

Clasificarea melodiilor in functie de genurile muzicale

Lupu Elena Stoean Elena-Ramona

Grupa 322AC

Partea I

Descrierea problemei

1.1. Introducere

În ziua de azi, sunt numeroase servicii care distribuie consumatorilor muzica oferită de către diversi artiști. Se poate observa, în momentul în care descarcăm o melodie, că nu primim doar fișierul audio cu numele piesei și al artistului, ci și alte informații auxiliare, cum ar fi **genul** muzical de care aparține piesa respectivă. Din punct de vedere practic, aceste informații pot fi folosite pentru a face recomandări consumatorilor pe baza preferințelor lor.

1.2. "Vizualizarea" datelor

La prima vedere, pare dificil de manipulat un fișier audio. În realitate, există modalități prin care semnalul audio poate fi "vizualizat".

Spectogramele constituie reprezentări ale variației frecvenței și amplitudinii unui semnal audio dintr-un fișier specific. Astfel, putem spune că am convertit sunetul într-o imagine.

Reprezentarea grafică a undelor, în funcție de timp și amplitudine, este o altă modalitate de ilustrare în care se observă rapid anumite asemănări între semnale.

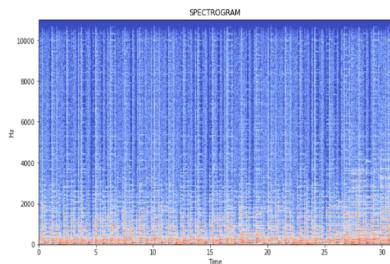


Figura 1: Spectogramă

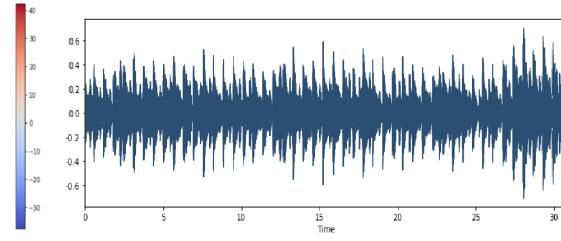


Figura 2: Reprezentare grafică

Vom vizualiza reprezentările grafice ale semnalelor, cât și spectogramele pentru 3 exemple proprii din genuri diferite, astfel conturându-se diferențe observabile între cele 3 semnale audio (pop, reggae, hiphop):

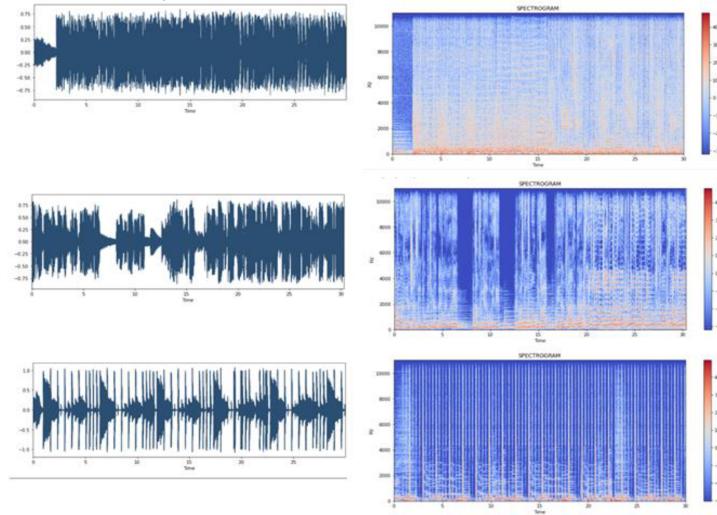


Figura 3: Reprezentările grafice și spectogramele pentru 3 genuri diferite

Acum realizăm spectogramele pentru mai multe exemple (tot proprii) din același gen:

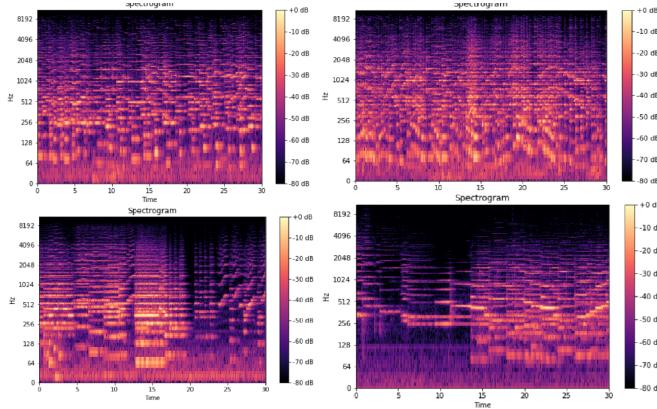


Figura 4: Spectogramele a 4 fragmente de muzică clasică

Așadar, problema noastră se poate aborda asemenea unei clasificări de imagini, pentru care există numerosi algoritmi.

1.3. Baza de antrenare

Setul de date de colectare a genurilor, GTZAN, este bine cunoscut și folosit pentru clasificarea genurilor muzicale și a fost creat de G. Tzanetakis și P. Cook. Autorii au scris o lucrare despre clasificarea genurilor și au creat un set de date special pentru aceasta, au colectat fișierele audio în anii 2000-2001 în timpul studiului lor din diverse surse, inclusiv CD-uri de muzică și emisiuni radio. Setul de date cuprinde 1000 de fișiere muzicale fiecare având o durată de 30 de secunde. Autorii săi au definit 10 genuri diferite, fiecare conținând câte 100 de piese audio. Fișierele au fost stocate în format .wav. Acestea sunt genurile muzicale principale:

- Blues
- Clasic
- Country
- Disco
- Hiphop
- Jazz
- Metal
- Pop
- Reggae
- Rock

Partea II

Algoritmul CMMMP

2.1. Cum funcționează algoritmul CMMP?

Metoda CMMP este folosită pentru crearea celor mai simple clasificatoare **binare**.

Ideea este să se construiască o funcție liniară:

$$f(v) = a^T v + b$$

unde $v \in \mathbb{R}^d$ este semnalul de antrenare, iar $a \in \mathbb{R}^d$ și $b \in \mathbb{R}$ sunt parametrii necunoscuți ai funcției.

Pentru găsirea acestor parametrii, trebuie satisfăcută relația:

$$f(v) = \begin{cases} 1 & , daca \quad v \in C_1 \\ -1 & , daca \quad v \in C_2 \end{cases}$$

Evident că relația nu poate fi satisfăcută exact, deoarece semnalele pe care le vom trimite nu vor fi identice cu cele din baza de antrenament. Din acest motiv, vom căuta o soluție în sens CMMP.

Dacă adăugăm un vector coloană 1_n obținem un sistem CMMP.

Având acum vectorii de antrenament $(v_1, 1), \dots, (v_{n_1}, 1), (v_{n_1+1}, -1), \dots, (v_{n_1+n_2}, -1)$, construim:

$$\begin{pmatrix} v_1^T & 1 \\ \vdots & \\ v_{n_1}^T & 1 \\ v_{n_1+1}^T & 1 \\ \vdots & \\ v_{n_1+n_2}^T & 1 \end{pmatrix} \begin{pmatrix} w \\ b \end{pmatrix} = \begin{pmatrix} 1_{n_1} \\ -1_{n_2} \end{pmatrix}$$

Din relația de mai sus obținem valorile optime pentru w și b . Clasificarea se va face în felul următor:

$$v \in \begin{cases} C_1 & , daca \quad \text{sign}(w^T v + b) = 1 \\ C_2 & , daca \quad \text{sign}(w^T v + b) = -1 \end{cases}$$

2.2. Cum se aplică aici metoda CMMP?

Cum am precizat mai sus, această metodă este folosită pentru crearea de clasificatoare **binare**, dar în cazul nostru avem 10 clase.

Această problemă se rezolvă ușor. Se poate extrapola algoritmul pentru căte clase vrem. Pentru fiecare gen muzical se poate rezolva un sistem diferit în sens CMMP pentru a realiza clasificarea (dacă piesa aparține genului respectiv sau nu). Diferența între sisteme este dată de coloana y (dacă scriem sistemul sub forma $Ax = y$), unde atribuim eticheta 1 genului pe care îl verificăm și eticheta -1 la restul.

Așadar, vom proceda în felul următor (cum este sugerat și în figura 5):

- Creăm spectogramele fiecărui fragment din GTZAN;
- Toate acestea se stochează sub formă de vectori de caracteristici (căte unul pentru fiecare fragment), cu tot cu etichetele corespunzătoare;
- Se scoate coloana cu etichetele și se adaugă coloana 1_n , obținând astfel matricea de caracteristici;
- Pe baza coloanei originale de etichete se creează pentru fiecare sistem CMMP vectorul y cu 1 și -1;
- Se realizează primul sistem, iar dacă datele noastre de intrare se încadrează în genul respectiv, algoritmul se oprește, altfel repetă procedura pentru următorul gen...:

2.3. Avantaje/Dezavantaje

Avantaje

- Este relativ ușor de implementat în comparație cu alți algoritmi mai eficienți, dar mai sofisticăți;

Dezavantaje:

- Este foarte lent, o rulare poate dura până la 1 minut, deoarece rezolvă până la 10 sisteme CMMP, care este operație complexă și costisitoare;
- Are precizie mică.

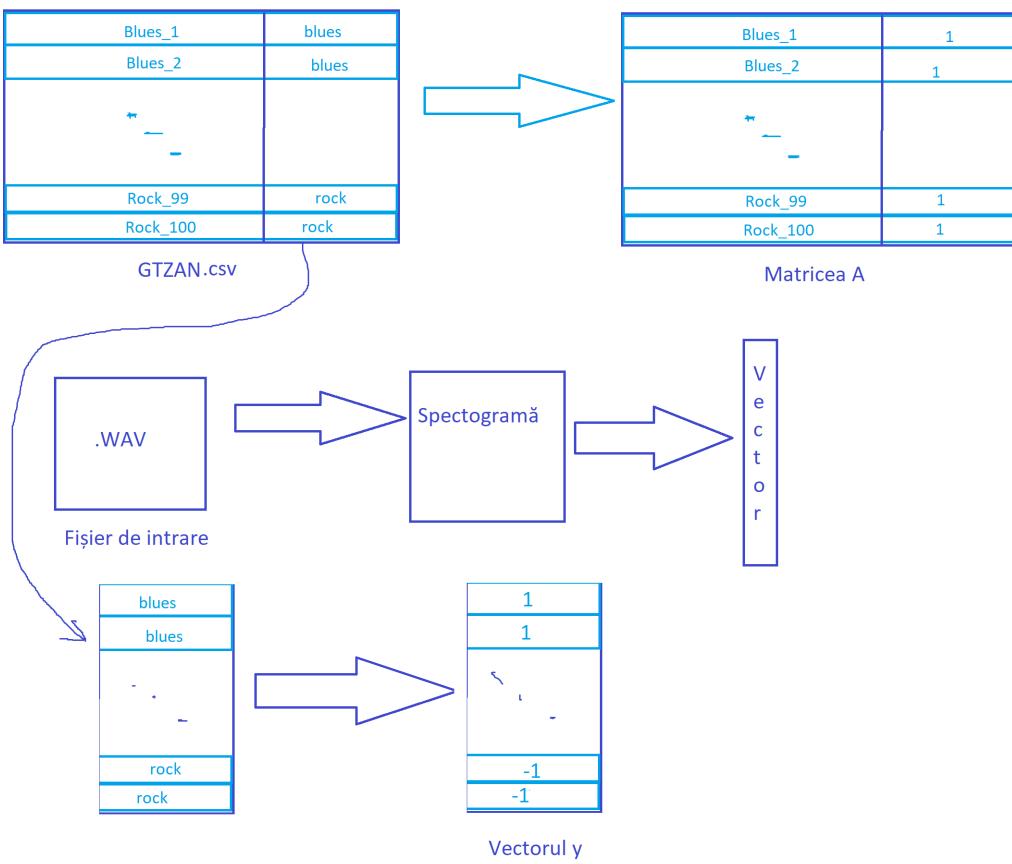


Figura 5: Prelucrarea datelor

Partea III

Algoritmul KNN

3.1. Ce este KNN?

Algoritmul se concentrează în principal pe 2 lucruri:

1. Analiza Componenelor Principale (PCA);
2. Digrama de dispersie.

KNN încercă să găsească distanța dintre un punct de referință de pe digrama de dispersie și celelalte puncte. Este o tehnică de **Machine learning** care preia ca date de intrare puncte de orice fel și calculează distanța dintre cele k puncte de date pre-etichetate. Clasificarea se bazează pe distanța obținută de la cele mai apropiate puncte de date.

Pentru a determina asemănările dintre melodii, distanța euclidiană este necesară pentru a testa dimensiunea care poate fi folosită ca interpretare a apropierii dintre 2 puncte.

3.2. Cum se aplică aici metoda KNN?

Sunt câțiva pași foarte simpli din punct de vedere matematic:

- Se alege numărul K al vecinilor; în general, se alege astfel încât K și numărul de clase să fie prime între ele pentru a evita erorile;
- Se calculează distanța euclidiană a celor K vecini:

$$d_{euclidian}(x, y) = \sqrt{\sum_{i=1}^N (x_i - y_i)^2}$$

- Se iau K cei mai apropiati vecini conform distanței euclidiene calculate;
- Se numără numărul de puncte date din fiecare categorie, printre cei K vecini;
- Se atribuie noile puncte date acelei categorii pentru care numărul de vecini este maxim.

3.3. Avantaje/Dezavantaje

Avanteje

- Este simplu de implementat;
- Este robust la datele de antrenament zgomotoase;
- Poate fi mai eficient dacă datele de antrenament sunt mai mari.

Dezavantaje:

- Tot timpul trebuie determinată valoarea K, fapt ce poate fi dificil câteodată;
- Are precizie mică, în comparație cu alții algoritmi mai sofisticăti;
- Costul de calcul este mare din cauza calculării distanței dintre punctele de date pentru toate eșantioanele de antrenament.