2014

Réalisé par:

CHION Romain
KETELS Martin

TRAITEMENT ET ANALYSE DES DONNEES SONORES ET VISUELLES (MOD 7.3)

BE3 : CLASSIFICATION DE LA MUSIQUE EN TYPES D'EMOTION

Table des matières

Table des illustrations

1 Introduction

L'objectif de ce BE est de développer différentes méthodes afin de systémiser et d'automatiser la classification de musiques par types d'émotion. On procédera ensuite à un certain nombre d'expérimentations pour les évaluer. Nous partons d'un ensemble de données consistent en plusieurs extraits de musique classique d'une durée de 4 secondes au format wav. Ces extraits sont regroupés dans 4 répertoires en fonction de l'émotion qui leur est associée : l'anxiété, la dépression, la satisfaction et l'exubérance

2 Extraction de descripteurs

Pour représenter les signaux audio selon des critères quantifiables et objectifs, nous utilisons plusieurs descripteurs. Ces descripteurs nous permettrons de les reconnaitre et de les classifier. La redondance est un outil vastement utilisée pour améliorer les algorithmes : l'emploi de plusieurs descripteurs ralenti les calculs mais facilite une classification plus claire. Dans le cadre de ce bureau d'étude on a choisi neuf descripteurs différents. Ces descripteurs pourront être temporels, fréquentiels, énergétiques… La fonction Matlab de création de ses descripteurs est ***descripteurs.m***

function ***[centroides,etendue,asymetrie,kurtosis,ptcoupure,centroidet,energieglob,deff,zcr] = descripteurs()***

Cette fonction ne prend pas d'arguments puisque l'ensemble des extraits musicaux est contenu en ***flat*** dans plusieurs répertoires du dossier Matlab. Elle retourne une matrice dont les colonnes correspondent aux différents descripteurs pour chacun des extraits.

Après ***initialisation*** des données et récupération des fichiers. On parcoure la liste de ces fichiers pour en calculer les descripteurs. Parmi les ***descripteurs spectraux***, on retrouvera grâce au ***calcul du spectre*** par une transformée de Fourier, les ***centroïdes spectraux***, les ***étendues***, les ***asymétries***, les ***Kurtosis*** et les ***points de coupure*** de tous les échantillons. Pour cela j'utilise des ***boucles for*** et ***while*** afin de faire des calculs sur l'extrait échantillonné. Parmi les ***descripteurs temporels***, on retrouvera, grâce au ***calcul de l'enveloppe*** par des moyennes quadratiques sur une fenêtre temporelle avec recouvrement, les ***centroïdes temporels***, les ***énergies globales***, les ***durées effectives***. Enfin un ***descripteur brut*** sera calculé à travers le ***taux de passage par zéro***. Les algorithmes utilisés sont extraits des méthodes de calcul présentées dans le poly. Un maximum d'automatismes a été utilisé comme la fonction ***rms*** pour le calcul des moyennes quadratiques ou ***fft*** pour les transformées de Fourier.

m_descripteurs = [centroides,etendue,asymetrie,kurtosis,ptcoupure,centroidet,energieglob,deff,zcr];

*save****('classe','classe');***

*save****('data','m_descripteurs');***

Les descripteurs calculés sont enregistrés dans le fichier ***data.mat*** après un long temps de calcul afin de pouvoir les réutiliser par la suite sans avoir à relancer la fonction et les calculs. En effet ces données ne changent pas, dans le cas d'analyses de nouveaux morceaux, il suffira d'ajouter des lignes aux données.

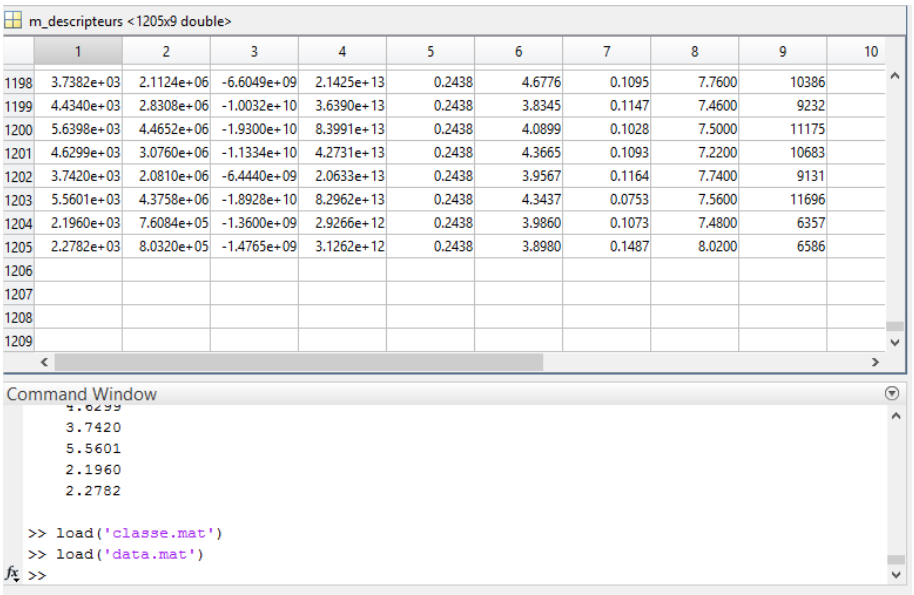

Figure 1 : Résultats sur la console après lancement de descripteurs.m

Les données récupérées sont cohérentes avec les attentes que ce soit en ordre de grandeur ou après comparaison des sets de données d'autres groupes. En réutilisant la fonction ***descripteur.m*** on crée une fonction ***description.m*** permettant de récupérer les données pour un unique extrait musique dont le ***path*** est utilisé en argument. Cette fonction nous permettra de créer des données « nouvelles » à utiliser pour tester la classification.

3 Classification en types d'émotion

L'objectif de cette partie est de développer le système de classification à partir des données de descripteur à disposition, afin de déterminer de manière automatique le type d'émotion associé aux titres musicaux recherchés. Ces derniers sont alors représentés par leur vecteur de descripteurs calculés précédemment grâce à la fonction ***description.m***.

Nous nous intéresserons ici à plusieurs approches qui ont été développées, codées et décrites, dans la fonction ***classification_audio.m***:

- la première basée sur la théorie bayésienne de la décision
- la seconde basée sur l'algorithme des K plus proches voisins avec distance euclidienne
- la troisième sur l'algorithme des K plus proches voisins avec distance Mahalanobis

Plusieurs problèmes se posent alors, la matrice de covariance calculée grâce à la fonction ***cov*** n'est pas inversible à cause des approximations numériques. Pour pallier à ces problèmes, on a utilisé la fonction ***pinv*** qui prend en comptes les approximations dans ses calculs.

Après test sur plusieurs échantillons, les trois méthodes de classifications donnent pour des tests ponctuels sur des échantillons uniques, trois résultats de classes différentes.

Cependant les résultats s'homogénéisent après utilisation d'un test généralisé par la fonction ***test.m***. On utilise ici 100% des données en apprentissage et 100% des données en test.

```
result =

   628
   628
   628


ans =

    0.5212
    0.5212
    0.5212
```

Figure 2 : Nombres de bonnes classifications pour un nombre n=1205 d'échantillons

On trouve alors 52,1% de bonnes classifications quel que soit la méthode utilisée. Cette performance est relativement faible mais supérieure à une répartition aléatoire qui obtiendrait en moyenne 25% de bonnes classifications. L'uniformité des résultats est due l'utilisation de 100% des données en apprentissage et en test. Afin de déterminer une méthode décisionnelle HCE (classification hiérarchique), on mesure ensuite les taux de réussites suivant les différentes classes.

```
result =

   122   256   160    90
   122   256   160    90
   122   256   160    90


ans =

    0.4841    0.6845    0.4734    0.3734
    0.4841    0.6845    0.4734    0.3734
    0.4841    0.6845    0.4734    0.3734
```

Figure 3 : Bonnes classifications par émotions

On retrouve ici des résultats très concluants, ainsi on trouve 68,4% de réussite pour l'émotion 2 de ***Contentement***. Il faudrait donc par la suite modifier le programme ***classification_audio.m*** afin de prendre en compte une classification hierarchique des échantillons selon le modèle:

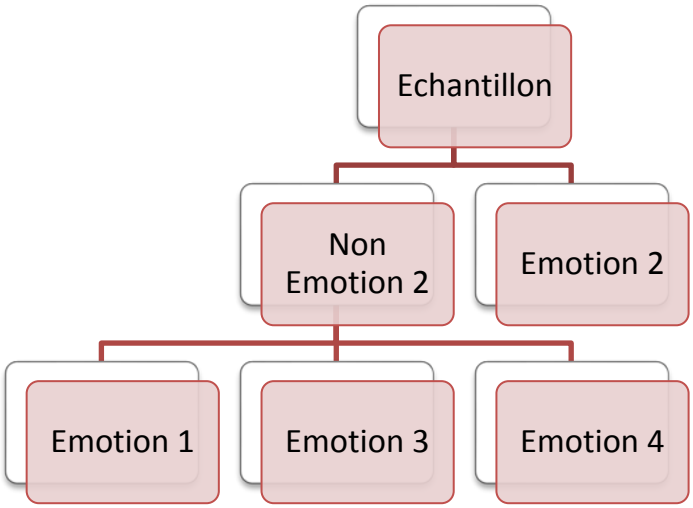

Figure 4 : Modèle de classification hiérarchique Ad Hoc

Après une nouvelle série de test on pourrait alors améliorer la classification hiérarchique entre les émotions 1, 3 et 4.

4 Conclusion

De nombreuses améliorations sont possibles. Premièrement dans ces différentes méthodes on utilise la totalité des 9 classificateurs de manière égale. Il faudrait étudier les performances de différents couples de classificateurs et plus particulièrement en les pondérant (notamment dans les différentes distances où les gros classificateurs vont rapidement l'emporter) afin d'optimiser les performances de classification.

Un autre problème est l'utilisation de la matrice de covariance.

```
C =

   1.0e-18 *

    0.0000    0.0000   -0.0000   -0.0000   -0.0000    0.0000    0.0000    0.0000    0.0000
    0.0000    0.0000   -0.0001   -0.0000   -0.0000    0.0000    0.0000    0.0000    0.0000
   -0.0000   -0.0001    0.3243    0.0000    0.0000   -0.0000   -0.0000   -0.0000   -0.0000
   -0.0000   -0.0000    0.0000    0.0000    0.0000   -0.0000   -0.0000   -0.0000   -0.0000
   -0.0000   -0.0000    0.0000    0.0000    0.0000   -0.0000   -0.0000   -0.0000   -0.0000
    0.0000    0.0000   -0.0000   -0.0000   -0.0000    0.0000    0.0000    0.0000    0.0000
    0.0000    0.0000   -0.0000   -0.0000   -0.0000    0.0000    0.0000    0.0000    0.0000
    0.0000    0.0000   -0.0000   -0.0000   -0.0000    0.0000    0.0000    0.0000    0.0000
    0.0000    0.0000   -0.0000   -0.0000   -0.0000    0.0000    0.0000    0.0000    0.0000
```

Figure 5 : Inverse de la matrice de covariance pour la classification

La matrice obtenue est a priori surprenante. Les coefficients ayant étés arrondis à la fois par le calcul de la covariance (fonction ***cov***) et de l'inversion (fonction ***pinv***) on obtient des valeurs extrêmement petites (1.0e-18) voire pour la plupart nulles. On pourra chercher à améliorer ces fonctions pour de meilleurs résultats.

Avec des premières performances à 52.1% de bonnes réponses et les améliorations tant au niveau de la classification hiérarchique qu'au niveau de l'utilisation des descripteurs, il est certain d'obtenir des résultats bien meilleurs.