

# L'anàlisi tècnica-emocional de la música.

Pol Coma i Barbarà

**Resum** — Sabem que la música és molt subjectiva i en aquest treball es proposa relacionar les valoracions emocionals sobre cançons recollides a través d'una App, amb les progressions harmòniques que tenen les cançons. Tot això per desenvolupar, fent ús de tècniques d'intel·ligència artificial (IA), un model classificador on cercar patrons per a fer recomanacions i/o classificació. Aquest treball és una extensió al projecte VO i el resultat esdevé una funcionalitat que serà inclosa a l'App.

**Paraules clau** — VO, música, emocions, progressions harmòniques, classificador, base de dades, python, KNN, aprenentatge computacional, IA

**Abstract**— It is a well-known fact that music is a highly subjective issue in terms of preferences. Taking this into account, the aim of this project is to relate emotional valorations from songs using an App, with harmonic progressions from them. Therefore, the main idea is to develop a classifying model to look for patterns in order to make recommendations or classification, according to the gathered information and by using an artificial intelligence system (AI). This work, in fact, is an extension of the VO project that will be included in the App.

**Index Terms** — VO, music, emotions, harmonic progressions, classifier, database, python, KNN, machine-learning, IA



## 1. Introducció

Des de ben petits som en contacte amb la música. A casa, en el metro, caminant, en un centre comercial, en el cotxe, fent esport... És més, durant els últims anys, gràcies a la tecnologia es pot dir que es té música en qualsevol lloc i moment.

Per un motiu o un altre, hi ha cançons que ens fan sentir bé, algunes que ens provoquen recordar moments passats, d'altres que fan que desconnectem i també les que ens porten a pensar en realitats fictícies fruit de la pròpia imaginació.

La música, present en més o menys quantitat en el nostre dia a dia, s'ha instaurat com a vital en les nostres vides per el fort vincle que té amb la psicologia.

El desenvolupament d'aquest treball de fi de grau, és una extensió que serà inclosa en un projecte autodidacte basat en una App anomenada VO i que té com a principal objectiu fer la música més interactiva.

Gràcies als sistemes de computació actuals, som davant un experiment social que posa a prova temes tan subjectius com la música i les emocions, per tal d'intentar trobar patrons naturals guiats per la subjectivitat de les persones.

Les aplicacions mòbils han revolucionat el món gràcies al sentiment d'utilitat que generen en el consumidor i la facilitat d'ús que suposen.

Qualsevol sistema d'intel·ligència artificial que actua com recomanador necessita prèviament un classificador que interpreti les característiques que es busquen i respongui amb un objecte d'interès a canvi. Tot classificador necessita doncs informació per poder associar característiques a certs objectes, en aquest cas cançons. És aquí on es fa ús de VO per recaptar informació dels usuaris i poder ser processada a posteriori.

El repte principal que planteja aquest projecte és doncs analitzar la informació que es genera a través d'un sistema informàtic i aconseguir un únic conjunt de dades format per dos àmbits diferents com és la teoria de la música i les emocions.

Arribar a aquest punt implica un alt nivell d'abstracció i interpretació que suposa una gran dedicació d'hores en aprendre els nous conceptes i camps.

Relacionar aquests dos àmbits ha comportat treballar en equip amb diferents persones les quals han aportat informació molt valuosa en el projecte i ha estat essencial per poder avançar i aconseguir resultats.

La memòria s'estructura de la següent manera: objectius, estat de l'art, treball previ, metodologia, resultats, conclusions i futures accions.

- E-mail de contacte: [pol.coma@e-campus.uab.cat](mailto:pol.coma@e-campus.uab.cat)
- Menció realitzada: Enginyeria de Computació
- Treball tutoritzat per: Josep Lladós (Computació)
- Curs 2019/20

## 2. Treball previ (VO)

El transcurs del TFG ha sigut sobre una base previa, el desenvolupament de l'App VO.

Gràcies a la informació que genera aquesta aplicació, s'obren les portes a la implementació d'una funció d'intel·ligència artificial supervisada basada en la classificació de la música.

VO és una aplicació multiplataforma (iOS / Android), client-servidor desenvolupada en **Xamarin** (C#) per a dispositius mòbil. La idea va sorgir al 2015 i fins que no es va arribar assolir coneixements de programació (nivells bàsics i avançats) no es va començar a desenvolupar (2018). A dia d'avui, encara es troba en construcció.

El desenvolupament d'aquesta App sempre ha estat guiat per a prototips desenvolupats amb Adobe Photoshop, la qual cosa ha suposat visualitzar més fàcilment el seu desenvolupament i veure'n la viabilitat. Per a aconseguir tenir les funcionalitats desitjades s'ha vinculat amb serveis com Amazon S3 Spotify, Genius... Per altra banda, el servidor ha estat dissenyat i implementat des de zero i s'ha utilitzat **Python** com a llenguatge de programació per aquesta part. Concretament s'ha utilitzat sockets TCP/IP i s'ha elaborat un propi protocol i sistema d'criptació. Aquest també disposa d'una IP fixa que permet connectar-s'hi des d'arreu del món.

Sobre les dades que genera i treballa, cada petició que comporta realitzar una nova acció o enviar alguna informació, s'emmagatzema a la base de dades implementada en una arquitectura relacional (MYSQL). Previ a l'inici del desenvolupament d'aquest treball, VO era funcional només en la plataforma iOS. Com que la font d'informació en aquest projecte són els usuaris i no tots usen el mateix sistema operatiu, s'ha dedicat un període de temps en adaptar el codi a la plataforma Android.

Això també ha comportat implementar noves funcions i mesures de seguretat per a poder garantir el correcte ús de l'aplicació. Entre d'altres, sent el servidor capaç d'acceptar diverses connexions entrants, respondre concurrentment de manera eficient, tractant excepcions i errors i finalment emmagatzemant correctament la informació a la base de dades per poder ser analitzada posteriorment.

## 2. Objectius

El context en el que es desenvolupa aquest treball planteja molts objectius i sovint ha sigut difícil discernir el TFG de la part de la App.

A continuació es precisen els detalls de cada un d'ells:

### 2.1. Implementació de la funcionalitat a VO.

Per facilitar les coses als usuaris (com que VO no es troba finalitzat i encara hi ha components en construcció), es necessita un apartat dedicat al TFG per

tal de centralitzar la informació i facilitar als usuaris la tasca. Aquest inclou una llista de cançons seleccionades per VO i a part, permet que els usuaris proposin cançons per a ser valorades pels demés.

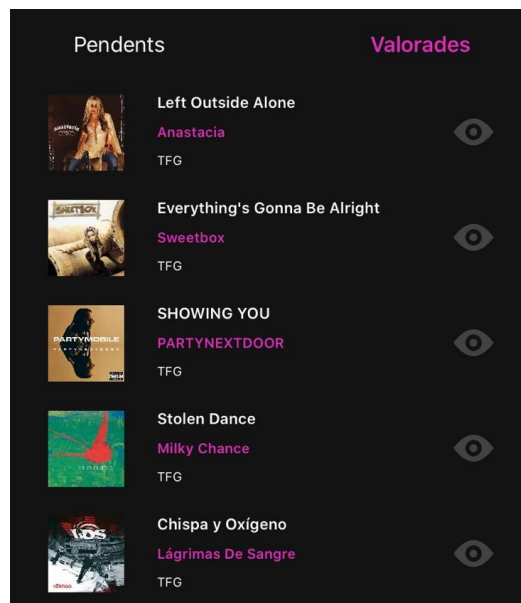


fig 1. Apartat del TFG dins VO

### 2.2. Abstracció d'un model computacional.

Un dels principals objectius del projecte és aconseguir relacionar un conjunt de característiques de les cançons amb les emocions i obtenir un model de dades el qual ofereixi la possibilitat cercar característiques referents a les cançons per a fer classificacions i recomanacions.

### 2.3. Implementació del classificador.

Per últim, un cop s'hagin assolit els objectius anteriors, es dona pas a l'últim i principal objectiu del projecte; implementar un algorisme capaç d'aprendre a classificar la música en diferents emocions humanes.

És l'últim i alhora el principal objectiu del projecte, però la dependència de les dades fan que sigui el l'últim a desenvolupar.

## 3. Estat de l'art

En plena era digital, constantment sabem de noves Apps que han sortit al mercat, ja sigui per algun amic o mitjançant la publicitat. Actualment existeixen multitud d'aplicacions musicals en les quals un usuari pot escoltar i guardar-se música. Exemples clars i que són àmpliament coneguts són Spotify, Apple Music, Youtube Music entre d'altres. Aquests també fan recomanacions i per tant, disposen d'un classificador que relaciona les cançons d'una manera o altre.

Tots ells tenen en comú la funcionalitat principal que és la reproducció de música. El que les diferencia unes de les altres normalment és la tarifa mensual i poc més.

Vibe out, l'app que utilitza aquest projecte per extreure dades dels usuaris, pretén marcar una nova tendència en el món de les apps musicals.

Gràcies a l'expansió en la que es troba actualment la IA, molts sistemes s'estan actualitzant a versions que són capaços d'aprendre i millorar els seus resultats. [1]

La gran majoria de classificadors musicals actuals es basen en mostrejos freqüencials d'ones intentant obtenir i observar patrons. [2]

En aquest projecte es pretén introduir una nova manera de comprendre i entendre la música per tal de poder-la classificar i recomanar.

#### 4. Metodologia

La metodologia del treball es divideix en dues parts ben diferenciades entre les quals existeix una relació de dependència directa. La primera comprèn la recollida, l'estudi i anàlisi de les dades que s'usen i formen la base per a la segona fase. Aquesta segona part fa referència a com es processaran i s'utilitzaran les dades per tal d'assolir l'objectiu principal que és desenvolupar un classificador.

Tot el procés que comporten els objectius definits anteriorment, s'ha desenvolupat amb Python fent ús de l'IDE Sublime Text i executant el codi en el terminal.

Per altra banda, també s'ha treballat amb WorkBench SQL per a administrar i extreure dades de la base de dades.

A continuació es detallen els passos que s'han seguit per assolir els objectius marcats.

##### 4.1 Generació d'un model abstracte de dades

Com s'ha definit anteriorment, un dels principals objectius del projecte és crear un model de dades que permeti poder classificar cançons en estats d'ànim.

Per aconseguir-ho, en aquest treball es proposa tractar amb les emocions i les característiques tècniques d'una cançó. En aquest cas els acords.

Relacionar aquests dos àmbits completament diferents ha significat mesos de treball i investigació per la dificultat que té sobretot la música a nivell de teoria musical. Les emocions han suposat també dedicació però el grau de dificultat ha estat més fàcil per la quotidianitat i presència en les nostres vides d'aquestes.

##### 4.1.1 Generació del model emocional

Primer de tot s'ha d'entendre que és el que passa quan escoltem una cançó. Sovint escoltant música, per motius molt subjectius i personals hi ha moments on experimentem sensacions i emocions que ens provoca aquesta.

No s'ha de confondre emoció i estat d'ànim.

Una emoció és una sensació que vivim normalment durant un curt període de temps, si ho comparem amb la durada que té un estat d'ànim que tendeix a perdurar més temps en la vida d'una persona.

Normalment els causants de les emocions són events específics lligats completament a la subjectivitat i manera de ser d'una persona. La felicitat que sents quan et diuen que has aprovat un examen, o quan et

diuen que has entrat en una nova feina són emocions positives. Perdre les ganes, tenir ganes d'evadir-te quan perds un partit on et jugaves entrar a la final, són d'altres emocions en aquest cas negatives, però que tenen en comú amb les positives, que en més o menys temps, aniran perdent intensitat.

No es pot parlar d'intensitats concretes ja que també té a veure amb la persona i cadascú ho viu d'una manera diferent, però sí que totes les emocions normalment es caracteritzen per la seva curta durada.

Aquesta fluctuació més ràpida és provocada per la espontaneïtat que ofereixen les emocions. Mentre que per altra banda, l'estat d'ànim varia més lentament i defineix una manera d'estar, de romandre d'una manera concreta en un període més ampli de temps[3].

Seguint la teoria del psicòleg Bertrand Russell (1980), les emocions estan compreses entre dues variables: Grat-desgrat i nivell d'activació. [4]

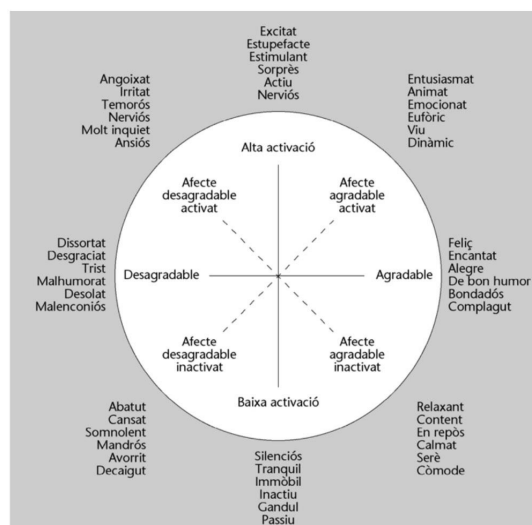


fig 2. Model de Russell

Gràcies a aquesta teoria i model de classificació de les emocions, s'han seleccionat diverses emocions i se n'han dissenyat icones per tal que els usuaris puguin decidir sobre cada cançó què és el que senten o amb quina emoció s'identifiquen [5].

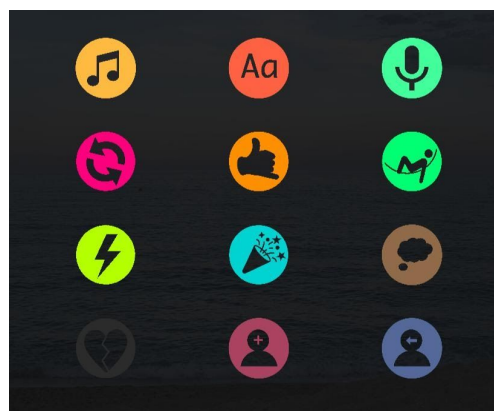


fig 3. Icones de VO

De les dotze icones que apareixen a la imatge, set són les que permeten saber què pensen els usuaris sobre les cançons. Les altres són per a altres funcionalitats inherents a la app, sent també referències emocionals però subjectes a un altra emoció, és a dir, és una especialització d'una emoció i en aquesta part només volem tractar amb emocions absolutes:

- 1.- Alegria / Bon rotllo
- 2.- Calma / Tranquil·litat
- 3.- Energia / Motivació
- 4.- Ganes de festa
- 5.- Reflexió
- 6.- Tristesa
- 7.- Nostàlgia

Als usuaris se'ls ofereix la possibilitat de seleccionar més d'una etiqueta i per tant cal analitzar la relació de les etiquetes triades. Per això s'ha elaborat la següent matriu on s'identifica a quina categoria del model de Russell pertanyen dues etiquetes diferents.

Emoció	Good vibes	Motivation	Party	Nostalgia	Sadness	Chill	Reflexion
Good vibes		AA	AA	AB	X	AB	AB
Motivation			AA	AA	X	AB	AA
Party				AA	X	AN	X
Nostalgia					DN	NB	DA
Sadness						DB	DA
Chill							NA
Reflexion							

- (NA) Neutre Alta Activació
- (AN) Agradable Neutre
- (AA) Agradable Alta activació
- (NB) Neutre Baixa Activació
- (DA) Desagradable Alta Activació
- (DN) Desagradable Neutre
- (AB) Agradable Baixa Activació
- (DB) Desagradable Baixa Activació

fig 4. Icones de VO

Gràcies aquesta matriu, si hi ha més d'una etiqueta en la valoració d'un usuari es pot deduir a quina família, dins del model emocional proposat per Russell, pertany. L'algorisme per a l'obtenció de la família és la representació de la matriu de la figura 4 en codi.

1. **Per a cada etiqueta dins del conjunt d'etiquetes:**  
input = [alegria, festa, motivació]
  - 1.1. S'emparella amb la resta d'etiquetes i s'emmagatzema el resultat segons la matriu.  
Output = [AA,AA,AA]
2. **Acumulació dels resultats.**
3. **Resultat = el resultat més repetitiu.**

### Etiquetatge de cançons

Ara que ja s'ha aconseguit saber i ubicar en el model d'emocions el que un usuari expressa a través de les icones, s'ha d'etiquetar totes les cançons per tal de

poder-les agrupar en emocions i posteriorment analitzar-ne la part tècnica.

A continuació s'indiquen els passos següents:

1. **Per a cada cançó:**
  - 1.1. Per a cada usuari que l'ha valorat.
    - 1.1.1. Incrementar els comptadors de les icones seleccionades.
2. **Per a cada cançó amb els recomptes de les emocions.**
  - 2.1. Nivell límit. Valor a partir del qual s'etiqueta una cançó amb aquella emoció.
  - 2.2. Si el nombre de vegades que s'ha seleccionat la icona > nivell límit, aquella cançó es defineix amb aquella emoció.

Emoció	Cançons etiquetades
Alegria	1,2,5,6
Calma	3,4,8,11
Energia	7,9,10,15
Ganes de Festa	12,13,14,16,20
Reflexió	15
Tristesa	17,18,19
Nostàlgia	21,22

fig 5. Resultats de l'etiquetatge

Com s'explica en el pas 2.1 de l'algorisme, el nivell límit és un valor per decidir quina és la quantitat de votacions que ha de tenir una icona per a que passi a ser l'etiqueta emocional que la descriu.

El valor límit ha estat calculat a partir del nombre total d'usuaris que han valorat aquella cançó i una tolerància per determinar la mesura de vots necessaris que s'ha d'assolir per a que aquella etiqueta emocional esdevingui la que defineix la cançó:

$$\text{Valor límit} = n^{\circ} \text{ usuaris} * \text{tol}\cdot\text{lèrancia}$$

Els valors que s'han usat per a la tolerància es comenten en l'apartat de resultats i conclusions.

### 4.1.2 Generació de l'anàlisi tècnic-musical.

La música, és sense dubte, un camp molt extens en el qual la seva base teòrica sovint és difícil d'interpretar i més, si es parteix de nocions bàsiques. Gràcies a experts en el tema aquest handicap s'ha fet més amè i ha permès assolir un grau de coneixement suficient per a definir quina informació és la valuosa i es pot utilitzar per a reconèixer patrons.

Abans de començar aquest treball, s'ha estat utilitzant la API de Spotify per a poder obtenir cançons i d' altres funcions per al desenvolupament de VO.

Aquesta API també ofereix l'anàlisi, tècnic de les cançons, obtenint com a resultat els BPM, el time signature, el timbre i finalment els pitches[6]. La primera idea va ser utilitzar el pitch per extreure quines notes estan sonant durant la cançó.

Les notes que utilitza són les dotze de l'escala cromàtica:

**Do, Do#, Re, Re#, Mi, Fa, Fa#, Sol, Sol#, La, La#, Si**

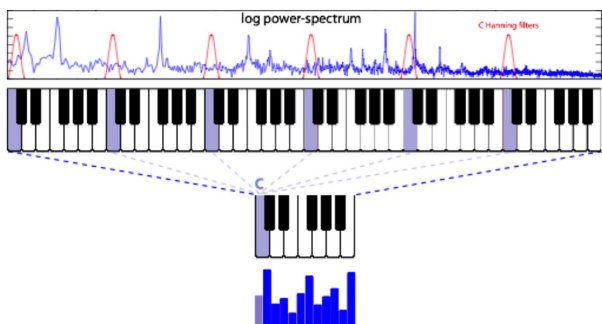


fig 6. Escala cromàtica resultant. Pitch

El pitch és el càlcul de les freqüències que emeten les notes que s'estan tocant en un determinat instant de temps. Spotify el calcula sobre les sis octaves d'un piano i fa el promig de cada nota en les diferents octaves oferint com a resultat un vector de dotze posicions on cada posició representa la mitjana d'una mateixa nota en les sis octaves. De manera que [Do,Do#...Si] té un vector equivalent a [0.3, 0,2 ...,1].

Els valor 0 indica que aquella nota no hi és i 1 significa el to pur.

La API retorna la informació en format JSON i s'ha utilitzat un fitxer per centralitzar la informació dels vectors del pitch i d'altres dades tècniques.

Format del fitxer generat on es recull la informació referent al pitch.

*ID cançó*

*Títol cançó - Artista*

segon 0 | valor DO, valor DO#, valor Re... valor SI

segon N | valor DO, valor DO#, valor Re... valor SI

Exemple:

*367fy3uofb3483g*

*Blue World - Mac Miller*

línia 0 s 0.03 | 0.1, 0.5, 0.2, 0.9, 1, 0.6 ... 0.2

línia 1 s 0.54 | 0.3, 0.3, 0.1, 0.8, 1, 0.7 ... 0.7

línia 2 s 0.92 | 0.2, 0.2, 0.1, 0.7, 1, 0.8 ... 0.1

+ BPM, time signature, timbre...

Seguint l'exemple anterior a partir de la segona línia, apareix la informació que l'algorisme utilitza:

Cal matisar que no hi ha una única fila d'informació per a cada segon en número enter. A l'exemple anterior tenim tres files que el segon en valor enter és el 0. El

número de línies que representen doncs a un segon en valor enter, va variant i no sempre obtenim el mateix resultat.

Saber les notes que hi ha a cada segon permet fer agrupacions a posteriori i obtenir com a resultat els acords.

### Càlcul de Notes

Un cop coneguda la informació amb la qual es treballa, a continuació es mostra l'esquelet del procediment que va analitzant línia per línia cercant les notes que hi ha.

En aquesta part, s'han desenvolupat tres versions diferents d'agrupació de notes ja que la informació que ofereix Spotify no és del tot precisa i s'ha volgut processar de diferents maneres per veure quina ofereix millor resultat.

Abans d'entrar en matèria, per poder entendre millor el procediment s'explica com funciona l'estructura de dades del diccionari de Python.

El diccionari té una clau i un valor, en aquest cas una llista de notes com a valor.

```
diccionari = { 'Segon1' : [nota1,nota2..nota N] }
diccionari['Segon1'] = llista de notes
```

### V1: Mitjana

Com s'ha comentat en l'exemple anterior el segon 0 (en valor enter) està format per tres línies d'informació.

En aquesta primera versió el que s'ha fet és acumular les tres línies d'informació sumant tots els valors referents a les notes i s'ha dividit cada valor resultant entre el nombre de línies, en aquest cas tres.

A continuació s'indiquen els passos que segueix l'algorisme i per fer-ho més visual utilitzarem el següent mini fitxer:

```
t(s) 0.03 | 0.1, 0.5, 0.2, ... 0.5
t(s) 0.54 | 0.3, 0.3, 0.1, ... 0.7
t(s) 0.92 | 0.2, 0.2, 0.1, ... 0.1
t(s) 1.03 | 0.7, 0.5, 0.2, ... 0.3
t(s) 1.52 | 0.5, 0.3, 0.4, ... 0.7
```

#### 1. Per a cada línia:

- 1.1. Agafar el valor enter del segon per utilitzar-lo com a clau i crear una entrada en el diccionari.
- 1.2. Acumular els valors en el segon que pertorqui del diccionari.

Output en aquest punt:

```
diccionari[0] = [0.6, 1.0, 0.4, ... 1.3]
diccionari[1] = [1.2, 0.8, 0.6, ... 0.9]
```

#### 2. Per a cada clau en el diccionari:

- 2.1. Agafar els valors acumulats de cada segon enter i dividir-lo entre el número de línies que el formen.

Output en aquest punt:

```
diccionari[0] = [0.2, 0.33, 0.13, ... 0.43]
diccionari[1] = [0.6, 0.40, 0.30, ... 0.50]
```

### 3. Per a cada valor mitjà:

3.1. Valor límit = 0.5

3.2. Comparem cada valor amb el límit a partir del qual es decideix si aquella nota es considera present o si en canvi no hi és. valor nota >= valor límit.

Output final:

```
t(s) 0 = []
t(s) 1 = [Do, ... Si]
```

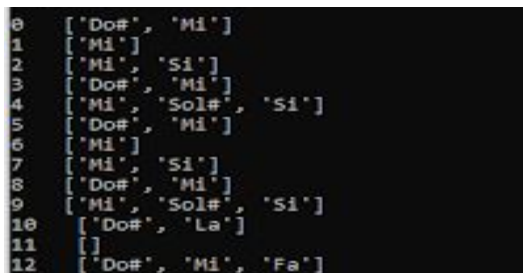


fig 7. Exemple de notes que superen la tolerància en una cançó.

Després de provar i investigar amb diferents nombres naturals entre 0 i 1. El valor 0.5 ha estat el que millors resultats ha ofert i s'ha procedit a treballar sempre amb aquest.

### V2: Argument Màxim

Seguint la mateixa estructura de l'algorisme de la V1, en aquesta versió es proposa trobar l'argument màxim dels valors que representen cada nota en un mateix segon enter, en comptes de fer la mitjana de totes les línies corresponents.

Seguint amb el fitxer que s'ha utilitzat anteriorment en la V1:

#### 1. Per a cada línia:

1.1. Emmagatzemar cada valor al diccionari utilitzant com a clau el segon corresponent a aquella línia d'informació.

Output en aquest punt:

```
diccionari[0] | 0.1, 0.5, 0.2, ... 0.5
               | 0.3, 0.3, 0.1, ... 0.7
               | 0.2, 0.2, 0.1, ... 0.1

diccionari[1] | 0.7, 0.5, 0.2, ... 0.3
               | 0.5, 0.3, 0.4, ... 0.7
```

#### 2. Per a cada clau en el diccionari:

2.1. Comparar tots els valors de la mateixa columna i quedar-se amb el més gran.

Output en aquest punt:

```
t(s) 0 | 0.3, 0.5, 0.2, ... 0.5
t(s) 1 | 0.7, 0.5, 0.4, ... 0.7
```

El següent pas de l'algorisme és igual al de la V1 i V2.

A diferència de la primera versió, aquesta utilitza com a valor límit el 0.8 i 1. A l'apartat de resultats se'n detalla el perquè d'aquests valors.

### V3: Cada línia

Aquesta tercera i última versió s'introdueix a meitat de projecte quan s'observa que a l'hora de reconèixer patrons en les progressions harmòniques, les versions anteriors no aproximen suficientment bé. És per això que s'introdueix aquesta versió on es cerquen les notes en cada línia d'informació independent.

L'algorisme és idèntic al que s'ha mostrat anteriorment. Si més no, ara cada línia d'informació s'analitza independentment i s'aporta més informació.

### Càlcul d'Acords

Arribats en aquest punt, es té organitzada la informació d'una manera que permet analitzar quins acords estan formant les notes que s'han trobat.

Un acord està format per una combinació de notes i pot ser de tipus major o menor. [7]

Existeixen una quantitat immensa d'acords.[8] En aquest treball per a cada nota de les dotze de l'escala cromàtica, s'ha programat els següents acords:

Major, Menor, Augmentat, Disminuit, Sèptima, Menor Sèptima, Major Sèptima. S'ha programat un total de setanta-quatre acords que són els que analitza el programa. Segueixen havent més acords però discutint-ho amb els companys experts en el tema, s'ha arribat a la conclusió que ja són suficients. Els altres acords són estructures més complexes i no són tant usats.

Com que anteriorment hem calculat les notes que hi ha a cada segon, ara l'algorisme ha d'anar recorrent cada segon i mirant quines notes es toquen en aquell instant de temps de manera que es trobi l'acord corresponent.

El càlcul d'aquests acords s'ha fet creant vectors booleans de dotze posicions, representant les dotze notes, on el 1 indica que s'ha trobat aquella nota i un 0 no.

#### 1. Per a cada clau (segon) del diccionari:

- 1.1. Mirar les notes que ha trobat l'algorisme de càlcul de notes.
- 1.2. Crear el vector booleà amb 1 on hi hagi aquella nota.
- 1.3. Comparar el vector generat amb tots els acords possibles.

Output en aquest punt:

```
Acord Do Major = [1,0,0,0,1,0,0,1,0,0,0,0]
Vector booleà en t(N) = [1,0,0,0,1,0,0,1,0,0,0,0]
```

La comprovació del vector generat en cada segon es fa amb els setanta-quatre acords programats. En aquest cas el programa ha trobat que en el segon N hi ha l'acord Major de Do.

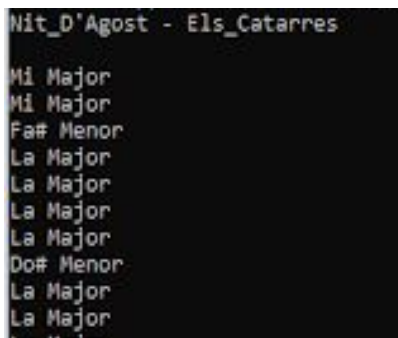


fig 8. Acords trobats

### Progressió harmònica

El problema, però, és que els acords en sí no ofereixen una informació valuosa a l'hora de buscar patrons. Molts acords presents, per exemple, en cançons tristes, apareixen també en d'altres cançons que provoquen alegria.

Després d'investigar, debatre i analitzar amb professionals el conjunt de dades que ofereix Spotify sobre una cançó, es va decidir utilitzar els acords per observar quina progressió harmònica segueixen.

Per entendre què és la progressió harmònica s'ha de començar per l'inici.

Totes les cançons tenen una tonalitat i totes estan formades per acords.

Un acord és una combinació de notes que tocadetes juntes formen el to que es desitja. Seguidament es mostren les notes naturals de l'escala.

Notes de l'escala (partint del Do):

I	II	III	IV	V	VI	VII
(DO)	(RE)	(MI)	(FA)	(SOL)	(LA)	(SI)

Segons la nota amb la que es basa la cançó aquesta escala varia. De manera que per exemple amb la nota Sol, la seva nota I és Sol, la II La, III Si, IV Do etc..

Quan es diu que una cançó té una tonalitat, vol dir que els seus acords estan formats naturalment a partir de la nota que marca la seva tonalitat.

Prenent com a exemple l'acord de Do Major, aquest està format per les notes: Do (nota fonamental), Mi (nota tercera) i Sol (nota cinquena).

Parlar de notes fonamentals, notes terceres, quartes, cinquenes etc... és parlar de la posició que ocupa aquella nota en la escala de tonalitat.

Escala de tonalitat per al Do Major:

1.- Tònica	( Sol Mi Do )
2.- Sobretònica	( La Fa Re )
3.- Mediant	( Si Sol Mi )
4.- Subdominant	( Do La Fa )
5.- Dominant	( Re Si Sol )
6.- Superdominant	( Mi Do La )
7.- Sensible	( Fa Re Si )

Per tant ara només cal saber quina nota és la fonamental per tal de començar a veure quines posicions dins de l'escala de tonalitat ocupen els acords trobats anteriorment.

A continuació s'indiquen els passos que segueix l'algorisme:

1. **Trobar la nota fonamental.**
2. **Agafar el diccionari que conté l'escala de tonalitat per aquella nota.**
3. **Per a cada acord de la cançó:**
  - 3.1. *Mirar quina tonalitat de l'escala és.*
  - 3.2. *Emmagatzemar en una llista la posició en valor enter que ocupa aquest acord dins de l'escala de tonalitat.*

Output en aquest punt:

*Progressions Harmòniques = [1,5,4,2,1,6,3,2,4,2,4,1]*

### Cerca de Patrons

Ara que ja es té els canvis en l'escala de tonalitat només cal cercar patrons.

El món de la música té una immensa quantitat de patrons ja establerts que funcionen molt bé i són presents en la gran majoria de cançons. [9]

En aquest treball s'han utilitzat uns quants per observar en quina freqüència apareixen i a quines cançons ho fan.

Per cercar patrons s'ha desenvolupat un petit procediment que s'encarrega de fer grups de tres valors donada una seqüència de nombres i comprovant si es reconeix com a patró o no.

1. **Donada la seqüència S ([1,2,1,2,3,4]).**
2. **Per a cada valor dins de S:**
  - 2.1. *Crear grups de 3 des del valor actual dins de S fins a S+2. ([1,2,1])*
  - 2.2. *Comprovar si aquest subgrup està dins del conjunt de patrons ja coneguts.*
  - 2.3. *Emmagatzemar en una llista els patrons reconeguts.*

Output en aquest punt:

*Cançó X = [1,2,1] [2,3,4]*

### 4.5 Creació de la base de coneixement.

Desenvolupar el model de dades requeria, per una banda, tenir dades emocionals, les quals s'han aconseguit en el punt 4.1.1 gràcies als usuaris i per altra banda característiques musicals que han estat les progressions harmòniques.

Ara que ja es disposa d'aquests dos elements, cal unir-los.

Com que en l'etiquetatge emocional (punt 4.1.1) s'ha aconseguit obtenir cançons representades per a la seva categoria emocional, ara només queda analitzar cada conjunt de cançons que formen una emoció, quins són els patrons de progressió harmònica que s'han trobat.

Els passos que s'han seguit s'indiquen a continuació:

### 1. Per a cada emoció del conjunt d'emocions:

- 1.1. Crear la llista buida (L) per emmagatzemar els patrons que representen a una emoció.
- 1.2. Per a cada cançó dins del conjunt:
  - 1.2.1. Emmagatzemar els patrons a L.

Output en aquest punt:

Emoció X = [1,2,1] [2,3,4]

Emoció Y = [5,1,5] [6,1,3] [3,2,1]

Emoció Z = [6,4,6] [1,2,5]

Cal afegir que en l'algorisme anterior, s'apliquen les tres versions d'obtenció de notes explicats en l'apartat 4.1.2. Això provoca obtenir tres llistes de patrons reconeguts per a cada versió. La motivació d'aquest fet és evitar obtenir conjunts buits de reconeixement de patrons ja que com es comenta anteriorment, la informació d' Spotify no sempre reflexa amb precisió els valors que haurien de ser.

### 4.6 Classificació KNN.

Ara que ja es disposa de l'element principal, les dades, es pot començar a desenvolupar l'objectiu principal del projecte. La classificació de cançons basades en emocions.

Per a dur a terme aquest classificador s'ha fet ús de l'algorisme KNN.[10] Aquest es defineix com a un algorisme supervisat que busca en les observacions més properes a la que s'està intentant classificar, el punt d'interès basat en la majoria de dades que el rodegen.

En aquest projecte els veïns són els patrons de progressió harmònica que segueix una cançó. Per tant, ara només cal comparar una cançó amb la base de coneixement que s'ha entrenat prèviament i votar els patrons reconeguts per a poder assignar-li una categoria o classe.

#### 1. Per a cada cançó del conjunt de test:

- 1.1. Anàlisi tècnic. (obtenció de patrons)
- 1.2. Cerca dels patrons trobat en les llistes de patrons associats a cada emoció.
- 1.3. L'emoció on els patrons reconeguts apareixen més vegades (votació), és el resultat de la classificació.

#### 2. Si el resultat és ambigu.

- 2.1. S'analitzen les emocions amb la matriu (fig 4) i s'atribueix com a resultat de la classificació, el resultat d'aplicar la matriu.

## 5. Resultats

Finalment després d'haver desenvolupat un model de dades que engloba les emocions i les progressions harmòniques d'acords. S'ha pogut desenvolupar un classificador que cerca característiques a la base de coneixement apresada durant l'entrenament, per poder classificar un nou objecte.

L'entorn on s'ha desenvolupat el test del resultat final ha estat mitjançant l'app VO, concretament l'apartat dissenyat exclusivament per a aquesta finalitat.

Malgrat la situació excepcional que s'ha viscut a causa de la pandèmia del Covid-19, només s'ha pogut instal·lar la app a uns quants usuaris, concretament a vuit. Aquests han valorat un total de 30 cançons que ha servit com a informació per entrenar el sistema. Pel que fa a la part de test, s'ha provat de classificar 70 cançons. Aquest contratemps o handicap, ha suposat violar un dels principis de la IA que és: El 70% de les dades han de servir per entrenar i el 30% per validar i en aquest cas ha estat al revés.

### 5.1 Resultats de la part emocional

Pel que fa les emocions els resultats obtinguts no es poden classificar en bons o dolents. La part emocional d'aquest treball és totalment subjectiva i personal. Si més no, s'ha observat una gran coincidència entre els usuaris alhora de valorar una cançó. Aquest fet ha comportat una bona fase d'etiquetatge facilitant la següent fase d'aprenentatge, l'anàlisi tècnica.

Cal comentar que la tolerància utilitzada per a la desició de les etiquetes ha estat del 80%. El 100% seria un model molt estricte el qual necessita que tots els individus valorin de la mateixa manera. El motiu d'aquest percentatge és que es dona un cert marge però segueix sent una bona majoria.

Tot i no haver disposat de gaires usuaris per a obtenir dades es pot afirmar que els resultats són significatius i a gran escala es podria aconseguir un bon sistema de classificació.

### 5.2 Resultats de la part tècnica

A l'hora d'utilitzar una informació per desenvolupar qualsevol producte, s'ha d'assegurar que aquesta és fiable i és adient per al que es vol dur a terme.

En aquest treball com s'ha anat comentant, s'ha fet ús de la plataforma Spotify per obtenir dades tècniques de les cançons.

Després de desenvolupar dues versions (V1 i V2) del procediment que cerca les notes i observant-ne els resultats, es va arribar a la conclusió que la informació oferta pel servei, no és sempre precisa i no concorda amb el que hauria de ser.[11]

Tot i encertar en la majoria de cançons, es va proposar introduir una última versió. Aquesta al ser totalment exhaustiva, ha ofert molt millors resultats alhora de trobar patrons harmònics.

Per anar sobre segur, en l'anàlisi de patrons s'ha treballat amb les tres versions simultàniament.

La V1 que feia referència a la mitjana d'informació relativa en un segon (valor enter), ha resultat escassos però acceptables amb el valor límit 0.5 (el millor de tots els que es va provar).

La diferència entre la primera versió i la segona és que, en la V2 tot i seguir agrupant la informació referent a un segon, ara s'utilitza el valor més gran per a descriure cada nota en aquell segon, el que significa que utilitzant valors límit  $\geq 0.8$  s'obté els resultats més fiables de totes les versions.

Per últim la V3, tot i introduir certa redundància, ha donat molt bons resultats amb el valor límit 0.5 ja que en les anteriors versions, s'agrupa més d'un conjunt de dades per a descriure un únic segon i en aquesta s'analitza cada conjunt d'informació independentment.

En algun cas el sistema no ha sigut capaç de trobar acords, de manera que aquella cançó no ha pogut ser classificada.

### 5.3 Resultats globals final

Els resultats finals de la classificació han estat força acceptables. Tot i no poder parlar d'excel·lència en aquests ja que són resultats que depenen de la percepció de la persona que els avalua, per a mi han estat molt encertats.

Per arribar als resultats finals s'ha estat provant l'execució de l'algorisme de classificació amb diversos valors per als paràmetres.

Aquestes variables han estat:

El valor llindar per decidir quan una nota hi és o no.

El grau de tolerància en la decisió de l'etiquetatge emocional.

## 6. Conclusions

Relacionar la música i les emocions és, a priori, una tasca molt laboriosa i que requereix una gran implicació a l'hora d'aprendre sobre els dos camps, la música i la psicologia.

Sent l'objectiu principal del projecte, no és una tasca impossible i s'ha demostrat amb els resultats que arribar a predir una emoció basant-se amb prèvies valoracions i patrons tècnics musicals es viable.

També s'ha aconseguit implementar la funcionalitat que es necessitava per a poder extreure les dades d'una manera més sistemàtica i no amb la pròpia App que és en construcció.

Tot i haver demostrat la capacitat d'aquest sistema per a deduir el resultat esperat, no sempre ha estat amb gran precisió i s'ha observat certes desviacions.

Per poder ser més precís i reduir el nombre d'errades en la classificació, faltaria comptar amb un conjunt de dades d'entrenament molt més ampli del que s'ha treballat.

Per altra banda, Spotify al processar les senyals físiques de les cançons i poder oferir dades al respecte, elimina informació rellevant que hauria ajustat molt millor els resultats en la part tècnica.

## 7. Futures Accions

Un dels objectius que finalment es va haver de refusar per falta de temps era l'implementació d'una xarxa

neuronal que fos capaç d'aprendre nous patrons de progressions harmòniques, de manera que l'aprenentatge i classificació fós dinàmic.

Tot i no ser un dels principals objectius del treball, s'ha començat a desenvolupar també una petita part que mostra en % el grau d'afinitat entre els usuaris que han valorat les cançons.

Juntament amb la resta del treball, s'acabarà de refinar, i s'introduirà com a funcionalitat en la aplicació.

## 8. Agraïments

En primer lloc, m'agradaria agrair a la meva família el suport incondicional que m'ha donat en tot moment durant els quatre anys que porto de carrera. Sense ells res de tot això hauria estat possible. També m'agradaria agrair a en Josep Lladós, el meu tutor del TFG, pels consells i l'ajuda que m'ha ofert durant aquests mesos de projecte.

Expressar també les gràcies a companys com en Sergi Duarte (estudiant de Tècnic de SO a ITES), en Guillem Bautista (estudiant graduat en Grau Superior en Interpretació de Violoncel, al Conservatori Professional de Música del Liceu), l'Àlicia Fornés (musicòloga i enginyera informàtica graduada a la UAB) i a la Lucila Martínez (estudiant de Psicologia a la UOC) per als coneixements i material aportat. Sense la seva participació i ajuda no hauria sigut capaç de desenvolupar aquest treball.

També agrair l'acompanyament de la meva parella i amics que han estat constantment donant-me suport i ajudant-me a reflexionar sobre el tema.

Gràcies.

## 9. Referències

- [1] <https://www.ee.columbia.edu/~dpwe/papers/Wang03-shazam.pdf>
- [2] <https://towardsdatascience.com/music-genre-classification-with-python-c714d032f0d8>
- [3] <http://www.psychologytoday.com/us/blog/hot-thought/201805/what-are-moods>
- [4] UOC. Grau en Psicologia. Mòdul Emoció. (PDF p.34)
- [5] [https://simple.wikipedia.org/wiki/List\\_of\\_emotions](https://simple.wikipedia.org/wiki/List_of_emotions)
- [6] <https://api.spotify.com/v1/audio-analysis/{id}>
- [6] <https://developer.spotify.com/documentation/web-api/reference/tracks/get-audio-analysis/>
- [7] <https://es.wikipedia.org/wiki/Acorde>
- [8] <https://www.acordespiano.com/lista/>
- [9] <https://www.24bitsescuelaproduccionmusical.com/10-progresiones-de-acordes-infalibles/>
- [9] <https://blog.landr.com/es/10-progresiones-de-acordes-emocionales/>
- [10] <https://www.aprendemachinlearning.com/clasificar-conk-nearest-neighbor-ejemplo-en-python/>
- [11] [http://docs.echonest.com/s3-website-us-east-1.amazonaws.com/\\_static/AnalyzeDocumentation.pdf](http://docs.echonest.com/s3-website-us-east-1.amazonaws.com/_static/AnalyzeDocumentation.pdf)

## 10. Annex

Imatge dels resultats obtinguts en la fase de test amb els

valors:

V1 = 0.5

V2 = 1

V3 = 0.5

K-usuaris = 5 ( 30/30 valoracions)

```
[('GOOD VIBES', 3)] Way_Down_We_Go - KALEO
[('GOOD VIBES', 0)] No_Filter - River_Omelet
[('GOOD VIBES', 0)] How_Does_it_Feel - DLMT
[('SADNESS', 1)] Lost_Tonight - Saje
[('GOOD VIBES', 2)] String_Quartet_No._6_in_G_Major,_Op._101:_III._Lento - Dmitri_Shostakovich
[('GOOD VIBES', 1)] War - Drake
[('SADNESS', 1)] Not_You_Too_(feat._Chris_Brown) - Drake
[('GOOD VIBES', 2)] Neither_Do_I - Stwo
[('GOOD VIBES', 0)] Ellas - Nach
[('NOSTALGIA', 1)] Sera - El_Canto_Del_Loco
[('GOOD VIBES', 0)] In_My_Place - Coldplay
[('GOOD VIBES', 0)] Joro - WizKid
[('GOOD VIBES', 0)] Mariage_D'Amour - George_Davidson
[('REFLEXION', 2)] Prelude_in_G_Minor - Yiruma
[('REFLEXION', 4)] Too_Young - Post_Malone
[('NOSTALGIA', 2)] Fado_boemio_e_vadio - Enrique_Amoroso
[('GOOD VIBES', 0)] Not_Nice - PARTYNEXTDOOR
[('GOOD VIBES', 1)] Fix_You - Coldplay
[('GOOD VIBES', 0)] All_I_Ask - Adele
[('GOOD VIBES', 0)] Baby_You're_A_Rich_Man_-_Remastered_2009 - The_Beatles
[('REFLEXION', 2)] Toosie_Slide - Drake
[('GOOD VIBES', 3)] Loco - Beéle
[('GOOD VIBES', 1)] Canary_Ass - Locoplaya
[('NOSTALGIA', 1)] Renee's_Song - Bazzi
[('SADNESS', 1)] Inolvidable - Ovy_On_The_Drums
[('SADNESS', 1)] Desde_cero_(con_Melendi) - Beret
[('SADNESS', 1)] Peces_de_Ciudad - Pablo_Alborán
[('MOTIVATION', 1)] Love_Galore_(feat._Travis_Scott) - SZA
[('SADNESS', 1)] HIGHEST_IN_THE_ROOM - Travis_Scott
[('GOOD VIBES', 0)] Won't_Be_Late_(feat._Drake) - Swae_Lee
[('GOOD VIBES', 1)] Diles - Malú
[('GOOD VIBES', 0)] Just_Can't_Get_Enough - Black_Eyed_Peas
[('SADNESS', 1)] Titanium_(feat._Sia) - David_Guetta
[('SADNESS', 1)] Lucky - Jason_Mraz
[('SADNESS', 1)] Precisamente_ahora - David_DeMaría
[('NOSTALGIA', 1)] Everything's_Gonna_Be_Alright - Sweetbox
[('GOOD VIBES', 0)] Cero - Dani_Martín
[('SADNESS', 2)] Peter_Pan - El_Canto_Del_Loco
[('GOOD VIBES', 3)] Turn_Me_On - Kevin_Lytle
[('GOOD VIBES', 0)] Broken_Strings - James_Morrison
[('GOOD VIBES', 0)] Payphone - Maroon_5
[('GOOD VIBES', 0)] before_I_close_my_eyes - XXXTENTACION
[('GOOD VIBES', 1)] Chispa_y_Oxígeno - Lágrimas_De_Sangre
[('GOOD VIBES', 1)] In_the_night - Oques_Grasses
[('GOOD VIBES', 0)] NOT_TOO_DEEP_(feat._6LACK) - India_Shawn
[('GOOD VIBES', 0)] Loco_Por_Llegar - Fuego
[('GOOD VIBES', 0)] Genesis - Daniela_Andrade
[('REFLEXION', 4)] Begging_You - Russ
[('SADNESS', 1)] BRACKETS - J._Cole
[('GOOD VIBES', 1)] Blue_World - Mac_Miller
[('GOOD VIBES', 1)] Intentions - Justin_Bieber
[('GOOD VIBES', 0)] Say_So - Doja_Cat
[('GOOD VIBES', 2)] Life_Is_Good_(feat._Drake) - Future
[('GOOD VIBES', 0)] Scared_to_Be_Lonely - Martin_Garrix
[('GOOD VIBES', 0)] Location - Khalid
[('GOOD VIBES', 0)] Lost_on_You - LP
[('GOOD VIBES', 0)] Signs - Drake
[('REFLEXION', 3)] Somebody - Internet_Money
[('GOOD VIBES', 0)] I_Can't_Help_(with_Sarcastic_Sounds) - Shoffy
[('GOOD VIBES', 0)] Only_Human - Jonas_Brothers
[('GOOD VIBES', 3)] BENDICIONES - Bad_Bunny
[('GOOD VIBES', 0)] Halftime - Young_Thug
[('GOOD VIBES', 0)] Nunca_Estoy - C._Tangana
[('SADNESS', 1)] ROCKSTAR_(feat._Roddy_Ricch) - DaBaby
[('GOOD VIBES', 1)] Adicto_(with_Anuel_AA_&_Ozuna) - Tainy
[('SADNESS', 1)] Sola - Manuel_Turizo
[('GOOD VIBES', 0)] everything_i_wanted - Billie_Eilish
[('GOOD VIBES', 0)] Elegí_(feat._Dimelo_Flow) - Rauw_Alejandro
[('GOOD VIBES', 0)] L.I.F.E._(feat._Kokane) - Cypress_Hill
[('GOOD VIBES', 0)] Qué_bonito_es_el_amor - La_Pegatina
[('NOSTALGIA', 1)] Nit_D'Agost - Els_Catarres
[('GOOD VIBES', 2)] Solve_Di_Problem_(feat._Lasai) - Fyahbwoy
[('GOOD VIBES', 0)] Movin'_On - Collie_Buddz
[('GOOD VIBES', 2)] Smokin'_Love - Stick_Figure
[('GOOD VIBES', 0)] Temps_Al_Temps - Tixín
[('GOOD VIBES', 1)] Lejos_de_Tí - Rels_B
[('GOOD VIBES', 0)] changes - XXXTENTACION
```