BE reconnaissance des formes

Veysseire Daniel Fabre Michaël

Université Paul Sabatier

11 novembre 2014

Résumé

Cet article vise à comparer l'efficacité de deux méthodes de classification (méthode de classification par loi normal multidimensionnel et méthode des K Plus Proche voisin), ainsi que les choix de paramétrisation des données (FFT, cepstre, MFCC), principalement dans le cadre de la reconnaisance de la parole.

Dans un premier temps, nous ferons une présentation théorique de ces méthodes et paramétrisations. Dans un deuxième temps nous présenterons le protocole expérimental mis en place afin de comparer leurs éfficacités.

Nous intèrprèterons ensuite les résultats obtenus puis nous finirons par une conclusion sur l'efficacité des différentes méthodes et paramétrisations.

Mots Clef

Methodes de classification, reconnaisance de la parole, loi normale, K plus proche voisins, paramétrisation, FFT, Cepstre, MFCC, apprentissage supervisé.

Abstract

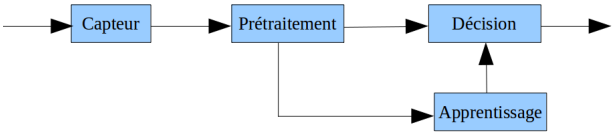
This paper aims to compare the efficiency of two methods of classification (method of classification with normal distribution multidimensional and Nearest neighbor search (NNS)), and the choice of parameterization (FFT, cepstrum, MFCC), mainly in the context of the speech recognition. Primary, we will make a theoretical presentation of these methods and parameterizations. in Secondly, we present the experimental protocol implemented to compare their efficiencies. Finally we interpret the results then finish with a conclusion on the efficiency of these different methods and parameterizations.

Keywords

methods of classification, speech recognition, normal distribution, Nearest neighbor search, NNS, parameterization, FFT, Cepstrum, MFCC, Supervised learning.

1 Introduction

La reconnaissance automatique de la parole est une technique informatique qui permet d'analyser un signal de parole. Voici un schéma qui illustre les différentes étapes de la chaîne de reconnaissance.



Dans le cas qui nous intéresse ici le capteur est un microphone, il transforme le signal physique en un signal numérique. Le prétraitement est une représentation allégée du signal numérique. Il consiste à réduire la dimension de l'espace, décoréler les paramètres et rechercher les paramètres discriminants. La décision assigne une classe à un vecteur par rapport aux formes acquises par apprentissage. L'apprentissage est constitué de références type. Dans notre cas l'apprentissage est supervisé car nous connaissons le nombre de classe et nous savons à quelle classe appartiens quel vecteur.
On se place ici dans le cas où on classifie chaque syllabe individuellement. On dispose d'une référence de 1000 éléments sonore de 64ms échantillonés à 16KHz et quantifiés sur 16 bits. On a ainsi 100 échantillons pour chacune des dix syllabes suivantes :
[ɑ],[e],[ɛ],[ə],[ɪ],[ø],[ɔ],[o],[u],[y]
correspondant aux classes :
'aa','ee','eh','eu','ii','oe','oh','oo','uu','yy' ;

2 paramétrisations et méthodes

2.1 Les différentes paramétrisations

Nous allons utiliser différentes paramétrisations des données et les comparer pour ne conserver que celles qui offrent les meilleurs résultats.

Transformé de Fourrier Rapide (FFT)
La transformée de Fourrier Rapide est un algorithme permettant de traiter un signal afin d'obtenir son spectre. Le

spectre d'un signal nous fournit l'intensité de chacune des plages de fréquences pendant un intervalle de temps t. Elle s'effectue sur un certain nombre de points ; augmenter ce nombre de points diminue la taille des plages de fréquences, et augmente le nombre de plages. On ne garde que la valeur absolue du résultat pour ne pas manipuler des nombres complexes.

En générale on effectue plusieurs FFT sur le signal partitionné, à l'aide de fenêtres glissantes, afin d'obtenir l'intensité des fréquences à plusieurs instants t. Puis on utilise des algorithmes comme le DTW (Dynamic time warping). Mais dans le cas présent dans cette étude, les échantillons sont extremement courts (64ms avec une fréquence d'échantillonage de 16KHz). Utiliser une fenêtre glissante ne s'avère pas nécessaire. On est donc dans un cas simplifié, on ne cherche qu'à comparer des voyelles prononcées dans un temps très court. Une simple FFT sur tout le signal est donc suffisante, on obtient ainsi un vecteur de taille variable selon le nombre de point sur lesquels on a réalisé la FFT. On comparera par la suite ces vecteurs entre eux (e.g par distance euclidienne). On effectue souvent un lissage du signal par Hamming lorsqu'il y a un recouvrement de fenêtre pour éviter de trop grandes discontinuités entre les fenêtres. Il n'est pas nécessaire de faire un lissage par Hamming ici, puisqu'on n'a pas utilisé de fenêtres glissante.

Le cepstre et les MFCC

Le cepstre est obtenu à partir du spectre. On effectue la transformée inverse du logarithme de la transformée de Fourrier (ou spectre) obtenu précedement. En pratique on ne garde que la valeur absolue du résultat. On obtient ainsi une transformation du signal dans un domaine analogue au domaine temporel. "Les MFCC (Mel-Frequency Cepstral Coefficients) sont des coefficients cepstraux calculés par une transformée en cosinus discrète appliquée au spectre de puissance d'un signal. Les bandes de fréquence de ce spectre sont espacées logarithmement selon l'échelle Mel" (wikipédia). Les MFCC sont proches du cepstre, mais diffèrent par l'utilisation de l'échelle Mel, échelle basée sur la perception humaine. Pour calculer ces MFCC j'ai utilisé la fonction MELCEPST disponible sur la toolbox voicebox. Elle réalise une RFFT (DFT of real data, DFT = Discrete Fourier Transform) sur le signal lissé par une fonction hanning adapté à la fréquence d'échantillonage. Une fois la DFT appliqué, on multiplie la partie réel avec la partie conjugué obtenue.

On applique ensuite sur ces données obtenues après passage à l'échelle MEL(à l'aide de la fonction MELBANKM qui sert à calculer la matrice de passage à l'échelle MEL), un log adapté à la valeur max des données . Puis on refait une RDCT (Discrete cosine transform of real data). On obtiens ainsi les différents coefficients cepstraux de Mel (ce sont des vecteurs).

2.2 Les différentes méthodes de classifications

Comme dit précédement nous allons comparer les deux méthodes de classifications. Pour classifier des données, il faut effectuer au préalable un apprentissage supervisé à partir de données de références. Il y a donc une phase d'apprentissage et une phase de reconnaissance.

classification par loi normale multidimensionnel
Pour utiliser la méthode de classification par loi normale (ou loi gaussienne) multidimensionnel, on suppose que chacune des composantes des vecteurs obtenus par paramétrisation suit une distribution aléatoire. Cette classification prend en paramètre la moyenne et la matrice de variance-covariance des données d'apprentissage. La matrice de variance-covariance est une matrice carrée de taille N*N (N le nombre de composante du vecteur). Chaque élément placé ligne i et colonne j dans la matrice vaut cov(Xi, Xj) avec Xi la ieme composante du vecteur. Ainsi sur la diagonal se situent les variances de chaque composante du vecteur. La covariance se calcule à l'aide de la formule suivante :

$$\text{Cov(x,y)} = \text{E(XY)} - \text{E(X)E(Y)}$$

La matrice de covariance permet de prendre en compte l'éloignement des données à la moyenne, leur dispersion.

On utilise ensuite la règle du maximum de vraissemblance pour la décision. Si on observe y et qu'on nomme k les 10 classes :

$$c^* = \underset{k_i}{\text{argmax}}\ P(k_i/y) \tag{1}$$

En effet on observe y et on choisit donc la classe la plus probable. En utilisant Bayes :

$$c^* = \underset{k_i}{\text{argmax}}\ \frac{P(y/k_i)P(k_i)}{P(y)} \tag{2}$$

Or P(y) est une constante et $P(k_i)$ = 1/10 car on a 10 classes considérées comme équiprobable. On peut donc simplifier l'équation en :

$$c^* = \underset{k_i}{\text{argmax}}\ P(y/k_i) \tag{3}$$

En utilisant la loi gaussienne multidimenssionelle on obtient :

$$c^* = \underset{k_i}{\text{argmax}}\ \frac{exp((-1/2)(y-\mu_i)^t\Sigma_i^{-1}(y-\mu_i))}{2\pi^{d/2} \mid \Sigma_i \mid^{1/2}} \tag{4}$$

Avec d la dimension des données, Σ_i matrice de variance-covariance de la classe i et μ_i vecteur moyenne de la classe i. On suppose par la suite que $|\Sigma_i| \neq 0$.
Le terme $2\pi^{d/2}$ étant une constante positive, on peut simplifier l'équation puis appliquer le logarithme. Le log étant

une fonction croissante et continue sur R^+, en l'utilisant sur la formule (on peut car exp(x) >0) on obtient :

$$c^* = \underset{k_i}{\text{argmax}}\ log(\frac{1}{\mid \Sigma_i \mid^{1/2}}) - (1/2)(y-\mu_i)^t \Sigma_i^{-1}(y-\mu_i) \quad (5)$$

$$c^* = \underset{k_i}{\text{argmax}}\ -(1/2)log(\mid \Sigma_i \mid) - (1/2)(y-\mu_i)^t \Sigma_i^{-1}(y-\mu_i) \quad (6)$$

la fonction y=-(1/2)x étant décroissante et continue sur R, en divisant la formule par -1/2 on arrive à

$$c^* = \underset{k_i}{\text{argmin}}\ log(\mid \Sigma_i \mid) + (y-\mu_i)^t \Sigma_i^{-1}(y-\mu_i) \quad (7)$$

C'est cette formule (7) qui sera appliqué pour la décision.

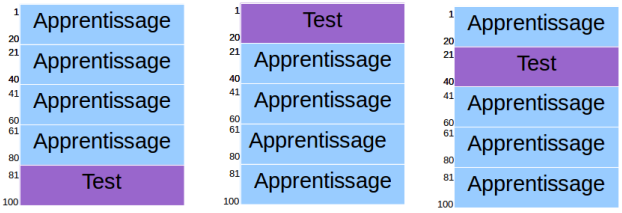
classification par les K plus proche voisin
La méthode de classification des K Plus proche voisin est relativement simple. Pour chaque vecteur de test, nous allons trier tous les vecteurs d'apprentissages en fonction de leur distance à ce vecteur. Puis on ne conserve que les K plus proche (K étant un entier choisis au préalable), et si une classe est représenté majoritairement on attribue cette classe à ce vecteur. Sinon on rajoute à la liste les prochains vecteur d'apprentissage les plus proches jusqu'à obtenir une classe majoritaire.

3 protocole expérimental

Nous allons utiliser une méthodologie simple et naturel pour comparer nos résultats. Du fait du manque de données (nous n'avons qu'un échantillon de 100 signaux pour chacune des 10 classe) nous utiliserons la validation croisé et rappelerons brièvement son principe.

3.1 principe de la validation croisée

Nous allons réaliser une validation croisé. C'est à dire que nous allons faire varier nos échantillons d'apprentissage et nos échantillons de test et faire une moyenne avec les résultats trouvés. Notre premier échantillon d'apprentissage est composé des 80 premiers échantillons de chaque classe et les 20 restants pour les tests. On refait une validation mais cette fois ci en prenant les échantillons de 21 à 100 pour l'apprentissage et 1 à 20 pour les tests. Puis 41 à 100 et 1 à 20 pour l'apprentissage et 21 à 40 pour les test, et on continue ainsi en décalant de 20.



Ensuite on fait une moyenne sur les résultats obtenus.

3.2 pour la loi normale

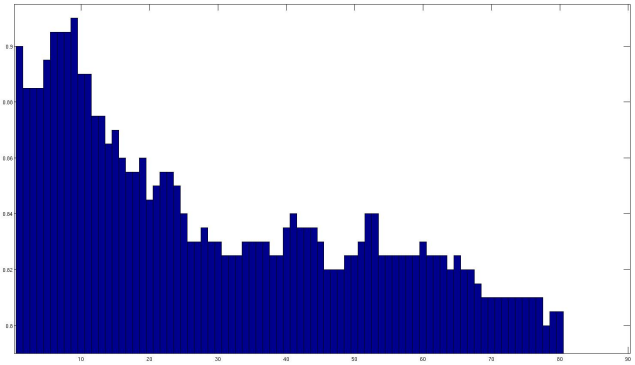
On effectue des tests sur chaque paramétrisation pour évaluer l'efficacité de la classification par loi normale. On effectue ainsi 15 tests car on a découpé les données en 5 choix d'échantillonage possible comme expliqué ci-dessus.

	Ech.1	Ech.2	Ech.3	Ech.4	Ech.5	Moyenne
FFT	99.5	95.5	96.5	97.5	95.5	96.9
cepstre	92	93	94.5	97	95	94.3
MFCC	100	100	100	100	100	100

On observe que la loi normale offre une moyenne de 96,9% de réussite avec la paramétrisation FFT, 94,3% avec le cepstre, et 100% avec les MFCC.

3.3 pour les-K Plus Proche Voisin

Il y a un paramètre en plus à prendre en compte pour les KPPV. La valeur de K correspondant au nombre de voisins observés. En générale K ne doit pas être trop grand, sinon le taux de reconnaissance diminue beaucoup. Exemple sur le graphique ci-dessous obtenu avec le spectre en paramétrisation. K est en abcisse et le taux de reconnaisance est en ordonnées.



On choisit donc une valeur de K arbitraire, on prend ici K=3.
On obtient les résultats suivants :

	Ech.1	Ech.2	Ech.3	Ech.4	Ech.5	Moyenne
FFT	94	79,5	97.5	99	99	93.8
cepstre	88.5	93	96	97.5	97.5	94.5
MFCC	100	100	100	100	100	100

On observe que la loi normale offre une moyenne de 93,8% de réussite avec la paramétrisation FFT, 94,5% avec le cepstre, et 100% avec les MFCC.

3.4 interprétation des résultats

Les MFCC sont redoutablement efficace et ont bien classé toutes les observations dans chaque situation. D'après le résultat de notre évaluation, la méthode de classification par loi normal multidimenssionelle offre de meilleurs résultats avec la transformée de fourrier qu'avec la paramétrisation cepstral, alors que la méthode de classification des K plus proche voisin offre de meilleur résultat pour la paramétrisation cepstral. Cependant les résultats peuvent être faussé, on observe dans le cas de K Plus proche voisins que pour l'échantillon 2, la paramétrisation par transformée de Fourrier a produit un résultat anormalement décevant par rapport aux autres échantillons, ce qui a descendu de beaucoup la moyenne finale, alors que pour le même échantillon, la

paramétrisation cepstral a fournit de bons résultats. Dans la majorité des cas la parémétrisation FFT semble être plus efficace que la paramétrisation cepstral.

4 conclusion

On ne peut pas encore se prononcer sur quelle méthode de classification est meilleur que telle autre car nous ne disposons pas d'un échantillon assez grand. En revanche, on peut affirmer que la paramétrisation par MFCC est très efficace dans ces conditions d'évaluation peu importe la méthode de classification, elle n'a même souffert d'aucune erreur. Nous nous sommes placé, dans cet article, dans des conditions idéales, tous les échantillons sonores font la même taille, et seuls des voyelles sont identifiées. Il pourait être intéressant de tester la parametrisation MFCC avec des conditions moins idéales pour tester ses limites.

nous contacter

Pour tout commentaires, erreurs, remarques, ajouts à faire, etc. Veuillez nous contacter à ces adresses :

wedg@hotmail.fr (Veysseire Daniel)
mickaelfabre@free.fr (Fabre Mickael)