πιο διάσημο κομμάτι από όλα, ενώ την τελευταία το λιγότερο διάσημο από όλα, που σημαίνει ότι όσο μεγαλύτερη αριθμητικά είναι η θέση στην λίστα τόσο λιγότερο διάσημο είναι το κομμάτι. Γενικά αποτελούν ένα μέσο ενημέρωσης και ψυχαγωγίας για τον μέσο χρήστη αλλά και ένα σημαντικότερο εργαλείο για τους επαγγελματίες στον χώρο της μουσικής βιομηχανίας.

Α΄.1 Το πρόβλημα και τα κίνητρα επίλυσης του

Υπάρχουν διάφορες πλατφόρμες που εκδίδουν λίστες δημοφιλών μουσικών κομματιών, μερικές από αυτές είναι το [billboard](http://www.billboard.com)¹, το [spotify](http://www.spotify.com)², το [last.fm](http://www.last.fm)³ και το [shazam](http://www.shazam.com)⁴. Τα κριτήρια, το όριο καθώς και ο τρόπος υπολογισμού κάθε λίστας διαφέρουν αρκετά από πλατφόρμα σε πλατφόρμα. Παρόλο που τις περισσότερες φορές οι πλατφόρμες αυτές είναι αρκετές για να δούμε απλά ποια είναι μερικά από τα διασημότερα τραγούδια της εβδομάδας δεν παρέχουν περαιτέρω ανάλυση, παρακολούθηση και επεξήγηση των δεδομένων. Οι πλατφόρμες δεν μας παρέχουν την δυνατότητα να κάνουμε κάποιας μορφής αντίστροφη αναζήτηση έτσι ώστε να παρακολουθήσουμε την πορεία δημοτικότητας που είχαν ένας καλλιτέχνης ή ένα μουσικό κομμάτι για κάποιο χρονικό διάστημα. Μια διαγραμματική απεικόνιση τέτοιων δεδομένων θα ήταν καίριας σημασίας για την παρακολούθηση της ιστορίας και των τάσεων της μουσικής. Έως τώρα δεν δείχνει να υπάρχει κάποια πλατφόρμα που να έχει ασχοληθεί επίσημα με το θέμα. Παρόλα αυτά αρκετές είναι οι προσπάθειες που έχουν γίνει από διάφορες ερευνητικές ομάδες οι οποίες συνήθως αναφέρονται στο θέμα με τον όρο "Hit song science". Η περαιτέρω έρευνα και πειραματισμός πάνω στην πρόβλεψη των λιστών θα συνέβαλε στην κατανόηση των τάσεων της μουσικής και στην ανάπτυξη της μουσικής βιομηχανίας. Δυστυχώς προς το παρόν καμία από τις πλατφόρμες δεν μας παρέχει την δυνατότητα να αντλήσουμε τα δεδομένα που χρειαζόμαστε με κάποιο απλό και πρακτικό τρόπο όπως θα ήταν ένα API. Κάτι που καθιστά τα δεδομένα δυσπρόσιτα και την άντληση τους χρονοβόρα.

Α΄.2 Συνεισφορά

Για την επίλυση των προβλημάτων που αναφέρονται στην ενότητα 1.1, βασιστήκαμε σε μια παλαιότερη βάση δεδομένων που είχε δημιουργηθεί το 2016 και επεκτάθηκε 2 χρόνια αργότερα, για την οποία θα αναφέρουμε περισσότερα στην ενότητα B.2. Συγκεκριμένα:

- Δημιουργήθηκε μια καινούργια βάση δεδομένων η οποία περιέχει δεδομένα από τρεις πηγές λιστών δημοτικότητας, χρονολογίας από το 1958 έως και σήμερα.
- Δημιουργήθηκε σύστημα αυτόματης ανανέωσης της βάσης δεδομένων, έτσι ώστε η βάση

¹www.billboard.com

²www.spotifycharts.com

³www.last.fm

⁴www.shazam.com

δεδομένων να ανανεώνεται αυτόματα κάθε εβδομάδα.

- Δημιουργήθηκε ιστοσελίδα για την επαναδιάθεση των δεδομένων σε διάφορες διαγραμματικές απεικονίσεις.

Τέλος με την χρήση της παραπάνω βάσης έγινε πείραμα πρόβλεψης των δεδομένων με την χρήση διαφόρων τεχνικών και αλγορίθμων μηχανικής μάθησης.

Α΄.3 Συνοπτική παρουσίαση κεφαλαίων.

Στο επόμενο κεφάλαιο θα γίνει μια συνοπτική παρουσίαση ανάλογων προσπαθειών και ερευνών που έχουν γίνει έως τώρα στο θέμα. Επίσης θα αναφερθούμε στην έρευνα πάνω στην οποία βασίστηκε η παρούσα προσπάθεια. Έπειτα θα γίνει μια ανάλυση της δίκης μας συνεισφοράς στο θέμα. Στο πρώτο μέρος θα γίνει μια ανάλυση της βάσης δεδομένων και του πληροφοριακού συστήματος που δημιουργήθηκε και στο δεύτερο θα γίνει παρουσίαση του πειράματος πρόβλεψης των δεδομένων καθώς και των αποτελεσμάτων που είχε. Τέλος θα παραθέσουμε μια ποιοτική ανάλυση του πειράματος και των συμπερασμάτων που είχαμε.

Κεφάλαιο Β΄

Σχετικές έρευνες.

Β΄.1 Προϋπάρχουσες έρευνες

Στην πραγματικότητα οι προσπάθειες για πρόβλεψη δεδομένων δημοτικότητας μουσικών κομματιών δεν είναι κάτι καινούργιο αλλά υπάρχουν από το 2005. Οι Dhanaraj & Logan(2005) χρησιμοποιώντας τις τεχνικές μηχανικής μάθησης Support vector machine και boosting classifiers πάνω σε μια συλλογή 1700 μουσικών κομματιών προσπάθησαν να ταξινομήσουν τα διάσημα από τα μη διάσημα κομμάτια, λαμβάνοντας υπόψιν τόσο τα ακουστικά χαρακτηριστικά(acoustic features), όσο και τους στίχους των κομματιών.

Αξιόλογο είναι επίσης ότι οι Pachet & Roy(2008) μίλησαν με τον όρο «Hit song science» θέλοντας να αναφερθούν στο πεδίο των ερευνών που στοχεύει στην πρόβλεψη της επιτυχίας που θα έχει ένα μουσικό κομμάτι πριν βγει στην αγορά. Ωστόσο η έρευνα τους καταλήγει στο συμπέρασμα ότι τα συνήθη μουσικά χαρακτηριστικά ενός κομματιού δεν είναι αρκετά ώστε να μας δώσουν αρκετές πληροφορίες για την υποκειμενική κρίση αισθητικής που έχει ο καθένας μας για την μουσική.

Υπάρχουν επίσης έρευνες οι οποίες επικεντρώθηκαν αποκλειστικά στους στίχους των κομματιών όπως η έρευνα των Singhi & Brown(2014) η οποία ξεπέρασε σε επιτυχία πρόβλεψης, κάποιες έρευνες που χρησιμοποιούσαν μόνο ακουστικά χαρακτηριστικά.

Έρευνα των Zangerle,Votter,Huber & Yang(2019) έφτιαξε ένα σετ δεδομένων 11664 μοναδικών κομματιών εκ των οποίων τα μισά είχαν υπάρξει κάποια στιγμή σε κάποια λίστα επιτυχιών του billboard ενώ τα άλλα μισά δεν είχαν υπάρξει ποτέ, δηλαδή δεν ήταν ποτέ επιτυχίες. Συνδυάζοντας χαμηλού και υψηλού επιπέδου μουσικά χαρακτηριστικά μαζί με αλγόριθμους

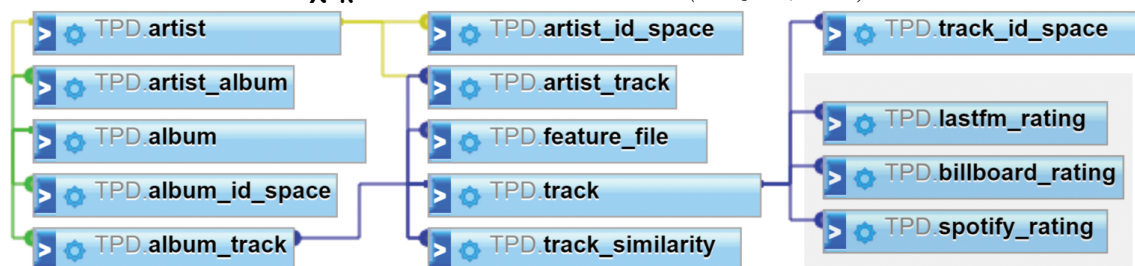
νευρωνικών δικτύων πέτυχαν 75.04% επιτυχία στην πρόβλεψη μουσικών κομματιών που θα γινόταν επιτυχίες.

Μια έρευνα αρκετά σχετική με την παρούσα έγινε το 2019 από τους Middlebrook & Sheik(2019). Στην συγκεκριμένη έρευνα αρχικά δημιουργήθηκε ένα σετ δεδομένων από 24.000 κομμάτια εκ των οποίων τα 12.000 αποτελούσαν κομμάτια που είχαν βρεθεί έστω και μια φορά στα Music charts του Billboard ενώ τα υπόλοιπα 12.000 ήταν τυχαία κομμάτια από το Spotify τα οποία δεν είχαν βρεθεί ποτέ στα charts του billboard. Στην συνέχεια αφού εξήγαγαν τα ακουστικά χαρακτηριστικά για κάθε κομμάτι μέσω του Spotify API, προσπάθησαν να ταξινομήσουν τα κομμάτια σε διάσημα και μη διάσημα χρησιμοποιώντας διάφορες τεχνικές μηχανικής μάθησης.

B'.2 MIR Track popularity prediction

Οι Karydis,Gkiokas & Katsouros(2016) βασιζόμενοι σε μουσικά κομμάτια που βρίσκονταν σε λίστες επιτυχιών από τις πηγές billboard,last.fm και spotify δημιούργησαν μια βάση δεδομένων 23.385 μουσικών κομματιών. Από τα 23.385 κομμάτια τα 9.193 χαρακτηρίστηκαν ως διάσημα αφού υπήρξαν κάποτε σε έστω μια από τις τρεις πηγές ενώ τα υπόλοιπα 14.192 χαρακτηρίστηκαν ως μη διάσημα αφού δεν είχαν υπάρξει ποτέ σε καμία πηγή. Στην πραγματικότητα τα 14.192 μουσικά κομμάτια αντλήθηκαν από τα άλμπουμ στα οποία ανήκαν τα 9.193 διάσημα μουσικά κομμάτια. Επίσης για κάθε διάσημο κομμάτι συμπεριλήφθηκαν τα δεδομένα δημοτικότητας του από τις 3 πηγές, δηλαδή η θέση που είχε για κάθε ξεχωριστή εβδομάδα που βρίσκονταν σε κάποια λίστα επιτυχιών. Σε δεύτερο μέρος της βάσης δεδομένων συμπεριλήφθηκαν τα ακουστικά χαρακτηριστικά για περίπου το 76% των κομματιών. Στην εικόνα B'.1 εμφανίζεται ένα συνοπτικό σχήμα του πρώτου μέρους της βάσης.

Σχήμα B'.1: dataset schema (karydis,2016)



Στην ιστοσελίδα του MIR¹ μπορεί κανείς να κατεβάσει την βάση δεδομένων ή να κάνει κάποιες δοκιμές πάνω σε αυτή. Η βάση επεκτάθηκε το 2018(Karydis,Gkiokas,Katsouros & Pliadis , 2018). Στην παρούσα εργασία βασιστήκαμε στην παραπάνω βάση δεδομένων για να δημιουργήσουμε την δικιά μας και επίσης προσθέσαμε κάποιες λειτουργίες όπως είναι η αυτόματη ανανέωση των δεδομένων της και η αυτόματη διαχείριση των σφαλμάτων κατά την λειτουργία της.

¹<http://mir.ilsp.gr/>

Κεφάλαιο Γ΄

Αποθήκευση και επαναδιαθήκη των δεδομένων

Παρακάτω θα γίνει ανάλυση του σχεδιασμού και της δημιουργίας της βάσης δεδομένων αλλά και επεξήγηση των πηγών και του αυτοματοποιημένου συστήματος ανανέωσης και επαναδιαθέσής της. Ο κώδικας για την συλλογή, ανανέωση και επαναδιαθήκη των δεδομένων γράφτηκε σε γλώσσα PHP¹. Η PHP είναι μία πολύ διαδεδομένη γλώσσα προγραμματισμού γενικής χρήσης, που είναι κατάλληλη για προγραμματισμό διαδικτυακών εφαρμογών. Είναι μια γλώσσα προγραμματισμού που εξειδικεύεται στην κατασκευή δυναμικών ιστοσελίδων. Ιστοσελίδων δηλαδή που μπορούν να τροποποιούνται από το διαχειριστή τους online ή να διαφοροποιούνται, ανάλογα με τα χαρακτηριστικά του χρήστη που τις προβάλλει, όπως για παράδειγμα, το λειτουργικό του σύστημα, η διεύθυνση IP του κ.ά. Για τον κώδικα της ιστοσελίδας χρησιμοποιήθηκαν επίσης οι γλώσσες HTML²,css³ και Javascript⁴. Η HTML είναι ένα σύνολο κανόνων για την διαμόρφωση της εμφάνισης και των επιμέρους στοιχείων μιας ιστοσελίδας. Η css χρησιμοποιείται για την μορφοποίηση των στοιχείων της σελίδας και ουσιαστικά αλλάζει την εμφάνιση της σελίδας. Η JavaScript είναι μια γλώσσα προγραμματισμού και στην παρούσα εργασία μας επιτρέπει την διασύνδεση των στοιχείων της html με τον κώδικα και τις λειτουργίες της php. Για την διαχείριση της βάσης

¹<https://www.php.net/>

²<https://developer.mozilla.org/en-US/docs/Web/HTML>

³<https://www.w3.org/Style/CSS/Overview.en.html>

⁴<https://developer.mozilla.org/en-US/docs/Web/JavaScript>

δεδομένων χρησιμοποιήθηκε η γλώσσα SQL⁵ και συγκεκριμένα το σύστημα διαχείρισης βάσεων δεδομένων MySQL⁶. Η MySQL είναι ένα σύστημα διαχείρισης σχεσιακών βάσεων ανοικτού κώδικα και είναι γνωστή κυρίως για την ταχύτητα, την αξιοπιστία, και την ευελιξία που παρέχει. Για να διαχειριστούμε το σύστημα της MySQL σε γραφικό περιβάλλον χρησιμοποιήθηκε το PhpMyAdmin⁷. Το PhpMyAdmin είναι ένα δωρεάν εργαλείο ανοικτού κώδικα γραφικής διαχείρισης κώδικα MySQL και είναι γραμμένο σε PHP. Η δημιουργία του κώδικα έγινε στο ολοκληρωμένο περιβάλλον ανάπτυξης (IDE) PhpStorm⁸ της JetBrains που βοηθάει σημαντικά στην εγγραφή και την αποσφαλμάτωση του κώδικα

Γ'.1 Πηγές

Όπως προαναφέρθηκε, υπάρχουν διαφορές πηγές λιστών δημοτικότητας μουσικών κομματιών. Αφού λήφθηκε υπόψη ο τρόπος υπολογισμού, ο τύπος δεδομένων και η ευκολία εξαγωγής των δεδομένων της κάθε πηγής, μετά από αρκετούς πειραματισμούς, επιλέχθηκαν τρεις πηγές λιστών δημοτικότητας μουσικών κομματιών. Πρώτη και βασική πηγή αποτέλεσε το billboard⁹. Τα δεδομένα δημοτικότητας που παράγει είναι μια συνάρτηση των πωλήσεων, των διαδικτυακών μεταδόσεων και των ραδιοφωνικών μεταδόσεων που είχε το κάθε κομμάτι κάθε εβδομάδα ("Billboard to Alter Chart Tracking Week for Global Release Date", 2015). Κάθε Τρίτη εκδίδεται και μια καινούργια λίστα ("Billboard Chart & Magazine Dates Now to Align Closer to Release Week", 2017). Τα δεδομένα έχουν αρχική μορφή html και αποτελούν τα πιο παλιά δεδομένα της βάσης μιας και ξεκινούν από το 1958. Δεύτερη πηγή αποτελεί το spotify¹⁰ και συγκεκριμένα το spotifycharts.com. Εκτός από το ότι δημιουργούνται με κάποιον μαθηματικό τύπο που προστατεύει από οποιαδήποτε τεχνητή αλλαγή των αποτελεσμάτων (Spotify for artists FAQ, χ.η.) δυστυχώς δεν υπάρχει κάποια άλλη επίσημη ενημέρωση για τον τρόπο που δημιουργούνται. Κάθε Πέμπτη εκδίδεται και μια καινούργια λίστα. Τα δεδομένα έχουν αρχική μορφή csv και είναι σχετικά πρόσφατα αφού ξεκινούν από τα τέλη του 2016. Τρίτη και τελευταία πηγή αποτελεί το shazam.com¹¹. Το shazam υπολογίζει τα δεδομένα δημοτικότητας με βάση το σύνολο των αναζητήσεων που έγινε στην εφαρμογή του

⁵https://www.w3schools.com/sql/sql_intro.asp

⁶<https://www.mysql.com/>

⁷<https://www.phpmyadmin.net/>

⁸<https://www.jetbrains.com/phpstorm/>

⁹<https://www.billboard.com/>

¹⁰<https://www.spotifycharts.com/>

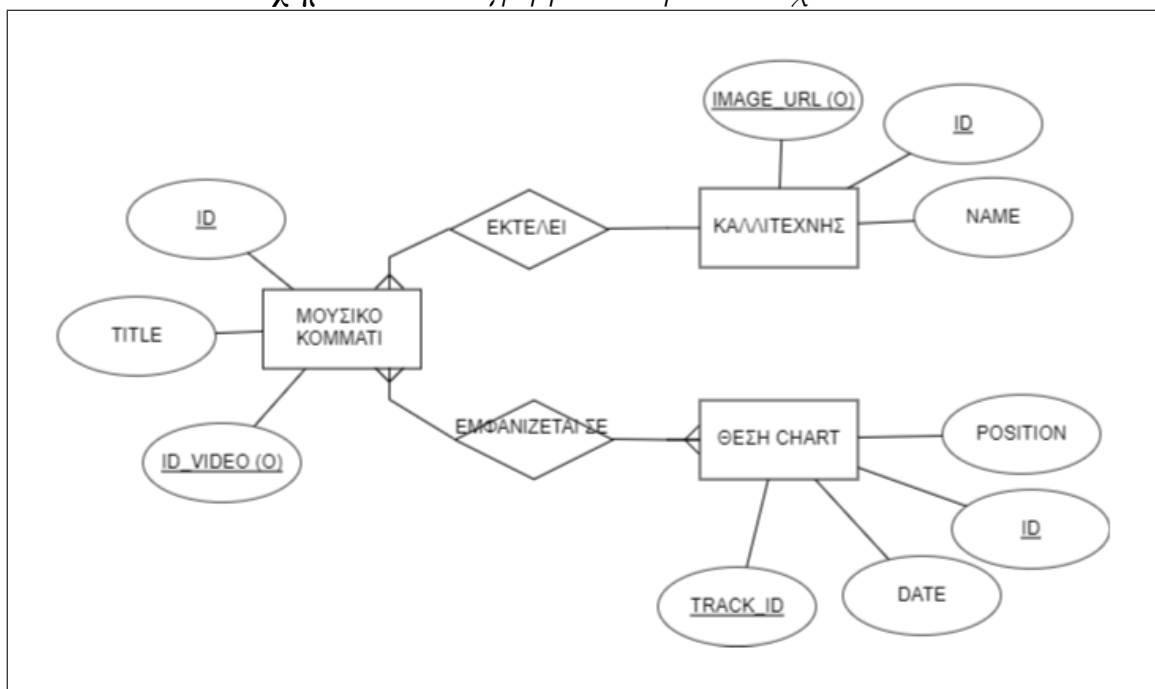
¹¹<https://www.shazam.com/>

shazam για κάθε κομμάτι για μια εβδομάδα (Shazam data,2021). Κάθε εβδομάδα όσο πιο πολλές αναζητήσεις είχε ένα κομμάτι τόσο πιο ψηλά θα βρίσκεται στην λίστα. Κάθε Κυριακή εκδίδεται και μια καινούργια λίστα. Δυστυχώς η σελίδα δεν παρέχει την δυνατότητα να δούμε παλαιότερες λίστες επιτυχιών πάρα μόνο την τωρινή. Τα δεδομένα έχουν αρχική μορφή json και ξεκινούν μόλις από το 2019 αφού λόγω του προβλήματος που προαναφέρθηκε δεν μπορούμε να τραβήξουμε παλαιότερα δεδομένα.

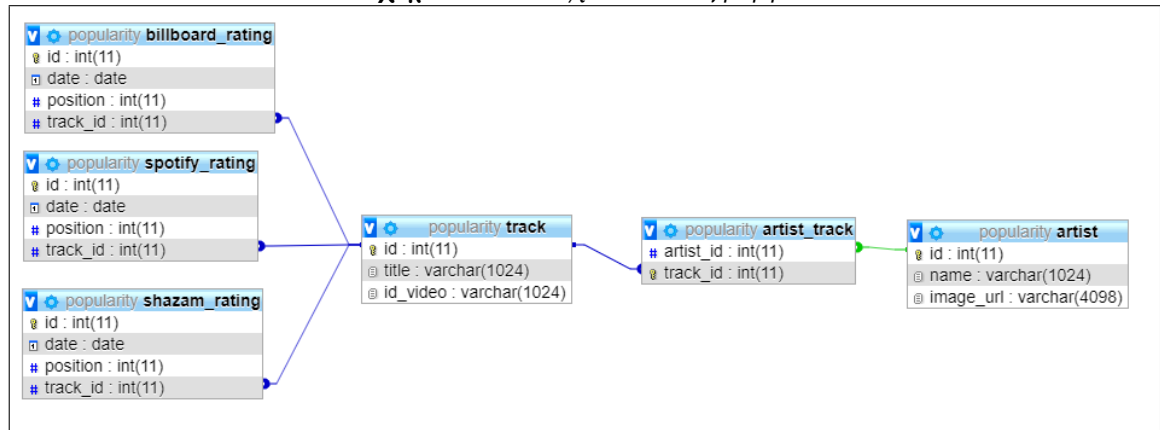
Γ'.2 Σχεδιασμός και μορφή

Αφού λήφθηκε υπόψιν η φύση των δεδομένων αλλά και ο σκοπός για τον οποίο θα χρησιμοποιηθούν , πραγματοποιήθηκε ο σχεδιασμός της βάσης δεδομένων. Σκοπός ήταν να δημιουργηθεί μια βάση απλή στην χρήση αλλά και εύκολη στην ανανέωση και την συντήρηση. Στην εικόνα Γ'.2 παρουσιάζεται το διάγραμμα οντοτήτων - συσχετίσεων ενώ στην εικόνα Γ'.3 το σχεσιακό διάγραμμα.

Σχήμα Γ'.2: Διάγραμμα οντοτήτων - συσχετίσεων



Σχήμα Γ'.3: Σχεσιακό διάγραμμα



Η βάση αποτελείται από έξι απλούς πίνακες. Αρχικά έχουμε τους πίνακες `billboard_rating`, `spotify_rating` και `shazam_rating` στους οποίους περιέχεται κάθε εγγραφή από τις λίστες δημοτικότητας που συλλέγονται από κάθε πηγή. Στην παρούσα εργασία συλλέξαμε τις λίστες δημοτικότητας εκατό κομματιών (Top 100) άρα κάθε φορά που ‘τραβάμε’ λίστες δημοτικότητας για μία εβδομάδα από τις τρεις πηγές, στον κάθε πίνακα προστίθενται από 100 εγγραφές. Όσο αναφορά την ταξινόμηση τους πρώτα έχουμε τα πιο παλιά δεδομένα και έπειτα τα πιο νέα. Δηλαδή κάθε νέα εγγραφή που συλλέγουμε κατατάσσεται τελευταία. Οι τρεις πίνακες έχουν τα εξής πεδία:

- `Id` (Μοναδικός κωδικός της κάθε εγγραφής. Αποτελεί το πρωτεύον κλειδί των πινάκων)
- `Date` (Ημερομηνία της λίστας επιτυχιών, στην οποία ανήκει η εγγραφή)
- `Track_id` (Ο μοναδικός κωδικός του μουσικού κομματιού)
- `Position` (Η θέση που είχε το συγκεκριμένο μουσικό κομμάτι στην συγκεκριμένη λίστα επιτυχιών. Εύρος τιμών 1 έως 100)

Μετά έχουμε τον πίνακα `track` που περιέχει κάθε μοναδικό μουσικό κομμάτι που υπήρξε έστω και μια φορά σε κάποια από τις λίστες επιτυχιών. Για κάθε νέα εγγραφή που προστίθεται από τις λίστες δημοτικότητας, ελέγχεται και αν υπάρχει ήδη το αντίστοιχο κομμάτι εδώ. Αν δεν υπάρχει τότε προστίθεται. Έχει τα εξής πεδία:

- Id (Μοναδικός κωδικός του κάθε κομματιού. Αποτελεί το πρωτεύον κλειδί του πίνακα)
- Title (Ο τίτλος του κομματιού)
- id_video (Το μοναδικό id που παρέχει το spotify για κάθε κομμάτι, το οποίο χρησιμοποιήθηκε στο πείραμα της πρόβλεψης των δεδομένων.)

Έπειτα έχουμε τον πίνακα "artist" που περιέχει κάθε ένα μοναδικό καλλιτέχνη που υπήρξε έστω και μια φορά σε κάποια από τις λίστες επιτυχιών. Όπως και στον πίνακα "track" για κάθε νέα εγγραφή που προστίθεται από τις λίστες δημοτικότητας, ελέγχεται και αν υπάρχει ήδη ο αντίστοιχος καλλιτέχνης εδώ. Αν δεν υπάρχει τότε προστίθεται. Συνδυασμοί καλλιτεχνών αποθηκεύονται στον πίνακα σαν να ήταν ένας νέος καλλιτέχνης ακόμα και αν ένας εκ των καλλιτεχνών υπάρχει ήδη στον πίνακα. Αυτό γίνεται για να αντιμετωπίσουμε το πρόβλημα των σχεδόν άπειρων τρόπων αναφοράς δυο η περισσότερων καλλιτεχνών σε ένα κομμάτι, όπως η χρήση σημείων στίξης, διάφορων συμβόλων η ακόμη και λέξεων τα οποία δεν θα μας επέτρεπαν να διακρίνουμε τους διάφορους καλλιτέχνες από ένα κομμάτι σε ένα αυτοματοποιημένο σύστημα συλλογής τους. Επίσης η προσέγγιση αυτή παρέχει στην συνέχεια την δυνατότητα της αυτόματης επαναδιάθεσης των δημοφιλών συνεργασιών που έχουν υπάρξει κατά καιρούς, κάτι που κάνει την περιήγηση του χρήστη σε αυτές πιο εύκολη και ενημερωτική. Ο πίνακας έχει τα εξής πεδία:

- id (Μοναδικός κωδικός του κάθε καλλιτέχνη. Αποτελεί το πρωτεύον κλειδί των πίνακα)
- name (Το όνομα του καλλιτέχνη)
- image_url (Σύνδεσμος με εικόνα του καλλιτέχνη/ών)

Τέλος έχουμε τον πίνακα "artist_track" που απλά συνδέει τον κάθε καλλιτέχνη με τα διάφορα κομμάτια του. Έχει τα εξής πεδία:

- artist_id (Μοναδικός κωδικός του κάθε καλλιτέχνη, Αποτελεί το πρωτεύον κλειδί του πίνακα)
- track_id (Μοναδικός κωδικός του κάθε κομματιού)

Γ'.3 Αρχική συλλογή δεδομένων

Λόγω του μεγέθους των δεδομένων αλλά και της δυσκολίας συλλογής τους από τις πηγές, μεγάλη έμφαση δόθηκε στην ταχύτητα, την ανθεκτικότητα και την αυτοματοποιημένη διόρθωση οποιουδήποτε προβλήματος μπορεί να παρουσιαστεί κατά την εκτέλεση του αρχικού συστήματος. Αρχικά το σύστημα μιμούμενο έναν απλό χρήστη σε έναν web browser περισυλλέγει(caching) τις λίστες επιτυχιών από τις τρεις πηγές με διάφορες τεχνικές ανάλογα και τον αρχικό τύπο δεδομένων που προσφέρει η κάθε πηγή (json,csv κ.λπ.). Αν και αυτή η ενέργεια αρχικά μοιάζει να χρήζει επιβεβαίωσης όσο αφορά τη νομιμότητά της για λόγους πνευματικών δικαιωμάτων καθώς (α) η χρήση στην οποία εφαρμόζεται είναι αμιγώς ακαδημαϊκή, και (β) δεν υπάρχει κανένα πλαίσιο οικονομικής εκμετάλλευσης, και (γ) ήταν - αλλά και παραμένει - μικρού εύρους, και (δ) αξιολογείται πως δεν έχει αρνητική επίπτωση στις αρχικές πηγές - τουναντίον, προωθεί τις δράσεις τους μέσω των διαδικτυακών αναφορών που δημιουργούμε - θεωρούμε πως σε πρόσκληση να αιτιολογήσουμε τις δράσεις μας, η επίκληση στην έννοια "fair use / fair dealing" (Dimitra Karydi, Ioannis Karydis, 2014) θα ήταν ευρέως αποδεκτή / βάσιμη. Τα δεδομένα αρχικά αποθηκεύονται πρώτα σαν αρχεία έτσι ώστε να χρησιμοποιηθούν αργότερα ή ακόμα και στο μέλλον σε περίπτωση απώλειας της βάσης δεδομένων. Επίσης πριν από κάθε επίσκεψη που κάνουμε στις πηγές υπάρχει και μια εκούσια χρονική καθυστέρηση από το σύστημα. Σε συνδυασμό αυτά τα δυο στοιχεία μειώνουν τον φόρτο εργασίας που ασκούμε στις πηγές αφού πρώτον ορίζουν ένα όριο στο πόσες φορές θα τις επισκεφθούμε και δεύτερον γιατί στο μέλλον δεν θα χρειαστεί να τις ξαναενοχλήσουμε για τα ίδια δεδομένα ακόμα και αν έχουμε μερική ή ολική απώλεια της βάσης δεδομένων. Έτσι μειώνουμε συνολικά τους πόρους που πρέπει να χρησιμοποιηθούν τόσο από τις πηγές όσο και από το τρέχων σύστημα. Στην συνέχεια τα αρχεία διαβάζονται και προστίθενται στην βάση δεδομένων τα ανάλογα δεδομένα. Για κάθε νέο καλλιτέχνη και κομμάτι που προστίθεται υπάρχει και ο ανάλογος έλεγχος έτσι ώστε να αποφευχθούν τα διπλότυπα. Έπειτα τρέχουμε διάφορα τεστ τα οποία ελέγχουν αν υπήρχε κάποιο λάθος στην μορφή των δεδομένων ή αν απουσιάζει κάποιο από αυτά. Τέλος δημιουργείται αυτόματα ένα δοκιμαστικό διάγραμμα που απεικονίζει την πορεία ενός τυχαίου κομματιού.

Γ'.4 Ανανέωση και συντήρηση

Η ανανέωση της βάσης δεδομένων με νέα δεδομένα γίνεται περίπου με τον ίδιο τρόπο που γίνεται η αρχική συλλογή των δεδομένων. Η κύρια διαφορά είναι πως το σύστημα πρώτα ελέγχει ποιες ημερομηνίες λείπουν από την βάση μέχρι την σημερινή μέρα και στην συνέχεια προσθέτει τις ανάλογες λίστες επιτυχιών. Το σύστημα είναι προγραμματισμένο να εκτελείται αυτόματα κάθε εβδομάδα αλλά λόγω του παραπάνω ελέγχου μπορεί να εκτελεστεί επιτυχώς ακόμα και αν κάποια εβδομάδα αποτύχει να εκτελεστεί όπως έπρεπε και κάποια δεδομένα λείπουν. Λόγω της αυτοματοποιημένης φύσης του κατά την διάρκεια της εκτέλεσης, σε κάθε κρίσιμο σημείο υπάρχει και ο ανάλογος κώδικας που ελέγχει αν το κομμάτι του κώδικα εκτελείται σωστά. Σε περίπτωση που δημιουργηθεί οποιοδήποτε πρόβλημα κατά την διάρκεια της εκτέλεσης το σύστημα διακόπτει αμέσως την εκτέλεσή του και καταγράφει αναλυτικές πληροφορίες για τον τύπο του προβλήματος καθώς και σε ποιο σημείο του κώδικα συνέβη. Έπειτα ενημερώνονται αμέσως μέσω σχετικού e-mail οι διαχειριστές του συστήματος για να διευθετήσουν το πρόβλημα άμεσα. Αντίθετα σε περίπτωση που το πρόγραμμα εκτελεστεί όπως θα έπρεπε και οι έλεγχοι του δείχνουν ότι οι βάσεις ανανεώθηκαν όπως θα έπρεπε, τότε ενημερώνονται αμέσως οι διαχειριστές του συστήματος μέσω email με ένα συνοπτικό μήνυμα που τους ειδοποιεί ότι η προγραμματισμένη εκτέλεση του συστήματος ήταν επιτυχής καθώς και για ποιες ημερομηνίες προστέθηκαν λίστες επιτυχιών στην βάση δεδομένων. Έτσι ακόμα και αν έχουμε κάποια μεγάλη αλλαγή στις πηγές ή οτιδήποτε άλλο δεν δουλέψει όπως θα έπρεπε οι διαχειριστές θα είναι θέση να διευθετήσουν το πρόβλημα εύκολα και γρήγορα. Κάτι που καθιστά το σύστημα ανθεκτικό στον χρόνο και στις συνεχείς αλλαγές που συμβαίνουν αρκετά συχνά στις τρεις πηγές λιστών επιτυχίας.

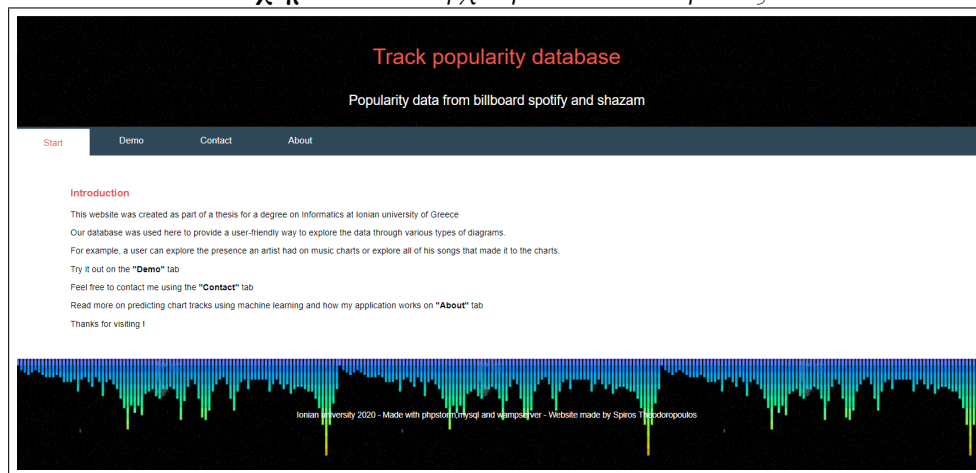
Γ'.5 Περιεχόμενο

Όσον αφορά τον όγκο των δεδομένων η βάση δεδομένων περιείχε περίπου 337.981 εγγραφές μέχρι τα μέσα Σεπτεμβρίου του 2020. Από αυτές περίπου 284.452 εγγραφές είναι δεδομένα δημοτικότητας από περίπου 2.844 λίστες επιτυχιών. Έπειτα περιέχει 22.600 μουσικά κομμάτια και 8.329 καλλιτέχνες. Γενικά τα δεδομένα του billboard υπερσχύουν αφού χρονολογούνται από το 1958. Τέλος τα δεδομένα του Spotify χρονολογούνται από το 2016 ενώ του Shazam από το 2019.

Γ'.6 Επαναδιαθεση δεδομένων

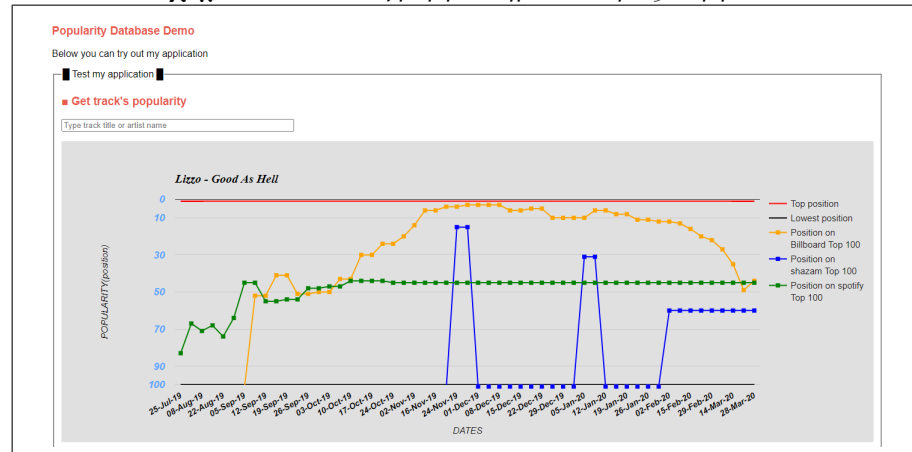
Με σκοπό να καλυφθούν τα κενά που παρουσιάζονται στην περαιτέρω ανάλυση των δεδομένων δημιουργήθηκε μια απλή ιστοσελίδα όπου πραγματοποιείται ανάλυση των δεδομένων σε διάφορα διαγράμματα. Η ιστοσελίδα έχει 4 διαφορετικές καρτέλες όπως φαίνονται και στην εικόνα Γ'.4 και παρακάτω θα τις επεξηγήσουμε μια προς μια. Στην πρώτη καρτέλα και αρχική

Σχήμα Γ'.4: Αρχική σελίδα και καρτέλες.



σελίδα, ο χρήστης μπορεί να διαβάσει μια μικρή περιγραφή του συστήματος του σκοπού του, της χρήσης του καθώς και πως φτιάχτηκε όπως φαίνεται και στην εικόνα Γ'.4. Στην δεύτερη καρτέλα της ιστοσελίδας ο χρήστης μπορεί να κάνει δοκιμή του συστήματος επαναδιάθεσης και να περιηγηθεί εκτενώς στα δεδομένα της βάσης δεδομένων. Ειδικότερα για αρχή μπορεί να αναζητήσει κάποιο μουσικό κομμάτι και να δει ένα διάγραμμα της πορείας που είχε στις λίστες επιτυχιών και των τριών πηγών, όπως φαίνεται στην εικόνα Γ'.5. Έπειτα ο χρήστης μπορεί να αναζητήσει έναν καλλιτέχνη για να περιηγηθεί σε μια λίστα από τα 10 πιο διάσημα κομμάτια του με βάση τα δεδομένα δημοτικότητας του συστήματος, όπως φαίνεται στην εικόνα Γ'.6. Εδώ ο χρήστης έχει την επιλογή να αναζητήσει είτε έναν μεμονωμένο καλλιτέχνη είτε τον καλλιτέχνη σε συνεργασία με κάποιον άλλο καλλιτέχνη, όπως φαίνεται στην εικόνα Γ'.7. Επίσης πατώντας σε οποιαδήποτε από τα κομμάτια που εμφανίζονται μπορεί να δει αναλυτικά την πορεία που είχε στις λίστες επιτυχιών των τριών πηγών, όπως φαίνεται στην εικόνα Γ'.8. Μετά παρέχεται η δυνατότητα να δούμε την πορεία που είχε ο ίδιος ο καλλιτέχνης στις λίστες επιτυχιών ενός ή και των τριών πηγών, όπως φαίνεται στην εικόνα Γ'.9, δηλαδή τότε εμφανίζεται σε λίστες επιτυχιών, για πόσο χρόνο και σε ποια πηγή. Εδώ ο χρήστης

Σχήμα Γ'.5: Διάγραμμα δημοτικότητας κομματιού.



μπορεί να αναζητήσει όλες τις εγγραφές που περιέχεται το όνομα του καλλιτέχνη που θέλει να αναζητήσει, δηλαδή ακόμη και συνεργασίες που είχε. Ενώ τσεκάροντας την επιλογή "Solo artist search" και γράφοντας το ακριβές όνομά του έχει την επιλογή να αναζητήσει μεμονωμένα τον καλλιτέχνη, όπως φαίνεται στην εικόνα Γ'.10. Επιπρόσθετα ο χρήστης μπορεί να περιηγηθεί στις ίδιες τις λίστες επιτυχιών για οποιαδήποτε από τις τρεις πηγές και οποιαδήποτε ημερομηνία το επιθυμεί. Επίσης δίνεται και η επιλογή να εμφανίσει όσα κομμάτια θέλει. Πατώντας σε οποιοδήποτε από τα κομμάτια που εμφανίζονται μπορεί να δει αναλυτικά την πορεία που είχε στις λίστες επιτυχιών των τριών πηγών. Η λειτουργία παρουσιάζεται στην εικόνα Γ'.11. Στην τρίτη καρτέλα της ιστοσελίδας ο χρήστης σε περίπτωση που το επιθυμεί μπορεί να συμπληρώσει μια μικρή φόρμα επικοινωνίας και να στείλει ένα μήνυμα στους διαχειριστές του συστήματος. Η λειτουργία παρουσιάζεται στην εικόνα Γ'.12. Στην τέταρτη και τελευταία καρτέλα της ιστοσελίδας ο χρήστης μπορεί να διαβάσει κάποιες περαιτέρω πληροφορίες για το σύστημα όπως είναι οι τεχνολογίες και τα εργαλεία που χρησιμοποιήθηκαν για να το δημιουργήσουμε καθώς και από ποιους φτιάχτηκε.

Γ'.7 Μελλοντική χρήση συστήματος

Για την μελλοντική αξιοποίηση της εφαρμογής μας σε ένα πραγματικό σενάριο και περιβάλλον χρήσης της χρησιμοποιήσαμε το Docker¹². Το Docker είναι λογισμικό ανοιχτού κώδικα το οποίο εκτελεί εικονοποίηση σε επίπεδο λειτουργικού συστήματος, γνωστή και ως "containe-

¹²<https://www.docker.com/>

Σχήμα Γ'.6: Αναζήτηση μεμονωμένου καλλιτέχνη.

■ Get artist's top 10 popular tracks

☐ Solo artist search ?(enter full artist name for best results)

ed sheeran

Many artists found. Please select one or type full artist name for better results:

1. Taylor Swift Featuring Ed Sheeran
2. Lupe Fiasco Featuring Ed Sheeran
3. Hoodie Allen Featuring Ed Sheeran
4. The Weeknd Featuring Ed Sheeran
5. Rudimental Featuring Ed Sheeran
6. Taylor Swift Featuring Ed Sheeran & Future
7. Eminem Featuring Ed Sheeran
8. Ed Sheeran & Justin Bieber
9. Ed Sheeran Featuring Chance The Rapper & PnB Rock
10. Ed Sheeran Featuring Khalid
11. Ed Sheeran With Chris Stapleton & Bruno Mars
12. Ed Sheeran & Travis Scott

Σχήμα Γ'.7: Αναζήτηση συνεργασιών.

■ Get artist's top 10 popular tracks

☒ Solo artist search ?(enter full artist name for best results)

ed sheeran

You mean this artist ?

1. Ed Sheeran

rization". Επιτρέπει την ταχεία μεταφορά και επιτυχή εκτέλεση ολοκληρωμένων εφαρμογών από οποιοδήποτε σύστημα σε οποιοδήποτε σύστημα. Δηλαδή από οποιαδήποτε λογισμικό σε οποιοδήποτε (mac,windows,linux) με όποιες ρυθμίσεις και σε όποια έκδοση και αν βρίσκεται το καθένα χωρίς να χρειαστεί να πειράζουμε/αλλάζουμε κάτι στο host σύστημα. Με τις κατάλληλες ρυθμίσεις η μεταφορά αυτή μπορεί να γίνει απλά με μια εντολή.(Getting started Docker,2021). Έτσι χρησιμοποιώντας το Docker φτιάξαμε το απαραίτητο αρχείο 'docker-compose' το οποίο μπορεί να στήσει αυτόματα το απαιτούμενο για την εφαρμογή περιβάλλον, σε όποιο σύστημα το επιθυμήσουμε και κατά συνέπεια να εκτελεστεί επιτυχώς η εφαρμογή.

Σχήμα Γ'.8: Αποτέλεσμα αναζήτησης για τα κομμάτια του καλλιτέχνη.

■ Get artist's top 10 popular tracks

☒ Solo artist search ?(enter full artist name for best results)

ed sheeran

1. Perfect - Top char position: 1
2. Shape of You - Top char position: 1
3. I Don't Care (with Justin Bieber) - Top char position: 1
4. Thinking out Loud - Top char position: 2
5. Galway Girl - Top char position: 2
6. Castle on the Hill - Top char position: 2
7. Perfect Duet (Ed Sheeran & Beyoncé) - Top char position: 3
8. Beautiful People (feat. Khalid) - Top char position: 3
9. Antisocial (with Travis Scott) - Top char position: 7
10. Don't - Top char position: 9

Σχήμα Γ'.9: Αναζήτηση καλλιτέχνη.

■ Get artist's time on top chart

☒ Solo artist search ?(enter full artist name for best results)

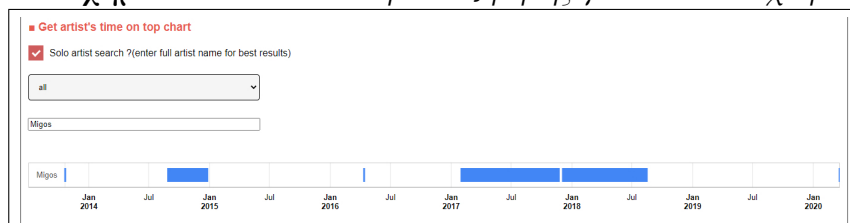
all

Migos

You mean this artist ?

1. Migos

Σχήμα Γ'.10: Αποτέλεσμα αναζήτησης για τον καλλιτέχνη.



Σχήμα Γ'.11: Αποτέλεσμα αναζήτησης λίστας δημοτικότητας.

■ Get a week's tracks on top chart

Spotify 2017-03-23

Select # of top results 5

1. **Shape of You** by Ed Sheeran
previous week position: 1
best position up until previous week: 1
weeks on chart up until previous week: 10
2. **It Ain't Me (with Selena Gomez)** by Kygo
previous week position: 3
best position up until previous week: 2
weeks on chart up until previous week: 4
3. **Something Just Like This** by The Chainsmokers
previous week position: 2
best position up until previous week: 2
weeks on chart up until previous week: 3
4. **Passionfruit** by Drake
previous week position:
best position up until previous week:
weeks on chart up until previous week:
5. **Galway Girl** by Ed Sheeran
previous week position: 2
best position up until previous week: 2
weeks on chart up until previous week: 2

Σχήμα Γ'.12: Φόρμα Επικοινωνίας.

Track popularity database

Popularity data from billboard spotify and shazam

Start Demo **Contact** About

[Contact us, send us your thoughts ideas or problems](#)

First Name

Last Name

Email

Subject

London university 2020. Made with [Alpine.js](#) and [webpack](#). Theme made by [Spinel Themes](#)

Κεφάλαιο Δ΄

Ανάλυση και πρόβλεψη των δεδομένων

Η πρόβλεψη των κομματιών που θα γίνουν δημοφιλή θα προέτρεπε τους καλλιτέχνες να επικεντρώσουν την δουλειά τους σε κομμάτια που είχαν τις περισσότερες πιθανότητες να γίνουν δημοφιλή. Κάτι τέτοιο θα αύξανε σημαντικά τα κέρδη των καλλιτεχνών και θα συνέβαλε στην ανάπτυξη της μουσικής βιομηχανίας. Εκτός αυτού αλγόριθμοι πρόβλεψης δημοφιλών κομματιών θα μπορούσαν να βοηθήσουν τις πλατφόρμες να προτείνουν καλύτερα κομμάτια στους χρήστες τους, κάτι το οποίο με την σειρά του θα βοηθούσε τους χρήστες να ανακαλύψουν περισσότερη μουσική που τους αρέσει ανεξάρτητα το είδος. Παρόλα αυτά δεν θα μπορούσαμε να μην αναφέρουμε και τις αρνητικές συνέπειες ενός τέτοιου συστήματος όπως είναι η καταπάτηση της καλλιτεχνικής έκφρασης στο όνομα της δημοφιλίας. Για την δημιουργία του σετ δεδομένων καθώς και για την προ-επεξεργασία και πρόβλεψη των δεδομένων χρησιμοποιήθηκε η γλώσσα `python`¹ σε περιβάλλον `Jupyter notebook`² και λειτουργικό σύστημα `Ubuntu`³. Τέλος για την ανάλυση και την πρόβλεψη των δεδομένων χρησιμοποιήθηκαν οι βιβλιοθήκες: `Numpy`⁴, `Pandas`⁵, `Matplotlib`⁶, `scikit-learn`⁷ της `python`. Παρακάτω θα γίνει παρουσίαση όλων των εργαλείων. Επειτα θα γίνει επεξήγηση του

¹<https://www.python.org/>

²<https://jupyter.org/>

³<https://ubuntu.com/>

⁴<https://numpy.org/doc/stable/contents.html>

⁵<https://pandas.pydata.org/>

⁶<https://matplotlib.org/>

⁷<https://scikit-learn.org/stable/>

σετ δεδομένων που δημιουργήθηκε καθώς και των τεχνικών που χρησιμοποιήσαμε. Στο τέλος θα γίνει ανάλυση και αξιολόγηση των αποτελεσμάτων.

Δ΄.1 Python και jupyter notebook

Η Python είναι μια γλώσσα προγραμματισμού με απλό συντακτικό, εξαιρετική αναγνωσιμότητα και φορητότητα. Είναι ιδιαίτερα δημοφιλής γλώσσα ιδιαίτερα στο πεδίο της επιστήμης των δεδομένων. Είναι μία γλώσσα υψηλού επιπέδου που σημαίνει ότι ο κώδικας της πρέπει να μετατραπεί σε γλώσσα μηχανής ώστε να εκτελεστεί από τον Υπολογιστή. Η επεξεργασία αυτή γίνεται από διερμηνευτές (interpreters) και μεταγλωττιστές (compilers). Στην περίπτωση της python η επεξεργασία γίνεται από διερμηνευτή. ("Python language",χ.η.)

Μπορεί να χρησιμοποιηθεί τόσο για τη ανάπτυξη ολοκληρωμένων εφαρμογών όσο και για την δημιουργία απλών σεναρίων εντολών. Τα προγράμματα σε Python είναι συμπαγή, ευανάγνωστα και συντηρούνται γρηγορότερα σε σχέση με άλλες δημοφιλείς γλώσσες προγραμματισμού. Τέλος διαθέτει πληθώρα έτοιμων βιβλιοθηκών που μπορούν να χρησιμοποιηθούν εύκολα και άμεσα, όπως αυτές που χρησιμοποιήθηκαν κατά την παρούσα έρευνα και αναφέρονται αναλυτικότερα παρακάτω. Το Jupyter Notebook είναι μια διαδικτυακή εφαρμογή ανοιχτού κώδικα που χρησιμοποιείται για την δημιουργία και τον διαμοιρασμό εγγράφων που περιέχουν ζωντανό κώδικα, εξισώσεις, οπτικοποιήσεις και κείμενο.(Install and Use Jupyter,2021). Είναι ένας δημοφιλής, γρήγορος και ισχυρός τρόπος για να γράψουμε και να εκτελέσουμε κώδικα Python, ιδιαίτερα όταν αφορά προβλήματα στο πεδίο της επιστήμης των δεδομένων. Αντί να γράφουμε και να ξαναγράφουμε ένα ολόκληρο πρόγραμμα, μας επιτρέπει να γράψουμε γραμμές κώδικα και να τις εκτελέσουμε μία-μία κάθε φορά. Στη συνέχεια, εάν θέλουμε να κάνουμε μια αλλαγή, μπορούμε να επιστρέψουμε στην γραμμή που θέλουμε, να κάνουμε την επεξεργασία και να εκτελέσουμε ξανά όλο το πρόγραμμα στο ίδιο παράθυρο. Όλος ο πειραματισμός έγινε σε λειτουργικό σύστημα Ubuntu. Το ubuntu είναι ένα ανοιχτού κώδικα, δωρεάν και δημοφιλές λειτουργικό σύστημα βασισμένο σε linux.

Δ΄.2 NumPy, Pandas και scikit-learn

Το NumPy (Numerical Python) είναι μια βιβλιοθήκη γραμμικής άλγεβρας στην Python. Πρόκειται για μια πολύ σημαντική βιβλιοθήκη στην οποία εξαρτάται σε μεγάλο βαθμό σχεδόν κάθε πακέτο Python για την επιστήμη δεδομένων ή τη μηχανική εκμάθηση, όπως το SciPy

(Scientific Python), το Matplotlib, το Scikit-Learn, κ.λπ. Το NumPy είναι πολύ χρήσιμο για την εκτέλεση μαθηματικών και λογικών λειτουργιών σε Πίνακες. Παρέχει μια πληθώρα χρήσιμων λειτουργιών για εργασίες σε πίνακες στην Python. (What is NumPy?, 2021) Το pandas είναι μια δωρεάν βιβλιοθήκη της Python που παρέχει γρήγορες, ευέλικτες και εκφραστικές δομές δεδομένων που έχουν σχεδιαστεί για να κάνουν την εργασία με ‘σχεσιακά’ ή ‘επισημασμένα’ δεδομένα εύκολη και διαισθητική. Οι δύο κύριες δομές δεδομένων του Pandas, τα Series (Μονοδιάστατη) και τα DataFrame (Δισδιάστατη), μπορούν να χρησιμοποιηθούν στην συντριπτική πλειονότητα των τυπικών περιπτώσεων χρήσης στην στατιστική, τις κοινωνικές επιστήμες και σε πολλούς τομείς της μηχανικής. Το pandas είναι χτισμένο πάνω από το NumPy. Είναι γρήγορο και είναι εξαιρετικά δημοφιλές στον τομέα της επιστήμης των δεδομένων. Τέλος είναι εξαιρετικό στην αντιμετώπιση διαφόρων προβλημάτων στον τομέα της επιστήμης των δεδομένων όπως είναι τα ελλιπή δεδομένα και η συγχώνευση και ένωση συνόλων δεδομένων.(Pandas Package overview, 2021)

Το Scikit-Learn είναι μια δωρεάν βιβλιοθήκη μηχανικής μάθησης ανοιχτού κώδικα που υποστηρίζει επιβλεπόμενη και μη επιβλεπόμενη μηχανική μάθηση. Παρέχει επίσης διάφορα εργαλεία για την προ-επεξεργασία των δεδομένων, την δημιουργία, επιλογή και αξιολόγηση μοντέλων και πολλά άλλα βοηθητικά εργαλεία. Είναι βασισμένο σε δημοφιλείς τεχνολογίες, όπως το NumPy το pandas και το Matplotlib. Διαθέτει υψηλό επίπεδο υποστήριξης και αυστηρή διακυβέρνηση για την ανάπτυξη της βιβλιοθήκης, πράγμα που την κάνει ένα απίστευτα ισχυρό εργαλείο. Αποτελείται από ένα σαφές και συνεπές στυλ κώδικα που διασφαλίζει ότι ο κώδικας μηχανικής εκμάθησης είναι εύκολο να κατανοηθεί και να αναπαραχθεί, κάτι που την καθιστά εύκολη στην εκμάθηση της. Επίσης υποστηρίζεται από πολλά τρίτα εργαλεία που εμπλουτίζουν ακόμη περισσότερο τη λειτουργικότητα της σε διάφορες περιπτώσεις χρήσης. Θεωρείται ως μια από τις πιο διαδεδομένες βιβλιοθήκες για machine learning της python. (Scikit-Learn Getting started,2021)

Η Matplotlib είναι μια δωρεάν ανοιχτού κώδικα βιβλιοθήκη της Python, που χρησιμοποιείται για τη δημιουργία γραφημάτων, διαγραμμάτων και σχημάτων υψηλής ποιότητας. Η βιβλιοθήκη είναι εκτεταμένη και είναι ικανή να αλλάζει την οποιαδήποτε λεπτομέρεια ενός σχήματος ή διαγράμματος. Έτσι μας επιτρέπει να δημιουργούμε αναλυτικά και όμορφα διαγράμματα με τα δεδομένα των αναλύσεων. Το matplotlib έχει σχεδιαστεί με τη φιλοσοφία ότι απλά με λίγες εντολές ή και μόνο μία εντολή μπορούμε να δημιουργήσουμε ολοκληρωμένα διαγράμματα.(Matplotlib User’s Guide,2021)

Δ'.3 Αρχικό Σετ Δεδομένων.

Χρησιμοποιώντας την δική μας βάση δεδομένων που δημιουργήσαμε κατά την συλλογή των δεδομένων δημιουργήσαμε το κατάλληλο σετ δεδομένων για τον πειραματισμό μας. Με την χρήση των κατάλληλων SQL ερωτημάτων και python κώδικα εξάχθηκε κάθε ξεχωριστή εγγραφή από τον πίνακα «track» μαζί με το όνομα του καλλιτέχνη του κάθε κομματιού και το μοναδικό id που παρέχει το Spotify για κάθε κομμάτι. Παρακάτω εμφανίζεται η SQL εντολή που χρησιμοποιήθηκε εντός του python κώδικα.

```
1 SELECT track.title,artist.name,track.id_video FROM track
2 LEFT JOIN artist_track ON track.id = artist_track.track_id
3 LEFT JOIN artist ON artist_track.artist_id = artist.id
```

Δυστυχώς παρόλο που το billboard συνήθιζε να παρέχει και το Spotify id στα δεδομένα του, πια δεν υποστηρίζει αυτή την λειτουργία. Το ίδιο πρόβλημα παρουσιάζει και το shazam. Κάτι που σημαίνει ότι έχουμε το Spotify id μόνο για τα κομμάτια που έχουν υπάρξει στις λίστες επιτυχιών του Spotify. Από τα 22.600 κομμάτια που περιέχει η βάση δεδομένων, το Spotify id υπήρχε μόνο για τα 1500 από αυτά. Έτσι προχωρήσαμε σε περαιτέρω εξόρυξη των δεδομένων έτσι ώστε να αξιοποιηθούν στο έπακρο οι δυνατότητες της βάσης δεδομένων μας. Χρησιμοποιώντας το api που παρέχει το Spotify αναζητήσαμε το Spotify id των υπόλοιπων κομματιών. Τα αποτελέσματα ήταν ιδιαίτερα καλά αφού καταφέραμε να βρούμε τα Spotify id για ακόμη 15930 κομμάτια. Έτσι καταλήξαμε να έχουμε το id για 17425 κομμάτια από τα 22.600 (Περίπου δηλαδή το 76.5%). Δυστυχώς για τα υπόλοιπα κομμάτια δεν μπορούσαμε να έχουμε το id γιατί δεν ήταν καταχωρημένα στην βάση δεδομένων του spotify. Τα δεδομένα αποθηκεύτηκαν σε μορφή csv όπου αποτελεί έναν απλό και διαδεδομένο τύπο αρχείου για προβλήματα ανάλυσης και πρόβλεψης δεδομένων. Έτσι καταλήξαμε να έχουμε ένα αρχείο csv με όλα τα διάσημα κομμάτια των πηγών από το 1958, μαζί με τον καλλιτέχνη τους και το μοναδικό Spotify id με το οποίο θα συλλέξουμε τα μουσικά χαρακτηριστικά τους. Στην συνέχεια χρησιμοποιώντας τις βιβλιοθήκες NumPy, pandas και Matplotlib προχωρήσαμε στην περαιτέρω επεξεργασία των δεδομένων.

Δ'.4 Spotify audio features.

Έπειτα αποκτήσαμε πρόσβαση στο api του spotify. Το spotify παρέχει δωρεάν πρόσβαση στο api του και χρειάζεται μόνο μια απλή εγγραφή. Μέσω αυτού μπορούμε να αντλήσουμε μια ποικιλία δεδομένων, όπως είναι στατιστικά δεδομένα χρηστών ή δεδομένα για καλλιτέχνες και μουσικά κομμάτια.(Spotify Web API,2021) Εμείς αντλήσαμε τα μουσικά χαρακτηριστικά (audio features) που υπολογίζει το spotify μετά από ανάλυση που κάνει στα κομμάτια. Τα βασικά audio features που παρέχει το spotify είναι τα εξής:

- duration_ms(Η διάρκεια σε milliseconds.)
- energy(Η ενέργεια είναι ένα μέτρο από 0,0 έως 1,0 και αντιπροσωπεύει ένα αντιληπτικό μέτρο έντασης και δραστηριότητας. Συνήθως, αισθανόμαστε τα κομμάτια με μεγάλη τιμή στο energy ως γρήγορα, δυνατά και θορυβώδη.)
- Key(Το κλειδί του κομματιού.)
- Mode(Ο μουσικός τρόπος. Παίρνει 2 τιμές 1 για 'ελάσσονες κλίμακες' και 2 για 'μείζονες κλίμακες'.)
- Time_signature(Η μουσική υπογραφή χρόνου του κομματιού.)
- Acousticness(Αυτή η τιμή περιγράφει πόσο ακουστικό είναι ένα τραγούδι, δηλαδή πόσο χρησιμοποιεί αποκλειστικά ή κυρίως όργανα που παράγουν ήχο μέσω ακουστικών μέσων. Μια τιμή 1,0 σημαίνει ότι το κομμάτι είναι πολύ πιθανό να είναι ακουστικό.)
- Danceability(Περιγράφει το πόσο κατάλληλο είναι ένα κομμάτι για χορό βάσει ενός συνδυασμού μουσικών στοιχείων, όπως το τέμπο, η σταθερότητα του ρυθμού, η δύναμη του ρυθμού και η συνολική κανονικότητα. Μια τιμή κοντά στο 0,0 σημαίνει ότι το κομμάτι δεν είναι τόσο κατάλληλο για χορό ενώ μια τιμή κοντά στο 1,0 δηλώνει ότι το κομμάτι είναι πιο κατάλληλο για χορό.)
- Instrumentalness(Αυτή η τιμή αντιπροσωπεύει την ποσότητα των φωνητικών στο τραγούδι. Όσο πιο κοντά είναι στο 1,0, τόσο πιο λίγα φωνητικά υπάρχουν στο τραγούδι.)
- Liveness(Αυτή η τιμή περιγράφει την πιθανότητα το κομμάτι να ηχογραφήθηκε με ζωντανό κοινό. Οι υψηλότερες τιμές Liveness δηλώνουν αυξημένη πιθανότητα το κομμάτι να εκτελέστηκε ζωντανά.)

- Loudness(Η συνολική ένταση ενός κομματιού σε ντεσιμπέλ (dB). Ο μέσος όρος των τιμών είναι σε ολόκληρο το κομμάτι. Το τυπικό εύρος τιμών είναι μεταξύ -60 και 0 db.)
- Speechiness(Ανιχνεύει την παρουσία προφορικών λέξεων σε ένα κομμάτι. Όσο πιο πολύ μοιάζει το κομμάτι με ομιλία (π.χ. ομιλία, βιβλίο ήχου, ποίηση), τόσο πιο κοντά είναι η τιμή του στο 1,0. Εάν σε ένα κομμάτι η τιμή είναι πάνω από 0,66, πιθανότατα είναι φτιαγμένο από προφορικές λέξεις, μια βαθμολογία μεταξύ 0,33 και 0,66 σημαίνει ότι είναι ένα κομμάτι που μπορεί να περιέχει μουσική και λέξεις και μια βαθμολογία κάτω από 0,33 σημαίνει ότι το κομμάτι δεν έχει ομιλία.)
- Valence(Ένα μέτρο από 0,0 έως 1,0 που περιγράφει τη μουσική θετικότητα που μεταδίδεται από ένα κομμάτι. Τα κομμάτια με με υψηλό Valence είναι πιο θετικά (π.χ. χαρούμενα, ευφορικά), ενώ τα κομμάτια με χαμηλό Valence ακούγονται πιο αρνητικά (π.χ. λυπημένος, κατάθλιψη, θυμωμένος.)
- Tempo(Ο συνολικός εκτιμώμενος ρυθμός ενός κομματιού σε ρυθμούς ανά λεπτό (BPM). Στη μουσική ορολογία, το tempo είναι η ταχύτητα ή ο ρυθμός ενός κομματιού και προέρχεται άμεσα από τη μέση διάρκεια του ρυθμού.)

Δ'.5 Σετ δεδομένων

Έτσι με τις κατάλληλες εντολές, εξήχθησαν για κάθε κομμάτι τα μουσικά χαρακτηριστικά και όλα μαζί αποθηκεύτηκαν σε ένα αρχείο csv. Χρησιμοποιήσαμε το `spotipy`⁸, για να διαχειριστούμε το Api του spotify. Παρακάτω εμφανίζεται ο κώδικας που κάνει την αρχική ρύθμιση.

```

1 # Importing libraries and setting up spotify api
2 import pandas as mp, csv, spotipy
3 from spotipy.oauth2 import SpotifyClientCredentials
4 client = SpotifyClientCredentials( '***', '***' )
5 sp = spotipy.Spotify(client_credentials_manager=client)

```

Έπειτα εμφανίζεται ο κώδικας μέσα στον οποίο χρησιμοποιήσαμε το `spotipy` για να εξάγουμε τα μουσικά χαρακτηριστικά για τα κομμάτια. Ο κώδικας στο τέλος αποθηκεύει ένα νέο csv αρχείο με τα μουσικά χαρακτηριστικά όλων των διάσημων κομματιών.

⁸spotipy.readthedocs.io/en/2.18.0/

```

1 # creating new csv file for track features
2 with open('sfeatures.csv', 'w', newline='') as f_handle:
3     writer = csv.writer(f_handle)
4     # Add the header/column names
5     header = ['duration_ms', 'energy', 'key', 'mode',
6             'time_signature', 'acousticness', 'danceability',
7             'instrumentalness', 'liveness',
8             'loudness', 'speechiness', 'valence', 'tempo']
9     writer.writerow(header)
10    # Iterate over 'data' and write to the csv file
11    for row in dtf["track_id"]:
12        try:
13            track= sp.audio_features(tracks=[row])
14            features = [track[0]['duration_ms'],
15                    track[0]['energy'],
16                    track[0]['key'], track[0]['mode'],
17                    track[0]['time_signature'],
18                    track[0]['acousticness'],
19                    track[0]['danceability'],
20                    track[0]['instrumentalness'],
21                    track[0]['liveness'], track[0]['loudness'],
22                    track[0]['speechiness'], track[0]['valence'],
23                    track[0]['tempo']]
24            writer.writerow(features)
25        except:
26            print("no features for "+str(row))
27            features = [0,0,0,0,0,0,0,0,0,0,0,0,0,0]
28            writer.writerow(features)

```

Επίσης προστέθηκε η στήλη "On_chart" η οποία δηλώνει αν το κομμάτι υπήρξε σε κάποιο chart ποτέ ή όχι, με την τιμή 1 να δηλώνει ότι υπήρξε ενώ την τιμή 0 να δηλώνει ότι δεν υπήρξε. Συνδυάζοντας όλα τα δεδομένα καταλήξαμε με ένα σετ δεδομένων που σε κάθε γραμμή του περιέχει ένα διάσημο κομμάτι, τον καλλιτέχνη του κομματιού, το μοναδικό id του ,τα μουσικά

χαρακτηριστικά που εξήχθησαν από το api του spotify καθώς και στήλη position. Προφανώς εδώ η στήλη "position" είχε παντού την τιμή ένα αφού όλα τα κομμάτια ήταν διάσημα. Έτσι είχαμε ένα ολοκληρωμένο σετ δεδομένων σε μορφή csv αρχείου με όλα τα διάσημα κομμάτια. Συνολικά είχε 17425 γραμμές και 17 στήλες που σημαίνει ότι καταλήξαμε με 17430 μοναδικά διάσημα μουσικά κομμάτια. Στην συνέχεια μέσω της ιστοσελίδας [Kaggle.com](https://www.kaggle.com/yamaerenay/spotify-dataset-19212020-160k-tracks)⁹ τραβήξαμε μια λίστα από κομμάτια που δεν υπήρξαν ποτέ σε οποιαδήποτε λίστα επιτυχιών. Μέσα στην λίστα υπήρχε και η στήλη για το μοναδικό spotify id για κάθε κομμάτι. Έτσι ακολουθώντας ξανά τα ίδια βήματα που χρησιμοποιήσαμε προηγουμένως για τα διάσημα κομμάτια δημιουργήσαμε ένα αντίστοιχο σετ δεδομένων σε μορφή csv με μη διάσημα κομμάτια. Προφανώς εδώ η στήλη "On_chart" είχε παντού την τιμή μηδέν αφού όλα τα κομμάτια ήταν μη διάσημα. Συνολικά από το δεύτερο σετ δεδομένων, και με σκοπό να αποφύγουμε το πρόβλημα ανισορροπίας των δεδομένων κρατήσαμε μόνο 17315 γραμμές δηλαδή 17315 μη διάσημα μουσικά κομμάτια. Το πρόβλημα ανισορροπίας των δεδομένων αναφέρεται στα προβλήματα που αντιμετωπίζουν οι αλγόριθμοι μηχανικής μάθησης όταν στο σετ δεδομένων μια κλάση αντιπροσωπεύεται από μεγάλο αριθμό παραδειγμάτων, ενώ η άλλη αντιπροσωπεύεται μόνο με μερικά. Το πρόβλημα της ανισορροπίας των δεδομένων είναι ζωτικής σημασίας, καθώς εμφανίζεται σε σετ δεδομένων που αφορούν τομείς μεγάλης περιβαλλοντικής, ζωτικής ή εμπορικής σημασίας και σε ορισμένες περιπτώσεις, αποδείχθηκε να προκαλεί σημαντικά προβλήματα στην απόδοση που επιτυγχάνεται με τυπικές μεθόδους μηχανικής μάθησης που προϋποθέτουν μια ισορροπημένη κατανομή των τάξεων. (Japkowicz and Stephen, 2001) Τέλος τα δυο αρχεία ενώθηκαν και καταλήξαμε σε ένα σετ δεδομένων με 34740 γραμμές και 17 στήλες, δηλαδή 17425 διάσημα κομμάτια και 17315 μη διάσημα κομμάτια.

Δ'.6 Καθαρισμός και ανάλυση δεδομένων

Πριν την πρόβλεψη των δεδομένων προηγήθηκε ο καθαρισμός και η ανάλυση των δεδομένων. Η ανάλυση των δεδομένων βοηθάει να κατανοήσουμε καλύτερα τα δεδομένα μας και να προβούμε σε καλύτερη προεπεξεργασία τους. Επίσης ο καθαρισμός των δεδομένων μας επιτρέπει να εντοπίσουμε προβλήματα στο σετ δεδομένων μας όπως κενές ή ελλιπείς τιμές ή λάθος τύπου δεδομένα και οδηγεί σε καλύτερα αποτελέσματα στην πρόβλεψη των δεδομένων. (Han, 2011)

⁹<https://www.kaggle.com/yamaerenay/spotify-dataset-19212020-160k-tracks>

Δ'.7 Καθαρισμός δεδομένων

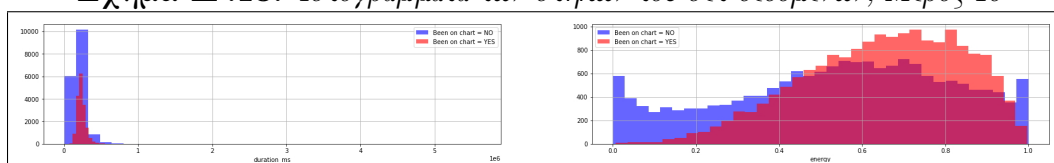
Αρχικά χρησιμοποιώντας τις βιβλιοθήκες "NumPy" και "Pandas" ελέγξαμε όλο το σετ δεδομένων για ελλιπείς τιμές ή για τιμές NULL. Τελικά δεν παρατηρήθηκε κάποιο πρόβλημα στο σετ δεδομένων, το οποίο ήταν και το πιθανότερο να συμβεί αν αναλογιστούμε ότι τα δεδομένα συλλέχθηκαν με έναν ημιαυτόματο τρόπο και ότι πριν την συλλογή τους προηγήθηκαν ήδη και αρκετοί άλλοι έλεγχοι. Επίσης το σετ δεδομένων ελέγχθηκε και για λάθος τιμές, δηλαδή τιμές σε οποιαδήποτε στήλη που δεν συμφωνούν με τον τύπο τιμών της συγκεκριμένης στήλης, αλλά ούτε εκεί παρουσιάστηκε κάποιο πρόβλημα.

Δ'.8 Ανάλυση δεδομένων

Έπειτα χρησιμοποιώντας τις βιβλιοθήκες NumPy, pandas και Matplotlib προχωρήσαμε στην ανάλυση του σετ δεδομένων. Αρχικά έγιναν κάποιες βασικές αναλύσεις που θα μας βοηθούσαν στην επιλογή των χαρακτηριστικών που θα ορίσουν το μοντέλο πρόβλεψης που θα δημιουργηθεί. Στην συνέχεια αναλύσαμε περαιτέρω τα δεδομένα με πιο εξειδικευμένα διαγράμματα έτσι ώστε απαντήσουμε και σε κάποια συμπληρωτικά ερωτήματα. Αρχικά δημιουργήθηκε ένα σύνολο από ιστογράμματα για να οπτικοποιήσουμε την κατανομή των διαφόρων τιμών στο σετ δεδομένων. Όπως φαίνεται και παρακάτω στις εικόνες Δ'.13, Δ'.14, Δ'.15, Δ'.16, Δ'.17, Δ'.18 και Δ'.19 το κάθε διάγραμμα περιέχει τις τιμές της κάθε στήλης ως προς το πόσες φορές εμφανίζεται η κάθε τιμή. Το μπλε χρώμα στο διάγραμμα δηλώνει ποιες τιμές εμφανίζονται σε θετικά δείγματα, δηλαδή σε δείγματα που τα κομμάτια είναι διάσημα. Το κόκκινο χρώμα στο διάγραμμα δηλώνει ποιες τιμές εμφανίζονται σε αρνητικά δείγματα, δηλαδή σε δείγματα που τα κομμάτια είναι μη διάσημα. Τέλος το μοβ χρώμα δηλώνει ποιες τιμές εμφανίζονται σε θετικά δείγματα αλλά και σε αρνητικά δηλαδή είναι τιμές και διάσημων και μη διάσημων κομματιών.

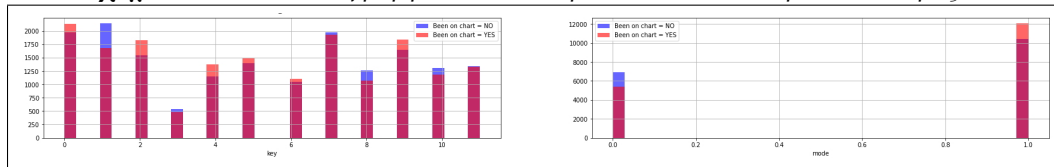
Στα παραπάνω διαγράμματα μπορούν να γίνουν κάποιες βασικές παρατηρήσεις για τα

Σχήμα Δ'.13: Ιστογράμματα των στηλών του σετ δεδομένων, Μέρος 1ο

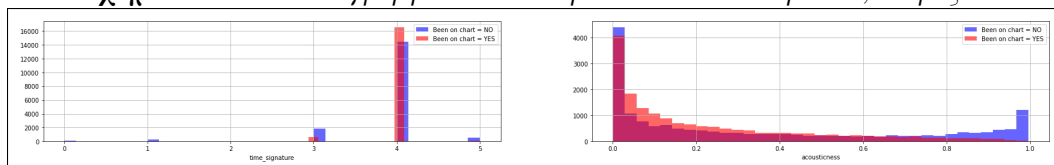


δεδομένα, όπως για παράδειγμα ότι το loudness τείνει να είναι μεγαλύτερο σε διάσημα κομμάτια ή ότι τα διάσημα κομμάτια τείνουν να έχουν μεγαλύτερη χρονική διάρκεια σε σχέση με

Σχήμα Δ'.14: Ιστογράμματα των στηλών του σετ δεδομένων, Μέρος 2ο.

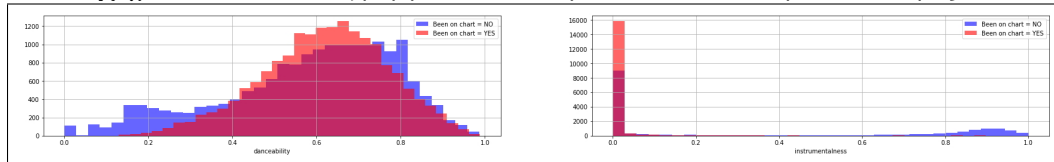


Σχήμα Δ'.15: Ιστογράμματα των στηλών του σετ δεδομένων, Μέρος 3ο.

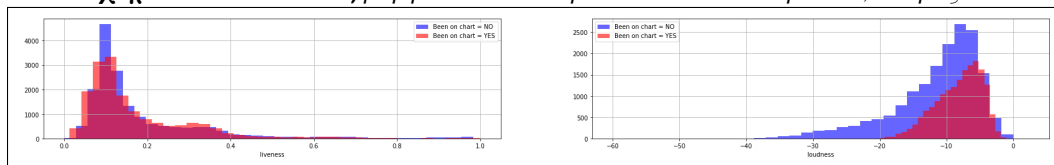


τα μη διάσημα. Έπειτα δημιουργήσαμε ένα Correlation matrix ή όπως συνήθως αναφέρεται Correlation heatmap το οποίο παρουσιάζεται στην εικόνα Δ'.20. Είναι ένα πολύ πρακτικό διάγραμμα για να παρατηρούμε την συσχέτιση που έχουν οι στήλες και κατ' επέκταση οι τιμές των στηλών μεταξύ τους. (Tim Bock, 2021) Το διάγραμμα είναι πολύ απλό στην κατανόησή του. Όσο πιο λευκό είναι το κάθε κελί τόσο πιο αντιστρόφως ανάλογες είναι οι τιμές και κατ' επέκταση οι στήλες που αντιπροσωπεύει ενώ από την άλλη όσο πιο κόκκινο είναι το κουτί τόσο πιο ανάλογες είναι οι τιμές και κατ' επέκταση οι στήλες που αντιπροσωπεύει. Από το παραπάνω διάγραμμα παρατηρούμε ότι υπάρχει μια αξιόλογη (αντιστρόφως ανάλογη) συσχέτιση μεταξύ του instrumentality και του On_chart. Το ίδιο επίσης παρατηρούμε και με το valence (ανάλογη συσχέτιση), που σημαίνει ότι πιθανότατα και τα δύο χαρακτηριστικά να παίζουν μεγάλο ρόλο στον καθορισμό ενός κομματιού ως διάσημου ή μη διάσημου. Έπειτα δημιουργήσαμε ένα διάγραμμα για να εμφανίσουμε τους καλλιτέχνες που εμφανίζονται περισσότερο στο σετ δεδομένων μας. Το διάγραμμα εμφανίζεται στην εικόνα Δ'.21 παρουσιάζει τους 15 καλλιτέχνες που εμφανίζονται περισσότερο στο σετ δεδομένων όσον αφορά μόνο τα διάσημα κομμάτια. Παρατηρούμε ότι τα περισσότερα ονόματα είναι γνωστά όπως είναι άλλωστε και φυσικό αφού πρόκειται μόνο για τα διάσημα κομμάτια.

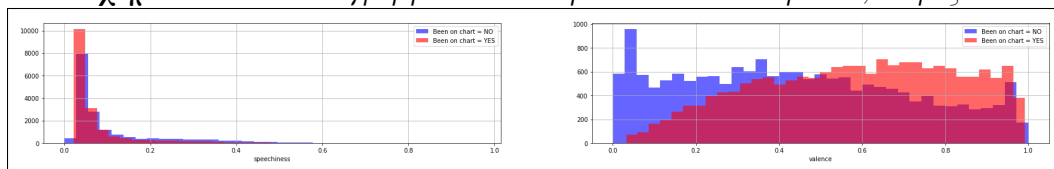
Σχήμα Δ'.16: Ιστογράμματα των στηλών του σετ δεδομένων, Μέρος 4ο.



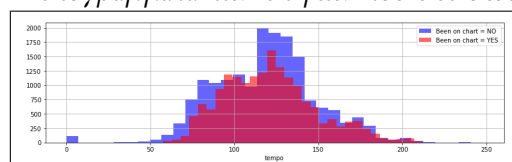
Σχήμα Δ'.17: Ιστογράμματα των στηλών του σετ δεδομένων, Μέρος 5ο.



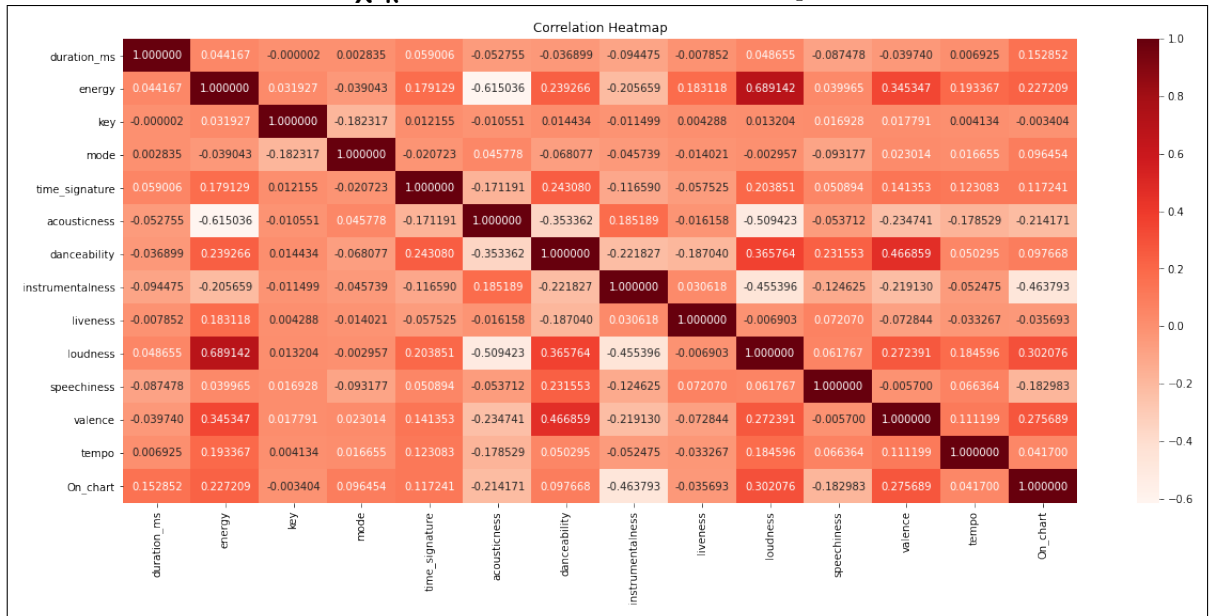
Σχήμα Δ'.18: Ιστογράμματα των στηλών του σετ δεδομένων, Μέρος 6ο.



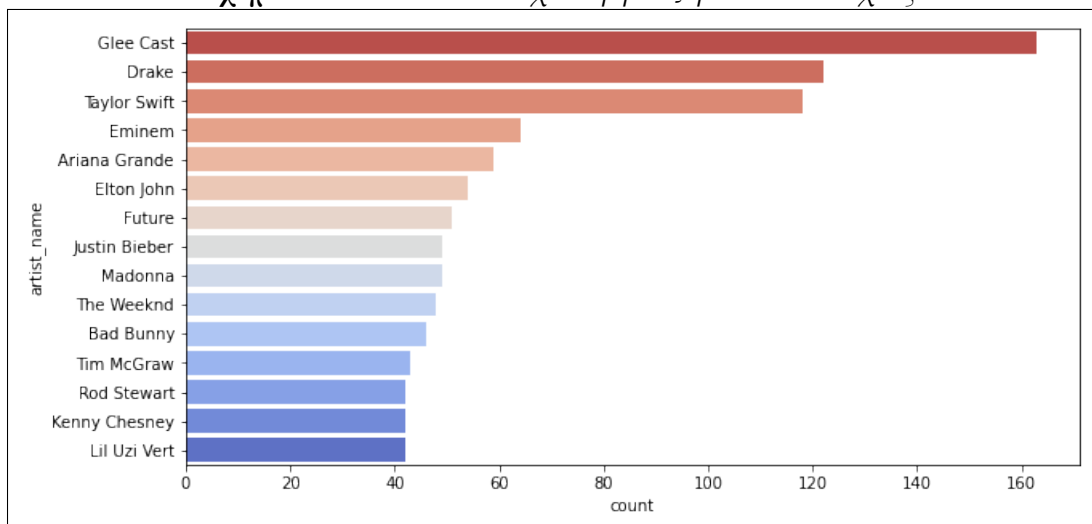
Σχήμα Δ'.19: Ιστογράμματα των στηλών του σετ δεδομένων, Μέρος 7ο.



Σχήμα Δ΄.20: Correlation heatmap.



Σχήμα Δ΄.21: Οι πιο συχνά εμφανιζόμενοι καλλιτέχνες.



Δ'.9 Επεξεργασία / Μετασχηματισμός των δεδομένων

Ο μετασχηματισμός των δεδομένων είναι ένα στάδιο προεπεξεργασίας, κατά το οποίο τα δεδομένα μετασχηματίζονται ή ενοποιούνται έτσι ώστε η διαδικασία εξόρυξης να είναι πιο αποτελεσματική, και τα μοτίβα που ανακαλύπτονται να είναι πιο εύκολα κατανοητά. Τα δεδομένα μετατρέπονται ή ενοποιούνται σε μορφές κατάλληλες για εξόρυξη. Μερικές στρατηγικές για τον μετασχηματισμό δεδομένων είναι η εξομάλυνση, η κανονικοποίηση και η διακριτοποίηση των δεδομένων (Han, 2011). Πριν την πρόβλεψη προηγήθηκε το τελευταίο στάδιο της επεξεργασίας των δεδομένων. Όπως προαναφέρθηκε τα δεδομένα είχαν ήδη καθαριστεί από ελλείψεις ή λάθος τιμές οπότε προχωρήσαμε στο στάδιο του μετασχηματισμού και συγκεκριμένα στην κανονικοποίηση των δεδομένων μιας και αυτή έδειξε να είναι η τεχνική που βελτίωνε καλύτερα τα αποτελέσματά μας. Η μονάδα μέτρησης που χρησιμοποιείται σε κάποια στήλη του σετ δεδομένων μπορεί να επηρεάσει την ανάλυση δεδομένων. Για παράδειγμα, η αλλαγή των μονάδων μέτρησης από μέτρα σε ίντσες για ύψος, ή από κιλά σε λίβρες για βάρος, μπορεί να οδηγήσει σε πολύ διαφορετικά αποτελέσματα. Σε γενικές γραμμές, εκφράζοντας ένα χαρακτηριστικό σε μικρότερες μονάδες μέτρησης οδηγούμαστε σε ένα μεγαλύτερο εύρος τιμών για αυτό το χαρακτηριστικό, που τείνουν να δίνουν στο χαρακτηριστικό μεγαλύτερη επίδραση ή αλλιώς 'βάρος'. Για να αποφύγουμε την εξάρτηση από την επιλογή των μονάδων μέτρησης, τα δεδομένα πρέπει να είναι κανονικοποιημένα ή τυποποιημένα. Η ομαλοποίηση των δεδομένων προσπαθεί να δώσει σε όλα τα χαρακτηριστικά ίσο βάρος. Η κανονικοποίηση είναι ιδιαίτερα χρήσιμη για αλγόριθμους ταξινόμησης όπως τα νευρωνικά δίκτυα ή αλγόριθμους που περιλαμβάνουν μετρήσεις αποστάσεων όπως η ταξινόμηση k-NN, κυρίως επειδή μειώνει τον χρόνο που χρειάζεται για την 'μάθηση' των δεδομένων. (Han, 2011) Σύμφωνα με τον Han (2011) υπάρχουν διαφορετικοί τρόποι κανονικοποίησης των δεδομένων. Οι πιο βασικοί είναι οι ακόλουθοι:

- Κανονικοποίηση min-max: οι τιμές κανονικοποιούνται, ώστε το εύρος τους να ανήκει σε ένα νέο, περιορισμένο εύρος, π.χ. [-1, 1], [0, 1] κοκ. Η νέα τιμή του χαρακτηριστικού σε κάθε πλειάδα υπολογίζεται χρησιμοποιώντας τον τύπο

$$v_n = \frac{v - \min}{\max - \min} (\max_n - \min_n) + \min_n$$

- Κανονικοποίηση z-score: οι τιμές κανονικοποιούνται με χρήση της μέσης τιμής και της τυπικής απόκλισης, έτσι ώστε τα δεδομένα να έχουν μέση τιμή 0 και τυπική απόκλιση 1. Η κανονικοποίηση γίνεται με τον παρακάτω τύπο, όπου m είναι η μέση τιμή του

χαρακτηριστικού και s η τυπική του απόκλιση.

$$v_n = \frac{v - m}{s}$$

- Κανονικοποίηση με δεκαδική κλίμακα: οι τιμές κανονικοποιούνται με τάξεις μεγέθους του 10. Η κανονικοποίηση γίνεται με τον τύπο:

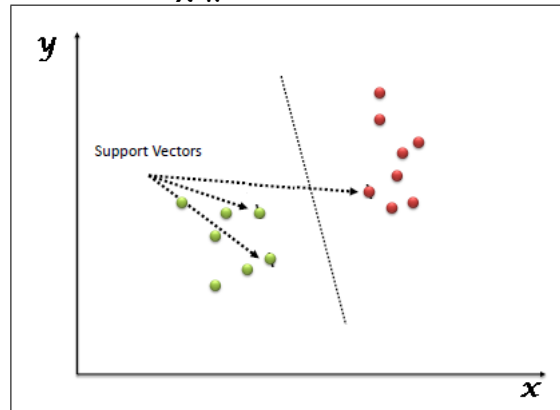
$$v_n = \frac{v}{10^j}$$

Στην παρούσα έρευνα λόγω του μεγέθους των δεδομένων αλλά και της μικρής υπολογιστικής ισχύς η απόπειρα δημιουργίας μοντέλου πρόβλεψης των δεδομένων χωρίς πρώτα να κανονικοποιήσουμε τα δεδομένα έκανε την εκπαίδευση του μοντέλου των αλγόριθμων k-NN και SVM να διαρκεί για πολύ μεγάλο χρόνο, σε σημείο που δεν ήταν δυνατή η καταγραφή του και η εκτέλεση του. Μετά την κανονικοποίηση η εκτέλεση της εκπαίδευσης διήρκεσε μόλις 4 λεπτά και στους δύο αλγόριθμους. Στην περίπτωση του αλγόριθμου Neural network η κανονικοποίηση μείωσε τον χρόνο εκτέλεσης από 3 ώρες σε 30 λεπτά. Από την άλλη η κανονικοποίηση δεν είχε επίδραση στον αλγόριθμο Random Forest (RF) του οποίου η εκτέλεση διήρκεσε περίπου 3 λεπτά και στις δύο περιπτώσεις. Παρατηρήσαμε ότι τρόπος κανονικοποίησης που λειτουργούσε καλύτερα με τα αριθμητικά δεδομένα μας αλλά και τους αλγορίθμους πρόβλεψης ήταν η κανονικοποίηση min-max. Η κανονικοποίηση μείωσε τον χρόνο που χρειαζόταν για να δημιουργηθεί το μοντέλο και εν συνέπεια μας έδωσε την ελευθερία να κάνουμε πολλές και διαφορετικές αλλαγές στις παραμέτρους των αλγορίθμων μέχρι να βρούμε τις βέλτιστες ρυθμίσεις για τον κάθε αλγόριθμο πρόβλεψης. Οι πιο σημαντικές αλλαγές από αυτές θα παρουσιαστούν εν συνέχεια. Η διαδικασία της αυτόματης ή μη, παραμετροποίησης των αλγορίθμων πρόβλεψης ονομάζεται Hyperparameter optimization (M Feurer, F Hutter, 2019).

Δ'.10 Πρόβλεψη των δεδομένων

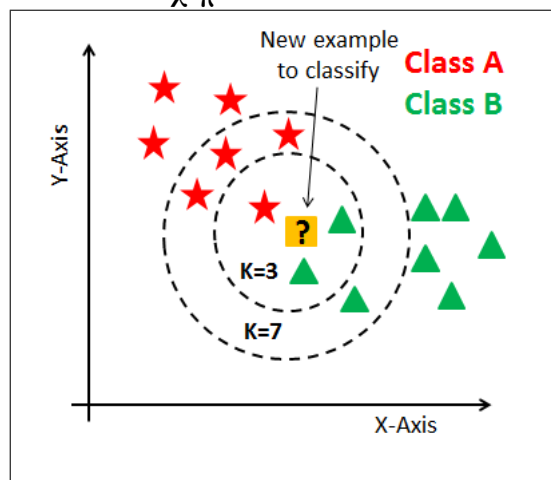
Αφού τα δεδομένα είχαν καθαριστεί όπως έπρεπε και είχαν υποστεί την σωστή προ-επεξεργασία είμαστε πια έτοιμοι να προχωρήσουμε στην εκπαίδευση του μοντέλου και στην πρόβλεψη των δεδομένων. Έγιναν δοκιμές με διάφορους αλγόριθμους πρόβλεψης αλλά εδώ θα αναλύσουμε μόνο τους αλγόριθμους που επέστρεψαν τα καλύτερα μοντέλα πρόβλεψης των δεδομένων. Υπάρχουν δύο κύριοι τύποι μηχανικής μάθησης η επιβλεπόμενη και η μη-επιβλεπόμενη μάθηση. Στην επιβλεπόμενη μάθηση ένας αλγόριθμος επιβλεπόμενης μηχανικής μάθησης (σε αντίθεση με έναν αλγόριθμο μη επιβλεπόμενης μηχανικής μάθησης) είναι ένας αλγόριθμος που βασίζεται σε επισημασμένα δεδομένα εισόδου για να εκπαιδεύσει ένα μοντέλο που παράγει μια κατάλληλη έξοδο όταν του δίνονται νέα μη επισημασμένα δεδομένα (Han,2011). Γνωρίζουμε δηλαδή όλες τις πιθανές εξόδους. Στην περίπτωση μας η μάθηση θα είναι επιβλεπόμενη αφού η επισημάνση είναι η στήλη On_chart η οποία επισημαίνει αν το κάθε δείγμα προέρχεται από κάποιο μουσικό κομμάτι που ήταν ή δεν ήταν ποτέ σε κάποια λίστα δημοτικότητας. Έτσι ο αλγόριθμός με την σειρά του 'μαθαίνοντας' από τα επισημασμένα δεδομένα θα μας δημιουργήσει ένα μοντέλο το οποίο και αυτό με την σειρά του θα αναγνωρίζει νέα δείγματα τα οποία δεν θα έχουν επισημανθεί ακόμα. Αντίθετα στην μη επιβλεπόμενη μάθηση έχουμε μη επισημασμένα δεδομένα και στόχος είναι η μοντελοποίηση της υποκείμενης δομής ή κατανομής στα δεδομένα, προκειμένου να μάθουμε περισσότερα για τα δεδομένα. Σε αντίθεση με την μάθηση με επίβλεψη δεν υπάρχουν σωστές ή συγκεκριμένες εξοδοι. Οι αλγόριθμοι αφήνονται στην τύχη τους για να ανακαλύψουν και να παρουσιάσουν κάποια ενδιαφέρουσα δομή στα δεδομένα (Han,2011). Παρακάτω θα αναλύσουμε τους αλγόριθμους επιβλεπόμενης μηχανικής μάθησης που χρησιμοποιήσαμε για να δημιουργήσουμε το μοντέλο μας. Οι Μηχανές Διανυσμάτων Υποστήριξης Support Vector Machines(SVM) είναι μια συλλογή αλγορίθμων επιβλεπόμενης μάθησης που χρησιμοποιούνται ευρύτατα σε προβλήματα κατάταξης. Ο αλγόριθμος SVM έχει το πλεονέκτημα να χειρίζεται πολύ καλά μεγάλο πλήθος χαρακτηριστικών και παρουσιάζει υψηλή απόδοση στην κατηγοριοποίηση δεδομένων. Είναι ικανός να κατασκευάζει μοντέλα αρκετά πολύπλοκα για να επιλύει δύσκολα προβλήματα του πραγματικού κόσμου. Στον αλγόριθμο SVM, απεικονίζουμε κάθε στοιχείο δεδομένων ως ένα σημείο στον n -διάστατο χώρο (όπου n είναι ο αριθμός των χαρακτηριστικών που έχουμε) με την τιμή κάθε χαρακτηριστικού να είναι η τιμή μιας συγκεκριμένης συντεταγμένης. Στη συνέχεια, πραγματοποιούμε ταξινόμηση βρίσκοντας το υπερεπίπεδο που διαφοροποιεί πολύ καλά τις δύο κλάσεις.(Han,2011) Ο αλγόριθμος k-Nearest neighbors (k-NN) είναι ένας τύπος αλγορίθμου μηχανικής μάθησης με επίβλεψη, ο οποίος μπορεί να χρησιμοποιηθεί τόσο για προβλήματα

Σχήμα Δ'.22: SVM



ταξινόμησης όσο και για προβλήματα πρόβλεψης παλινδρόμησης. Ωστόσο, χρησιμοποιείται κυρίως για προβλήματα πρόβλεψης ταξινόμησης. Οι ακόλουθες δύο ιδιότητες θα μπορούσαν να ορίσουν καλά τον k-NN. Ο k-NN δεν έχει εξειδικευμένη φάση εκπαίδευσης και χρησιμοποιεί όλα τα δεδομένα για ταξινόμηση. Είναι επίσης ένας μη παραμετρικός αλγόριθμος μάθησης επειδή δεν υποθέτει τίποτα για τα υποκείμενα δεδομένα. Ο αλγόριθμος k-NN υποθέτει ότι παρόμοια πράγματα υπάρχουν σε κοντινή απόσταση. Με άλλα λόγια, τα παρόμοια πράγματα βρίσκονται κοντά το ένα στο άλλο. Η αρχή πίσω από τον k-NN είναι η εύρεση ενός προκαθορισμένου αριθμού δειγμάτων(k) εκπαίδευσης που βρίσκονται σε κοντινότερη απόσταση από το νέο σημείο και η πρόβλεψη της ετικέτας (στην περίπτωσή μας αν είναι διάστημα ή όχι) από αυτά. (Han, 2011) Για την παραμετροποίηση του k-NN χρησιμοποιήσαμε τον παρακάτω

Σχήμα Δ'.23: k-NN



κώδικα ο οποίος δοκιμάζει διαφορετικές τιμές του k μέχρι να βρει το βέλτιστο το οποίο και χρησιμοποιήσαμε τελικά.

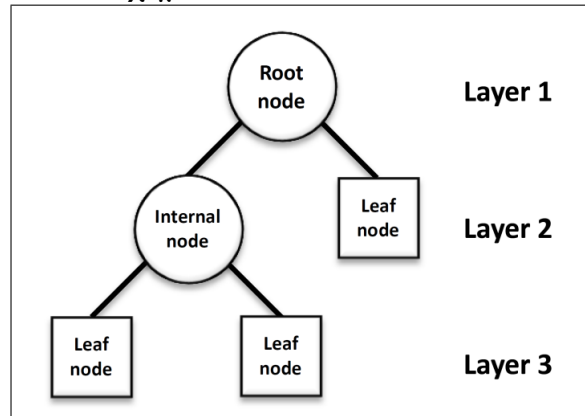
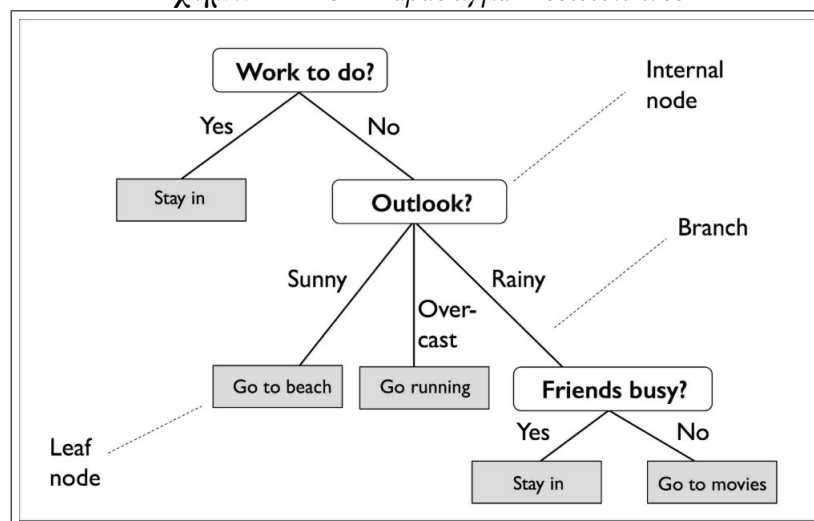
```

1 temp = 0
2 tempi = 0
3 df = pd.DataFrame(columns = ["k", 'score'])
4 for i in range (20,100):
5     model=KNeighborsClassifier(n_neighbors=i)
6     dtf1=dtf.drop('0n_chart',1)
7     dtf2=dtf['0n_chart']
8     cv_r2_scores_rf=cross_val_score(model,dtf1,dtf2,cv=10)
9     mean=np.mean(cv_r2_scores_rf)
10    df.loc[i]=i
11    df.loc[i,'score'] = mean
12    if mean > temp:
13        temp = mean
14        tempi = i
15 print ("biggest success:",temp," ,k=",tempi)

```

Οι αλγόριθμος Random Forest (RF) είναι μία από τις πιο δημοφιλείς μεθόδους που χρησιμοποιούνται στην ταξινόμηση. Ο RF είναι ένας από τους καλύτερους αλγορίθμους ταξινόμησης γιατί μπορεί να ταξινομήσει μεγάλα δεδομένα με ακρίβεια. Τα μοντέλα που παράγει ο RF είναι στην ουσία μια συλλογή από παραδοσιακά δέντρα αποφάσεων (Decision trees). Ο RF εκτός των άλλων μπορεί να 'αποφύγει' το πρόβλημα της υπερβολικής προσαρμογής του μοντέλου που εμφανίζεται στα παραδοσιακά δέντρα αποφάσεων. Ένα δέντρο αποφάσεων είναι μια δενδροειδής δομή που μοιάζει με διάγραμμα ροής, όπου ένας εσωτερικός κόμβος αντιπροσωπεύει ένα χαρακτηριστικό, το κλαδί αντιπροσωπεύει έναν κανόνα απόφασης και κάθε κόμβος φύλλου αντιπροσωπεύει το αποτέλεσμα. Ο κόμβος της κορυφής σε ένα δέντρο αποφάσεων είναι γνωστός ως κόμβος ρίζας. Τα δέντρα αποφάσεων είναι ιδιαίτερα απλά στην κατανόηση τους γιατί μιμούνται σε μεγάλο βαθμό την ανθρώπινη σκέψη όπως φαίνεται και στην εικόνα 18. Ο RF δημιουργεί ένα σύνολο από δέντρα αποφάσεων για τα δεδομένα και παίρνει το μέσο όρο για να βελτιώσει την ακρίβεια πρόβλεψης του συγκεκριμένου συνόλου δεδομένων. Αντί να βασίζεται δηλαδή σε μόνο ένα δέντρο απόφασης, ο RF λαμβάνει την πρόβλεψη από κάθε δέντρο και με βάση τις πλειοψηφικές ψήφους των προβλέψεων, προβλέπει την τελική έξοδο. (Han,2011)

Τα Τεχνητά Νευρωνικά Δίκτυα (ΤΝΔ) αποτελούν μια προσπάθεια προσέγγιση-μίμηση της

Σχήμα Δ'.24: *Decision trees.*Σχήμα Δ'.25: *Παράδειγμα Decision tree.*

λειτουργίας του ανθρώπινου εγκεφάλου. Το τεχνητό νευρωνικό δίκτυο είναι ένα μοντέλο επεξεργασίας πληροφοριών εμπνευσμένο από τον τρόπο με τον οποίο βιολογικά νευρικά συστήματα, όπως ο εγκέφαλος, επεξεργάζονται πληροφορίες. Τα τεχνητά νευρωνικά δίκτυα είναι μια συλλογή μεγάλου αριθμού συνδεδεμένων στοιχείων επεξεργασίας που ονομάζονται νευρώνες (Processing Units, PUs) οι οποίοι συνδέονται μεταξύ τους, λειτουργούν αρμονικά και είναι οργανωμένοι σε στρώματα (Layers). Κάθε PU έχει πολλές εισόδους (Inputs) αλλά μόνο μια έξοδο (Output) η οποία με την σειρά της μπορεί να αποτελέσει είσοδο για άλλες PUs. Οι συνδέσεις μεταξύ των PUs διαφέρουν μεταξύ τους και η σπουδαιότητά τους προσδιορίζεται από τον συντελεστή βάρους για κάθε σύναψη. Η συνεργασία PUs καθορίζεται από την συνάρτηση

μεταφοράς η οποία καθορίζει την κάθε έξοδο σε σχέση με τις εισόδους και τους συντελεστές βάρους. Η επεξεργασία κάθε νευρώνα καθορίζεται από την συνάρτηση ενεργοποίησης η οποία καθορίζει την έξοδο σε σχέση με τις εισόδους και τους συντελεστές βάρους (State, Vlamos 2000). Για να χρησιμοποιηθεί ένα δίκτυο πρέπει αρχικά να περάσει την διαδικασία εκπαίδευσης. Η μάθηση περιλαμβάνει αλλαγές στις συνοπτικές σχέσεις (βάρη) που περιέχονται μεταξύ των νευρώνων. Ο ρόλος των συντελεστών μάθησης μπορεί να ερμηνευτεί ως αποθήκευση γνώσης, η οποία παρέχεται στο σύστημα με την βοήθεια παραδειγμάτων κάτι που ισχύει και στα βιολογικά νευρωνικά δίκτυα. Με αυτό τον τρόπο τα ΤΝΔ μαθαίνουν το περιβάλλον τους, ή με άλλα λόγια το φυσικό μοντέλο που παρέχει τα δεδομένα. (Αντονια πλευρου, 2012).

Η αξιολόγηση της επιτυχίας των μοντέλων έγινε με την χρήση της τεχνικής k-Fold Cross Validation. Ειδικότερα μόλις τελειώσουμε με την εκπαίδευση του μοντέλου μας, δεν μπορούμε να υποθέσουμε ότι αυτό θα λειτουργήσει καλά σε δεδομένα που δεν έχει ξαναδεί, δηλαδή δεδομένα που δεν περιέχονται στο σετ δεδομένων μας. Με άλλα λόγια, δεν μπορούμε να είμαστε σίγουροι ότι το μοντέλο θα έχει την επιθυμητή ακρίβεια και επιτυχία σε ένα πραγματικό περιβάλλον. Χρειαζόμαστε κάποιου είδους διαβεβαίωση για την ακρίβεια των προβλέψεων που εξάγει το μοντέλο μας. Για να αξιολογήσουμε την απόδοση οποιουδήποτε μοντέλου μηχανικής μάθησης πρέπει να το δοκιμάσουμε σε κάποια αθέατα δεδομένα. Με βάση την απόδοση του μοντέλου σε αθέατα δεδομένα μπορούμε να πούμε αν το μοντέλο μας είναι υπό-προσαρμοσμένο, υπέρ-προσαρμοσμένο ή καλώς γενικευμένο σε σχέση με τα δεδομένα. Το k-Fold Cross Validation είναι μια από τις τεχνικές που χρησιμοποιούνται για τον έλεγχο της αποτελεσματικότητας ενός μοντέλου μηχανικής μάθησης. Είναι επίσης μια διαδικασία επαναληπτικής δειγματοληψίας που χρησιμοποιείται για την αξιολόγηση ενός μοντέλου εάν έχουμε περιορισμένα δεδομένα. Για να εκτελέσουμε ένα k-Fold Cross Validation πρέπει να κρατήσουμε στην άκρη ένα δείγμα/ένα μέρος των δεδομένων που δεν χρησιμοποιούνται για την εκπαίδευση του μοντέλου, και αργότερα να χρησιμοποιήσουμε αυτό το δείγμα για δοκιμή/επιβεβαίωση. Η τεχνική k-Fold Cross Validation είναι μια δημοφιλής και εύκολα κατανοητή τεχνική για να επικυρώσουμε το μοντέλο μας. Ειδικότερα εξασφαλίζει ότι κάθε παρατήρηση από το αρχικό σύνολο δεδομένων έχει την ευκαιρία να εμφανιστεί στο σύνολο εκπαίδευσης και στο σύνολο επικύρωσης του μοντέλου. Είναι μια από τις καλύτερες προσεγγίσεις αν έχουμε περιορισμένα δεδομένα εισόδου. (Han, 2011) Η μέθοδος ακολουθεί τα παρακάτω βήματα.

- Χωρίζουμε το σύνολο των δεδομένων τυχαία σε k μέρη (k-fold) (η τιμή του k δεν πρέπει να είναι πολύ μικρή ή πολύ υψηλή, ιδανικά επιλέγεται 5 έως 10 ανάλογα με το μέγεθος των δεδομένων). Η υψηλότερη τιμή του k οδηγεί σε λιγότερο αυστηρά αξιολογημένο

μοντέλο, ενώ η χαμηλότερη τιμή του k το αντίθετο.

- Στη συνέχεια, το μοντέλο εκπαιδεύεται χρησιμοποιώντας τα $k-1$ (k μείον 1) μέρη και επικυρώνεται χρησιμοποιώντας το εναπομένον k μέρος. Τα αποτελέσματα αποθηκεύονται.
- Η διαδικασία επαναλαμβάνεται έως ότου κάθε k μέρος χρησιμεύσει ως σύνολο δοκιμής και επικύρωσης του μοντέλου. Στη συνέχεια, ο μέσος όρος των καταγεγραμμένων αποτελεσμάτων υπολογίζεται. Ο μέσος όρος αυτός είναι η μέτρηση απόδοσης για το μοντέλο.

Έτσι γενικεύουμε την μέτρηση απόδοσης για το μοντέλο μας και είμαστε σίγουροι για το πόσο καλά θα ανταποκριθεί σε νέα σύνολα δεδομένων στο μέλλον. (Han, 2011)

Δ'.11 Αποτελέσματα

Για την αξιολόγηση του κάθε μοντέλου που παρήγαγαν οι αλγόριθμοι μας εκτός από την μέτρηση απόδοσης με την τεχνική k -Fold Cross Validation για μια πιο λεπτομερή αξιολόγηση και οπτικοποίηση της επίδοσης μπορούμε να χρησιμοποιήσουμε τον πίνακα σύγχυσης «Confusion matrix» ενός δυαδικού κατηγοριοποιητή. Ο πίνακας έχει διαστάσεις 2×2 , δηλαδή περιέχει συνολικά δύο γραμμές και δύο στήλες και περιέχει τον αριθμό των true positives, true negatives, false positives και false negatives. Κάθε στήλη αναπαριστά τον αριθμό των παραδειγμάτων στην κλάση που κατηγοριοποιήθηκαν ενώ κάθε γραμμή τον αριθμό των παραδειγμάτων στην πραγματική τους κλάση. Οι σωστές προβλέψεις βρίσκονται στην διαγώνιο του πίνακα, ενώ στα υπόλοιπα κελιά φαίνεται ο αριθμός των λανθασμένων προβλέψεων. Οπότε ιδανικά θα θέλαμε η διαγώνιος να περιέχει μεγάλους αριθμούς, ενώ τα υπόλοιπα κελιά να τείνουν όσο το δυνατόν περισσότερο στο μηδέν (Powers, 2011). Ο πίνακας φαίνεται στην εικόνα Δ'.26 Σε πολλά προβλήματα η ευστοχία μπορεί να μην αποτελεί καλή μετρική για την αξιολόγηση της κατηγοριοποίησης, επειδή μπορεί να νομίζουμε ότι ένας κατηγοριοποιητής είναι καλός, ενώ στην πραγματικότητα δεν είναι. Είναι πιθανό ένα μοντέλο με μια συγκεκριμένη τιμή ευστοχίας να είναι καλύτερο σε σύγκριση με ένα άλλο που έχει υψηλότερη τιμή ευστοχίας. Αυτό μπορεί να συμβεί στο παράδειγμα της δυαδικής κατηγοριοποίησης, όπου ο αριθμός των παραδειγμάτων που ανήκουν στη μία κλάση διαφέρει σε μεγάλο βαθμό από τον αριθμό των παραδειγμάτων που ανήκουν στην άλλη κλάση. (Han, 2011) Για τους παραπάνω λόγους, είναι

Σχήμα Δ'.26: Πίνακας σύγχυσης «Confusion matrix».

		True Class	
		Positive	Negative
Predicted Class	Positive	TP	FP
	Negative	FN	TN

προτιμότερο να λαμβάνουμε υπόψιν και άλλες μετρικές αξιολόγησης οι οποίες αναφέρονται από τον (Han,2011) ως εξής:

- Ακρίβεια (precision)(είναι το ποσοστό των παραδειγμάτων που ο κατηγοριοποιητής έχει κατηγοριοποιήσει ως θετικά και είναι πραγματικά θετικά.)

$$Precision = \frac{TP}{TP + FP}$$

- Ανάκληση (recall)(είναι το ποσοστό των θετικών παραδειγμάτων που κατάφερε να βρει ο κατηγοριοποιητής)

$$Recall = \frac{TP}{TP + FN}$$

- (F-score)(θεωρείται ο σταθμισμένος αρμονικός μέσος όρος της ακρίβειας και της ανάκλησης)

$$F1 = \frac{2 \cdot precision \cdot recall}{precision + recall}$$

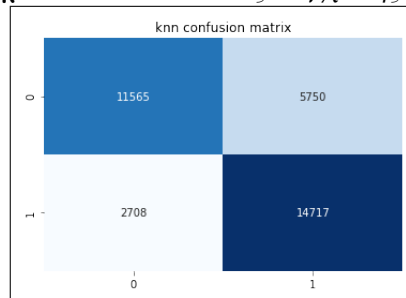
Παρακάτω παρουσιάζονται τα αποτελέσματα και για τους 4 αλγόριθμους πρόβλεψης.

Αρχικά στον παρακάτω πίνακα παρουσιάζεται η επιτυχία που είχαν οι αλγόριθμοι στην πρόβλεψη των δεδομένων.Όπως προαναφέρθηκε η επιτυχία υπολογίστηκε μέσω της τεχνικής k-fold cross validation.

Επιτυχία μοντέλων			
KNN	SVM	Neural Networks	Random forest
75.68	75.43	78.71	76.03

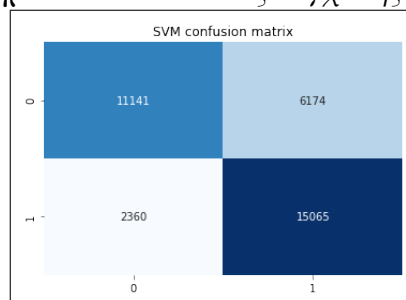
Για το k-fold cross validation χρησιμοποιήσαμε $k = 10$. Για όλους τους αλγόριθμους χρησιμοποιήσαμε τους παραμέτρους επέστρεψαν τα καλύτερα αποτελέσματα. Για τον Knn χρησιμοποιήσαμε $k = 40$. Για τον SVM χρησιμοποιήσαμε τις παραμέτρους Σταθερά $C = 1.0$, Συνάρτηση πυρήνα(kernel Function)= γραμμική(linear) και βαθμό συνάρτησης(degree)= 2. Για τον Neural network χρησιμοποιήσαμε: `hidden_layer_sizes = 50,50,50` (3 κρυφά στρώματα με 50 κρυφές μονάδες το καθένα), `activation = "relu"`, `solver = "adam"`, `alpha = 0.0001` και `learning_rate = "constant"`. Τελικά για τον αλγόριθμο RF χρησιμοποιήσαμε: `n_estimators`(αριθμός δέντρων) = 100 και `max_depth = 5`. Παρακάτω παρουσιάζονται οι πίνακες σύγχυσης για τα μοντέλα. Στην εικόνα Δ'.27 εμφανίζεται ο πίνακας σύγχυσης για το αλγόριθμο knn σύμφωνα με τον πίνακα έχουμε Precision = 0.6679 και Recall = 0.8103 Έπειτα στην εικόνα Δ'.28 εικονίζεται ο πίνακας σύγχυσης για το αλγόριθμο svm σύμφωνα

Σχήμα Δ'.27: Πίνακας σύγχυσης«Knn».



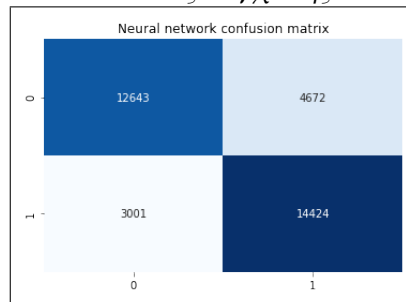
με τον πίνακα έχουμε Precision = 0.6434 και Recall = 0.8252 Έπειτα στην εικόνα Δ'.29

Σχήμα Δ'.28: Πίνακας σύγχυσης«Svm».



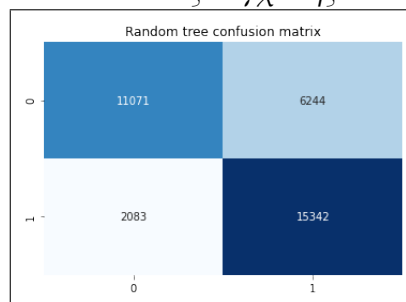
εικονίζεται ο πίνακας σύγχυσης για το αλγόριθμο Neural network σύμφωνα με τον πίνακα έχουμε Precision = 0.7302 και Recall = 0.8082 Τέλος στην εικόνα Δ'.30 εικονίζεται ο πί-

Σχήμα Δ'.29: Πίνακας σύγχυσης «Neural network».



νακας σύγχυσης για το αλγόριθμο Random forest σύμφωνα με τον πίνακα έχουμε Precision = 0.6394 και Recall = 0.8416 Παρατηρούμε οπότε ότι την μεγαλύτερη επιτυχία την είχε το

Σχήμα Δ'.30: Πίνακας σύγχυσης «Random forest».



μοντέλο που δημιούργησε ο αλγόριθμος πρόβλεψης neural network. Παρόλα αυτά, αυτό δεν σημαίνει ότι ήταν και το καλύτερο μοντέλο. Για παράδειγμα αν σκεφτούμε από οικονομικής άποψης, δεν θα θέλαμε να ξοδέψουμε πολλά χρήματα σε ένα κομμάτι το οποίο τελικά δεν θα γίνει διάσημο δηλαδή θα θέλαμε τα false positive (τα κομμάτια τα οποία ταξινομήθηκαν λανθασμένα ως διάσημα) να είναι όσο λιγότερα γίνεται. Έτσι όσο λιγότερα τα false positive τόσο μικρότερη η πιθανότητα να 'ξοδέψουμε' χρόνο και χρήματα για ένα κομμάτι το οποίο τελικά δεν θα γίνει διάσημο. Όπως μπορούμε να δούμε στα διαγράμματα σύγχυσης το μοντέλο που το πετυχαίνει καλύτερα αυτό είναι το μοντέλο που δημιουργείται από τον αλγόριθμο random forest αφού έχει τα λιγότερα false Positive από όλα τα άλλα μοντέλα, όπως φαίνεται και στην εικόνα Δ'.30.

Κεφάλαιο Ε΄

Συμπεράσματα και μελλοντική έρευνα

Κύριο θέμα της εργασίας ήταν η συλλογή και επαναδιάθεση δεδομένων δημοτικότητας μουσικών κομματιών και έπειτα η πρόβλεψη τους μέσω των μουσικών χαρακτηριστικών τους. Με το πέρας των πειραμάτων μας δίνεται η δυνατότητα πλέον να καταλήξουμε σε κάποια συμπεράσματα. Όπως αναφέρθηκε και στο κεφάλαιο της βιβλιογραφίας έχουν ήδη υπάρξει κάποιες έρευνες πάνω στο θέμα και μάλιστα η παρούσα εργασία βασίστηκε εν μέρη σε μία από αυτές. Παρόλα αυτά μόνο με την περαιτέρω μελέτη του θέματος σε συνδυασμό με την αναδυόμενη νέα τεχνολογία θα μας οδηγήσει σε χρήσιμα συμπεράσματα και εν τέλη σε ένα πρακτικό και ολοκληρωμένο σύστημα πρόβλεψης και αξιοποίησης των δεδομένων δημοτικότητας. Στο τελευταίο κεφάλαιο θα γίνει μια ποιοτική περιγραφή των αποτελεσμάτων καθώς επίσης θα παραθέσουμε τα συμπεράσματα που είχαμε από την εργασία. Επίσης θα γίνει μια ανάλυση της πιθανής μελλοντικής έρευνας πάνω στο θέμα.

Ε΄.1 Συμπεράσματα και παρατηρήσεις

Στην ενότητα Γ΄ αναφερθήκαμε στην προσπάθεια που έγινε πάνω στην συλλογή και στην επαναδιάθεση των δεδομένων. Δυστυχώς έκπληξη αποτέλεσε για εμάς το πόσο ελλιπείς ήταν οι πηγές στην διάθεση και στην επεξήγηση των δεδομένων. Οι περισσότερες πηγές (συμπεριλαμβανομένων των πηγών που χρησιμοποιήσαμε) δεν παρέχουν κάποιου είδους API για την ορθή χρήση και συλλογή των δεδομένων ή αν παρέχουν δεν έχουν μεριμνήσει για τις λίστες

δημοτικότητας. Η ανάγκη για ένα ολοκληρωμένο api επιβεβαιώνεται απο τα προβλήματα που παρατηρήθηκαν κατά το scraping των δεδομένων κυρίως στο Billboard, το οποίο όπως είναι λογικό, έπειτα από κάποιες αλλεπάλληλες επισκέψεις μας απέκλειε από την σελίδα του. Έπειτα οι περισσότερες πηγές αναφέρουν πολύ λίγες ή και καθόλου πληροφορίες όσον αφορά την διαδικασία και τις τεχνικές συλλογής των δεδομένων δημοτικότητας καθώς και τις αρχικές πηγές και δεν υπάρχει κάποιας μορφής επίσημο documentation. Στην συνέχεια στην ενότητα Δ' αναφερθήκαμε στην προσπάθεια που έγινε πάνω στην ανάλυση και πρόβλεψη των δεδομένων. Αρχικά και σε αυτό το κομμάτι των δεδομένων παρόλο που το spotify παρείχε ένα αρκετά καλό api για την συλλογή των spotify audio features δεν παρείχε κάποιο επίσημο documentantion που να αναφέρονταν στον τρόπο υπολογισμού ή στην επεξήγηση αυτών με αποτέλεσμα να αναγκαστούμε δυστυχώς να βασιστούμε σε τρίτες πηγές για να κατανοήσουμε και να εξηγήσουμε τα δεδομένα. Σύμφωνα με την ανάλυση που έγινε στα δεδομένα μας πριν την πρόβλεψη τους, αξιόλογη είναι η συσχέτιση των μουσικών χαρακτηριστικών instrumentality και loudness(και σε μικρότερο βαθμό του valence) με τον δείκτη On_chart. Κάτι που πιθανότητα δηλώνει ότι η ποσότητα των φωνητικών σε ένα κομμάτι, η ένταση του κομματιού καθώς και η θετικότητα που εκπνέει ένα κομμάτι παίζουν σημαντικό ρόλο στην δημοτικότητα που θα αποκτήσει ένα κομμάτι και εάν τελικά θα φτάσει σε κάποια λίστα chart η όχι. Έπειτα λαμβάνοντας υπόψιν την μικρή έκταση του πειράματος και την μικρή υπολογιστική ισχύ, που σε συνδυασμό με το μέγεθος των δεδομένων καθυστέρησε για κάποιο χρονικό διάστημα την εκτέλεση του πειράματος, τα αποτελέσματα κατέληξαν να είναι αρκετά καλά. Πόσο μάλλον αν σκεφτεί κανείς ότι αποτελούν αποτέλεσμα από cross-validation, που σημαίνει ότι τα ποσοστά αυτά ανταποκρίνονται σε μεγάλο βαθμό στην πραγματικότητα και δεν αποτελούν σε καμία περίπτωση υπερπροσαρμογή στο dataset μας.(Han,2011) Παρόλα αυτά ίσως απέχουμε ακόμα αρκετά από αποτελέσματα που θα μπορούσαν να οδηγήσουν σε κάποιο πρακτικό σύστημα υποστήριξης αποφάσεων στον τομέα της μουσικής βιομηχανίας.

Ε'.2 Μελλοντική έρευνα

Όπως και με τα περισσότερα συστήματα έτσι και το δικό μας επιδέχεται ακόμα πολλές βελτιώσεις τόσο στο κομμάτι της συλλογής και επαναδιάθεσης των δεδομένων όσο και στο κομμάτι της πρόβλεψης των δεδομένων. Αρχικά όσον αφορά την συλλογή των δεδομένων, προσπάθειες πρέπει να γίνουν στην περαιτέρω 'θωράκιση' τού αυτοματοποιημένου κώδικα από προβλήματα που μπορεί να προκύψουν με τις πηγές. Για παράδειγμα ένα έξυπνο μέτρο θα ήταν η αυτόματη

επανεκτέλεση του κώδικα σε κάποιο άλλο χρονικό διάστημα αν στην πρώτη εκτέλεση κάποια από τις πηγές δεν ήταν διαθέσιμη για οποιονδήποτε λόγο. Έπειτα η μελέτη και η προσθήκη νέων πηγών λιστών δημοφιλών μουσικών κομματιών θα έκανε τα δεδομένα μας πιο ολοκληρωμένα και θα βοηθούσε περαιτέρω στο κομμάτι της πρόβλεψης των δεδομένων. Επίσης θα μας έδινε κάποιες εναλλακτικές σε περίπτωση που κάποια από τις πηγές διακόψει την λειτουργία της. Μετά όσον αφορά το κομμάτι της επαναδιάθεσης των δεδομένων, περισσότερη έρευνα πρέπει να γίνει πάνω στο κομμάτι της διαγραμματικής απεικόνισης των δεδομένων. Συγκεκριμένα χρήσιμο θα ήταν να προσθέτουν νέοι πιο δυναμικοί τύποι διαγραμμάτων οι οποίοι θα επιτρέπουν στον χρήστη να περιηγηθεί πιο διαδραστικά στα δεδομένα και να τα κατανοήσει καλύτερα. Επίσης χρήσιμο θα ήταν στην σελίδα να προστεθούν και διαγράμματα των μουσικών χαρακτηριστικών των κομματιών για χρήση τους από τους επαγγελματίες της μουσικής βιομηχανίας. Επιπρόσθετα στο κομμάτι της πρόβλεψης των δεδομένων, αρχικά περισσότερη σημασία θα πρέπει να δοθεί στα μουσικά χαρακτηριστικά του spotify και ειδικότερα στα instrumentality, loudness και valence μιας και δυστυχώς έως τώρα λίγες είναι οι προσπάθειες που βασίζονται αποκλειστικά στα χαρακτηριστικά που εξάγει η πλατφόρμα. Εν κατακλείδι δεδομένα από νέες πηγές δημοτικότητας θα παρήγαγαν ακόμα καλύτερα μοντέλα. Σημασία επίσης πρέπει να δοθεί στην αξιοποίηση και άλλων δεδομένων των κομματιών όπως θα ήταν το είδος τους, η χρονολογία τους καθώς και τα trends που υπήρχαν την χρονιά που κυκλοφόρησαν, καθώς τα μουσικά χαρακτηριστικά από μόνα τους πιθανότητα, όπως προαναφέρθηκε να μην είναι αρκετά. Τελικά ένας υβριδικός τρόπος πρόβλεψης των δεδομένων ίσως να αποτελεί την λύση στο πρόβλημα.

Βιβλιογραφία

Antonia pleurou(2012). *Artificial neural networks simulating human brain*.

Billboard to Alter Chart Tracking Week for Global Release Date. (2015, June).
<https://www.billboard.com/articles/business/chart-beat/6605842/billboard-alter-chart-tracking-week-global-release-date/>

Billboard Chart & Magazine Dates Now to Align Closer to Release Week.(2017, December). <https://www.billboard.com/articles/columns/chart-beat/8071011/billboard-chartmagazinedatesreleaseweekchange>

Dhanaraj, R., & Logan, B. (2005, September). *Automatic Prediction of Hit Songs*. In ISMIR. (pp. 488-491).

David Martin Powers (2014). *Evaluation: from precision, recall and f-measure to roc, informedness, markedness and correlation*.

Feurer, M., & Hutter, F. (2019). *Hyperparameter optimization*. In Automated Machine Learning (pp. 3-33). Springer, Cham.

Getting started Docker.(2021). <https://docs.docker.com/get-started/>

Install and Use Jupyter.(2021). <https://jupyter.readthedocs.io/en/latest/install.html>

Japkowicz, N., & Stephen, S. (2002). *The class imbalance problem: A systematic study*. Intelligent data analysis, 6(5), 429-449.

Karydi, D., & Karydis, I. (2014). *Legal issues of aggregating and curating information flows: The case of rss protocol*. In International conference on information law.

Karydis, I., Gkiokas, A., Katsouros, V., & Iliadis, L. (2018). *Musical track popularity mining dataset: Extension & experimentation*. Neurocomputing, 280, 76-85.

Karydis, I., Gkiokas, A., & Katsouros, V. (2016, September). *Musical track popularity mining dataset*. In IFIP International Conference on Artificial Intelligence Applications and Innovations (pp. 562-572). Springer, Cham.

Karydis, I. (2016). *dataset part a schema2(image)*. Music Information Research.

Middlebrook, K., & Sheik, K. (2019). *Song hit prediction: Predicting billboard hits using spotify data*. arXiv preprint arXiv:1908.08609.

Pachet, F., & Roy, P. (2008, September). *Hit Song Science Is Not Yet a Science*. In ISMIR (pp. 355-360).

Record chart.(n.d).Recovered on 19 April,2021, From Wikipedia.
https://en.wikipedia.org/wiki/Record_chart#cite_note-whitburn-1

Singhi, A., & Brown, D. G. (2014). *Hit song detection using lyric features alone*. Proceedings of International Society for Music Information Retrieval.

State L. & Vlamos P. (2000). *The POMDP modeling in the Analysis of Genetic Algorithms*. The Annals of Bucharest University XLIX.

Spotify Web Api.(2021). <https://developer.spotify.com/documentation/web-api/>

Shazam-data.(2020). <https://help.chartmetric.com/en/articles/3273219-shazam-data>

Scikit-learn Getting started.(2021). https://scikit-learn.org/stable/getting_started.html

What is a Correlation Matrix?(2021). <https://www.displayr.com/what-is-a-correlation-matrix/>

Zangerle, E., Votter, M., Huber, R., & Yang, Y. H. (2019). *Hit Song Prediction: Leveraging Low-and High-Level Audio Features*. In ISMIR. (pp. 319-326).

F.A.Q.(2021). <https://artists.spotify.com/faq/stats>

Han, J., Kamber, M., & Pei, J. (2011). *Data mining concepts and techniques third edition*. The Morgan Kaufmann Series in Data Management Systems, 5(4), 83-124.

Python language.(2021). http://fourier.math.uoc.gr/mk/prog1718/files/notes-b/01_variables.pdf

What is NumPy?(2021). <https://numpy.org/doc/stable/user/whatisnumpy.html>

Pandas Package overview.(2021). https://pandas.pydata.org/docs/getting_started/overview.html

Matplotlib User's Guide.(2021). <https://matplotlib.org/stable/users/index.html>