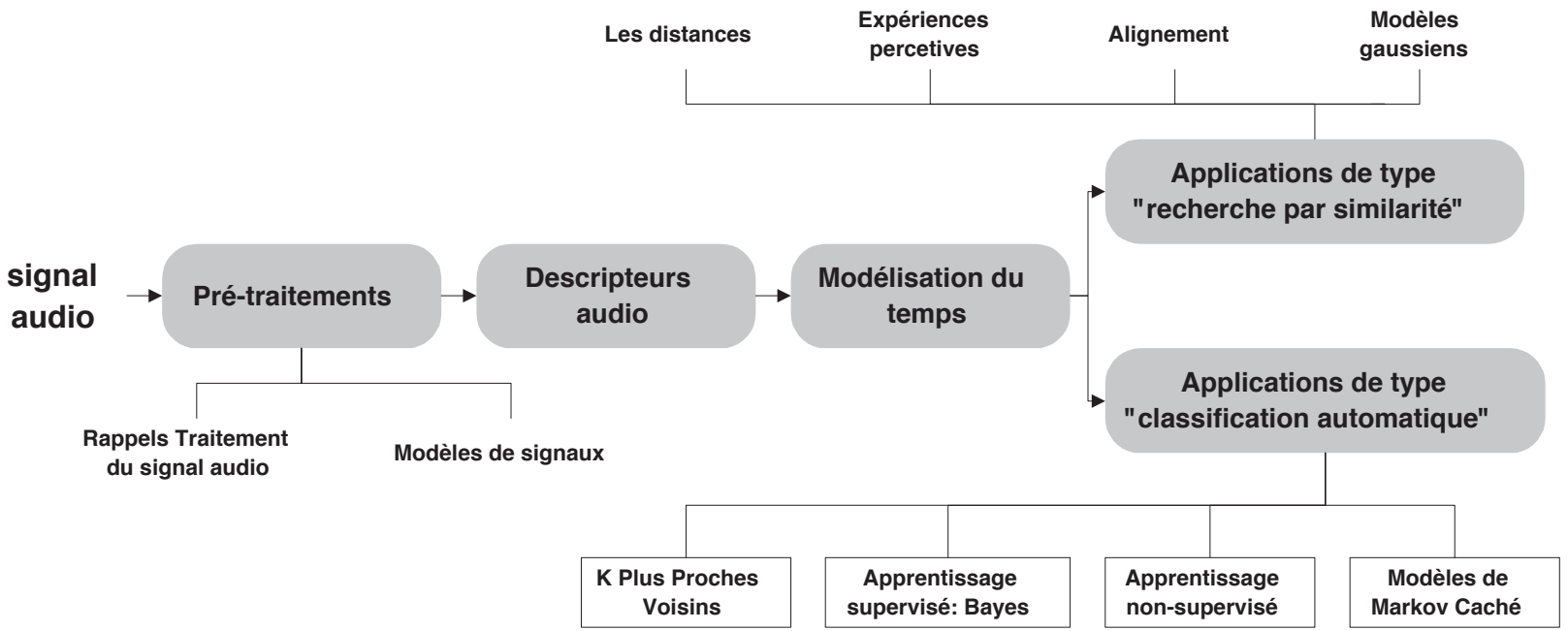
NSY122: Extraction d'information audio (partie 2) - peeters@ircam.fr

Geoffroy.Peeters@ircam.fr

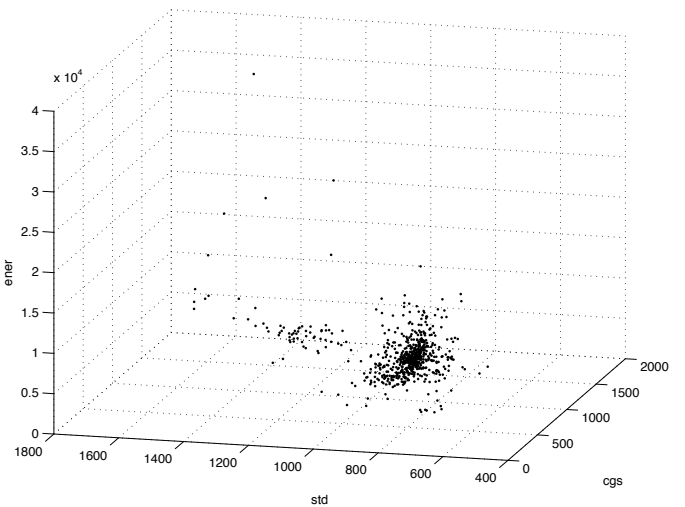
Plans


Les distances
Expériences percetives
Alignement
Modèles gaussiens
Applications de type "recherche par similarité"
signal audio
Pré-traitements
Descripteurs audio
Modélisation du temps
Applications de type "classification automatique"
Rappels Traitement du signal audio
Modèles de signaux
K Plus Proches Voisins
Apprentissage supervisé: Bayes
Apprentissage non-supervisé
Modèles de Markov Caché

Les distances

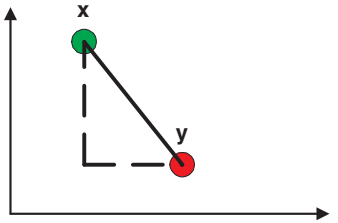
Notations:

- O: une observation (une trame d'un signal audio, tout un son, tout un morceau de musique),
  - O_l la l^{em} observation
- D: un descripteur extrait d'une observation
  - D_i le i^{em} descripteur
- On peut représenter l'ensemble des descripteurs audio $D_1, D_2, ...D_I$ extrait pour la donnée O_l
  - comme un vecteur $\underline{D}$ de dimension I
  - comme un point dans un espace à I dimensions (chaque dimension correspond à un descripteurs audio i)
  - on représente les différentes observations $O_1, O_2, ...O_L$ comme autant de points dans l'espace à I dimensions
- Comment calculer la distance entre deux points ?



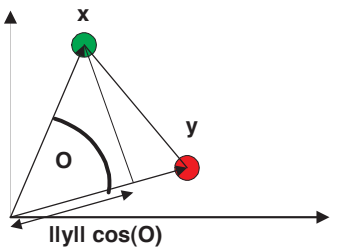
Les distances

- Distance de Minkowski:
  $D(x,y) = \left(\sum_{i=1}^{I}(x_i - y_i)^p\right)^{1/p}$
  - $p = 1$: distance de Manhatan: $\sum_i |x_i - y_i|$
  - $p = 2$: distance Euclidienne: $\sqrt{\sum_i (x_i - y_i)^2}$
  - $p = \infty$: distance de Chebyshev: $\max_i |x_i - y_i|$



- Distance euclidienne entre un point x et un point y:

$$D^2(x,y) = \sum_{i=1}^{I}(x_i - y_i)^2 = ||x||^2 + ||y||^2 - 2\underline{x} \cdot \underline{y} \quad (1)$$



- Produit scalaire: $\underline{x} \cdot \underline{y} = ||x|| \cdot ||y|| cos(\theta)$
- Distance cosinusoidale:
  $D(x,y) = 1 - cos(\theta) = 1 - \frac{\sum_i x_i y_i}{||x|| \cdot ||y||}$
  - 0 quand les vecteurs sont co-linaires ($\theta = 0$)
  - 1 quand les vecteurs sont orthogonaux ($\theta = \pi$)

  Intéréts:
  - distance normalisée $D(x,y) \in [0,2]$, insensible à la norme (cnfr chroma)

Recherche par similarité

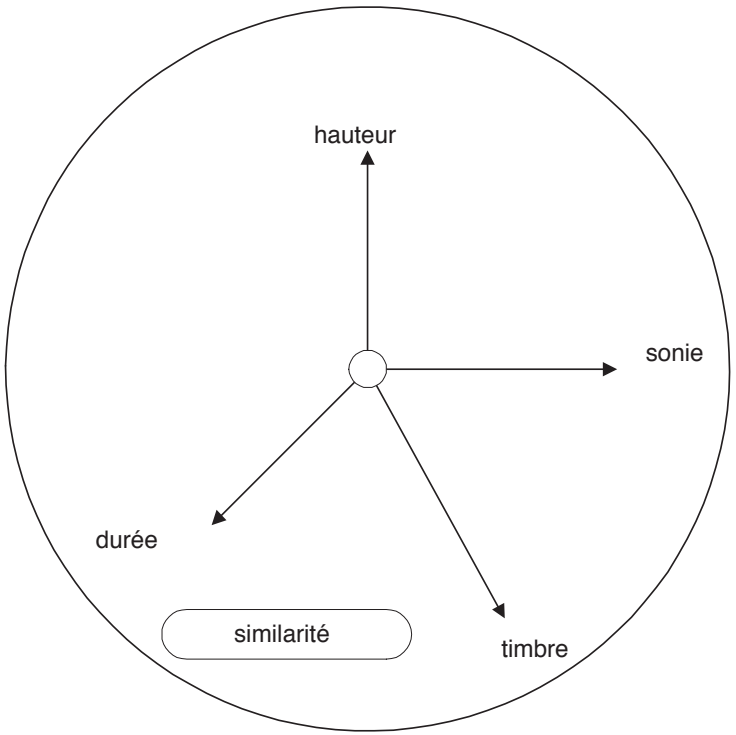
Qu'est-ce que la similarité ?

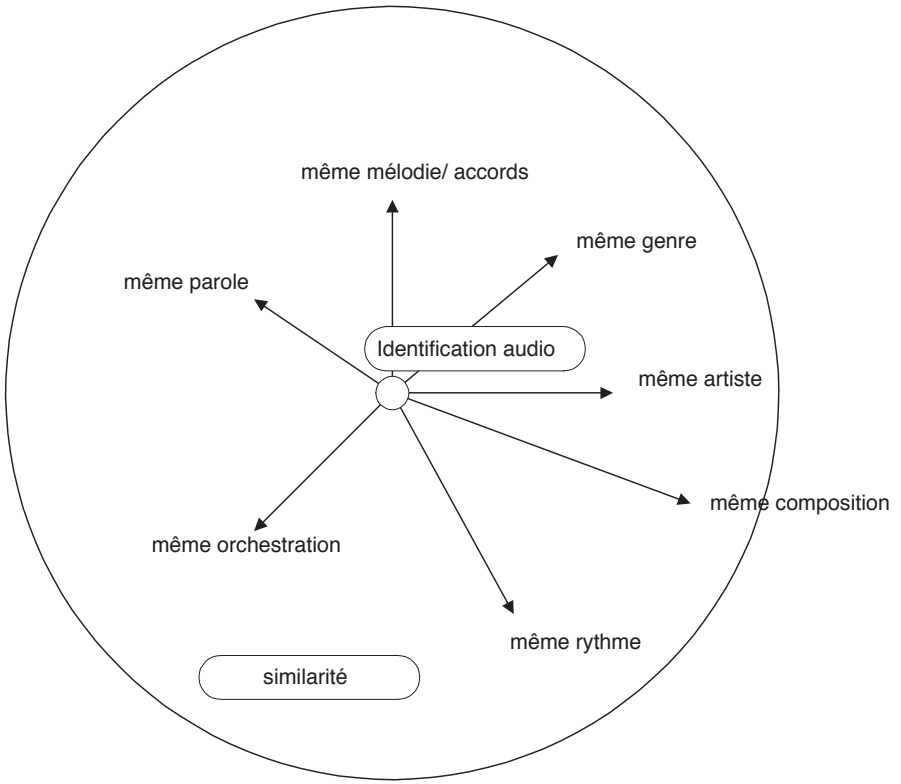
- Nombreux choix possibles
  - Profils utilisateurs (Amazon, iTune Genius): si un utilisateur aime A et aime B, si un autre utilisateur aime A on lui propose B
  - Similarité de classe: même source du son (trompette, violon), même artiste, même genre musical
  - Similarité basée sur le contenu: même valeurs de descripteurs audio
- Similarité basée sur le contenu
  - de timbre (instrument de musique, morceaux)
  - tonalité
  - rythme

Comment ça marche ?

- Trois exemples d'algorithmes de recherche par similarité
  - 0. Le niveau zéro de la similarité: l'identité (identification audio)
  - A. Similarité des descripteurs audio dans un espace euclidien
  - B. Similarité de l'évolution temporelle des descripteurs audio
  - C. Similarité entre deux modèles statistiques d'un contenu audio

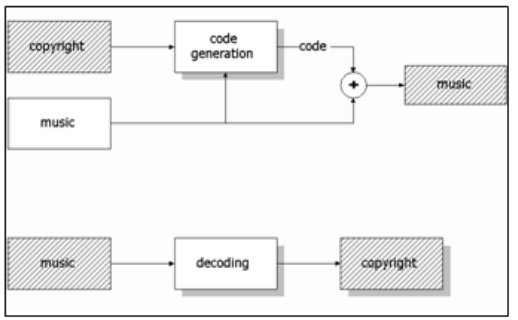
Recherche par similarité

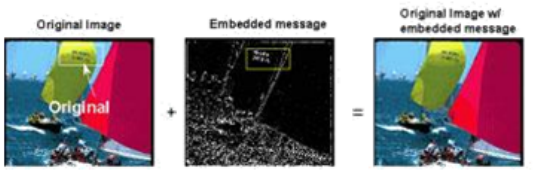


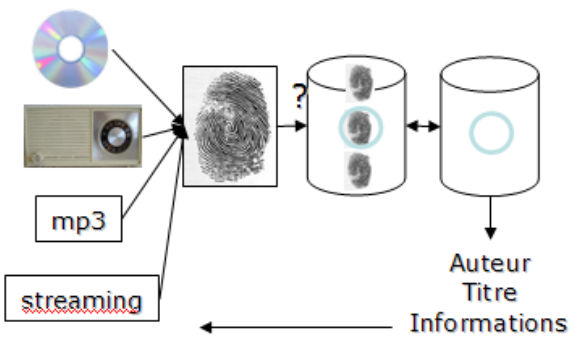


Identification audio

- Objectif: reconnaissance d'oeuvres diffusées sur radio, télé, Internet, bar, discothèque, ...
- Identifier l'enregistrement de manière exacte (insensible au dégradation du signal), mais ne permet pas d'identifier un autre enregistrement du même morceau
- **Watermarking**
  - Codage: introduction d'un code identifiant robuste mais inaudible dans le signal sonore
  - Décodage: pour un nouveau signal: extraction du code (si il est présent) et recherche de ce code dans une base de données
- **Fingerprint** (Shazam, Midomi, Philips, ...)
  - Déterminer un ensemble réduit de descripteurs audio extraits du signal sonore permettant d'identifier de manière unique un extrait musical
  - Codage: prise d'empreinte du signal, stockage dans une base de données
  - Décodage: pour un nouveau signal, prise d'empreinte, comparaison avec les empreintes de la base de données







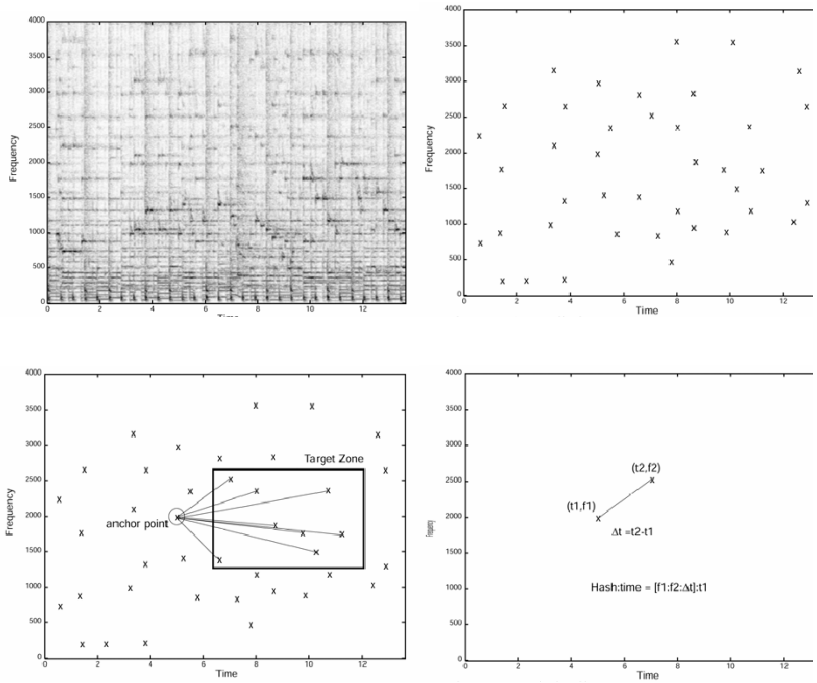
Systéme Shazam

Descripteurs audio Shazam

- Extraction de points saillants dans le plan temps/fréquence (onsets de sinusoides, maxima locaux): "constellation points"
- Représentation des "constellation points":
  - chaque point est pris comme un "anchor point" ayant une "target zone"
  - stoquage du Δ time et des fréquences des points par rapport à l'anchor

Matching:

- pour chaque points correspondants dans la base de données on mesure le décalage temporel: $time_{db} - time_{sample} = constant$
- puisque le décalage doit être constant, on calcul l'histogramme des Δ t, il doit présenter un peak

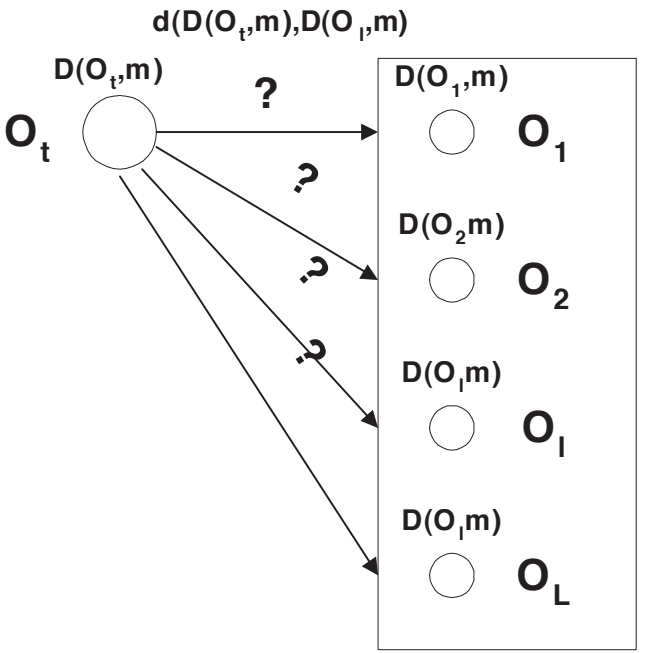


d'aprés Shazam Entertainment 2003

A. Similarité des descripteurs audio dans un espace eucidéen

On donne un son en exemple, O_t, (la cible/ "target") et on cherche les sons (d'une base de données) les plus "similaires"

- Calcul de la distance entre O_t et tous les sons de la base de données O_l $l \in L$
- pour calculer la distance, on compare les valeurs des descripteurs audio de O_t, $\underline{D}(O_t)$ à ceux des sons O_l, $\underline{D}(O_l)$ $\forall l \in L$.
- chaque son O est représenté par les valeurs de ses descripteurs $D_i(O)$ dans un espace multi-dimensionnel
- distance euclidienne entre $\underline{D}(O_t)$ et $\underline{D}(O_l)$ pour obtenir la dissimilarité entre le son O_t et les sons O_l
- on trie l'ensemble des distances $d(O_t, O_l)$
- les distances les plus petites sont les sons les plus proches



A. Similarité des descripteurs audio dans un espace eucidéen

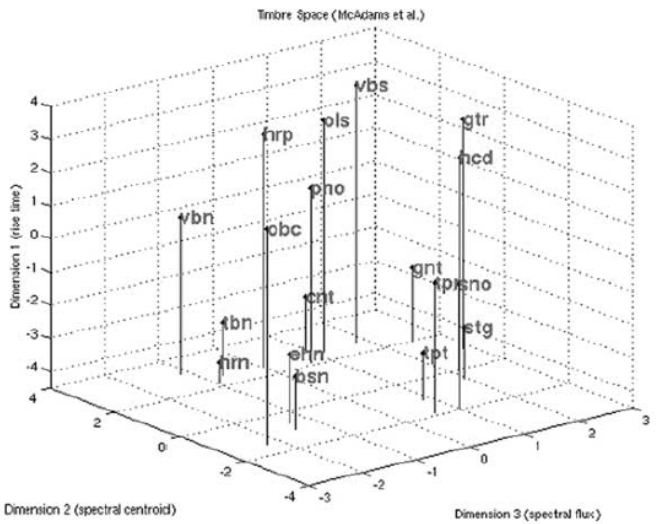
Sous-problèmes

- 1. Quels descripteurs choisir pour décrire la similarité ? Les MFCCs ? Le centroid spectral ? Le zcr ?
- 2. Quelles pondérations appliquer aux descripteurs ?
  - Est-ce que certains descripteurs sont plus importants que d'autres perceptivement ?
  - De plus, les différents descripteurs ont des plages de variations très différentes (exemples: centroid spectral en Hz $\in [0, 10000]$, noisiness $\in [0, 1]$), ceci va influencer la distance euclidienne
  - Solution: On pondère les descripteurs par α_i: $d(O_t, O_l) = \sqrt{\sum_i \alpha_i (d_i(O_t) - d_i(O_l)^2}$
  - Comment trouver les α_i ?
    - Expériences
    - Normalisation min-max: $\alpha_i = 1/(\max_l d_i(O_l) - \min_l d_i(O_l))$
    - Normalisation par l'écart type: $\alpha_i = 1/\sigma(d_i)$

A. Les espaces de timbre

Les espaces de timbre ?

- Études perceptives menées depuis les années 70 (Plomp, Wessel, Grey, Krumhansl, McAdams, ...) pour comprendre comment l'être humain perçoit le timbre des instruments de musique
- Le timbre ? Attributs qui permettent de distinguer deux sons de même hauteur, sonie, durée
- Les études ?
  - Ensemble de sons de même hauteur, sonie et durée et de source non-reconnaissable
  - On demande à un ensemble de personnes de juger la similarité/ dissimilarité entre chaque paire de sons (O_l, O'_l)
  - Analyse "Multi-Dimensional Scaling" permet de représenter le jugement de similarité/dissimilarité moyen des personnes par la distance entre des points dans un espace de faible dimension (2dim, 3dim)

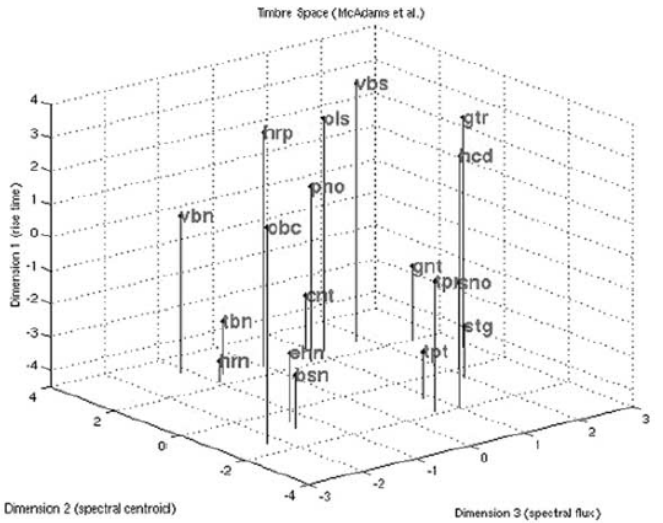


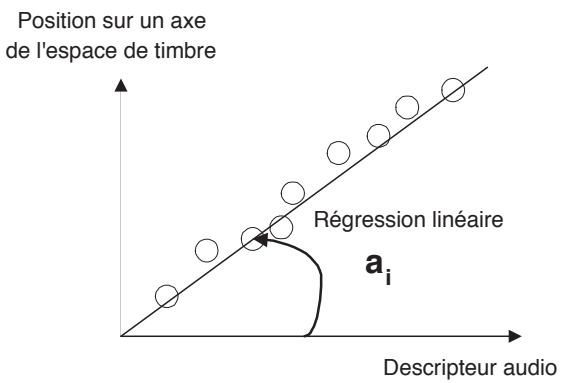
SONS

A. Les espaces de timbre

Utilisation des espaces de timbre pour la recherche par similarité

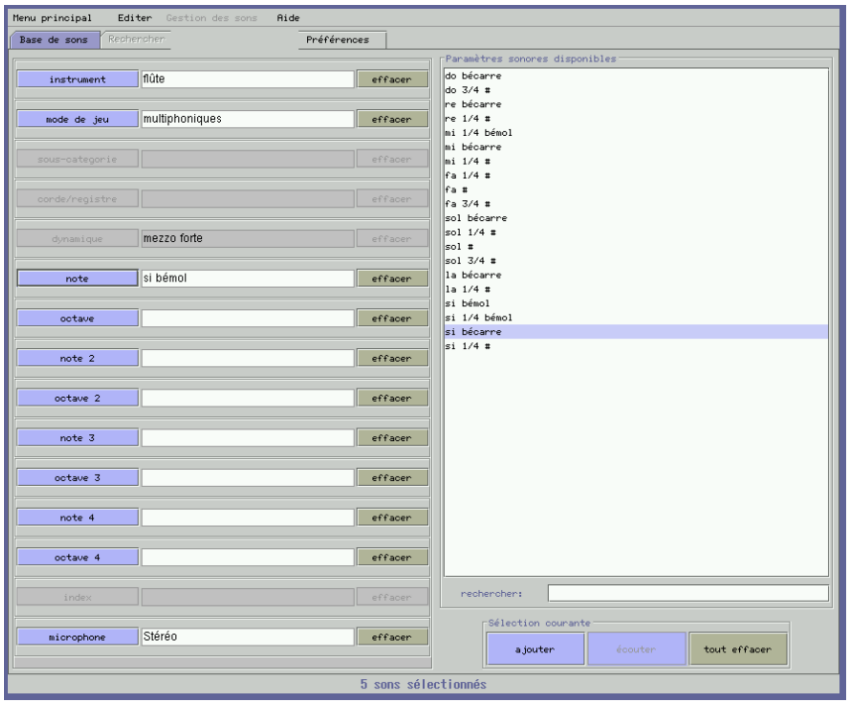
- les sons O_l sont représentés par des points
- pour chaque son O_l, on calcul les descripteurs D_i
- on cherche les descripteurs qui permettent le mieux d'expliquer la position des sons sur chaque dimension de l'espace
  - Mesure: corrélation entre chaque descripteurs D_i et la position des sons sur une dimension.
  - Coefficient de corrélation de Pearson: $c(x,y) = \frac{(x-\mu_x)(y-\mu_y)}{||x-\mu_x||||y-\mu_y||}$
  - Pour chaque dimension, on garde le descripteurs D_i le plus corrélé
- On cherche (par régression linéaire) la pondération α_i à appliquer à D_i afin d'expliquer au mieux les distance $d(O_l, O'_l)$ dans l'espace de timbre
  - $d^2(l,l') = \alpha_1 \Delta LAT^2 + \alpha_2 \Delta hsc^2 + \alpha_3 \Delta hsd^2 + \alpha_4 \Delta hss^2 + \alpha_5 \Delta hsv^2$
- Reproduit la distance perçue en moyenne



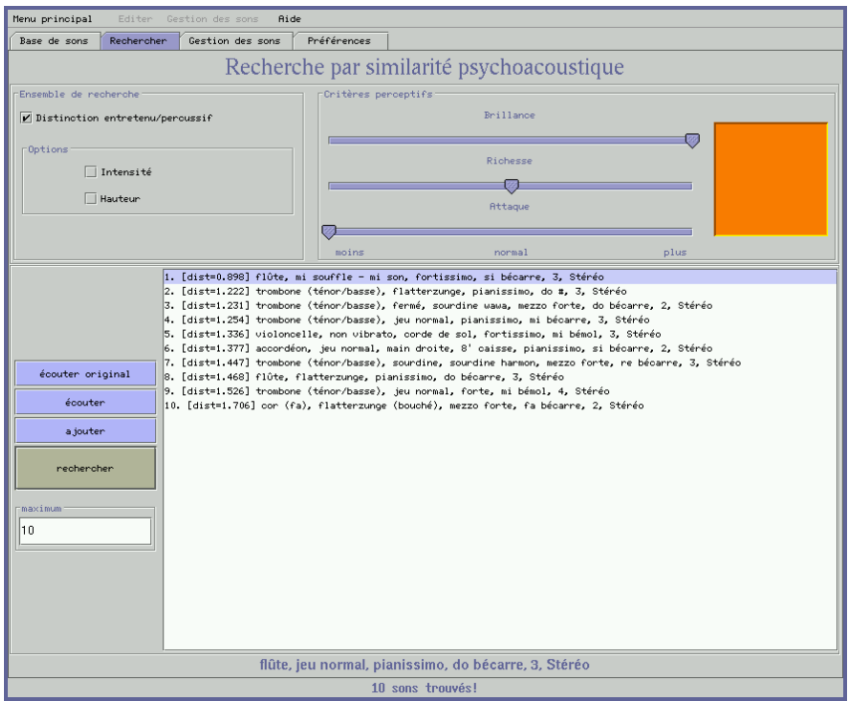


A. Exemple

Studio On Line: une base de données de sons en ligne de l'Ircam, 130.000 sons, moteur de recherche par similarité acoustique

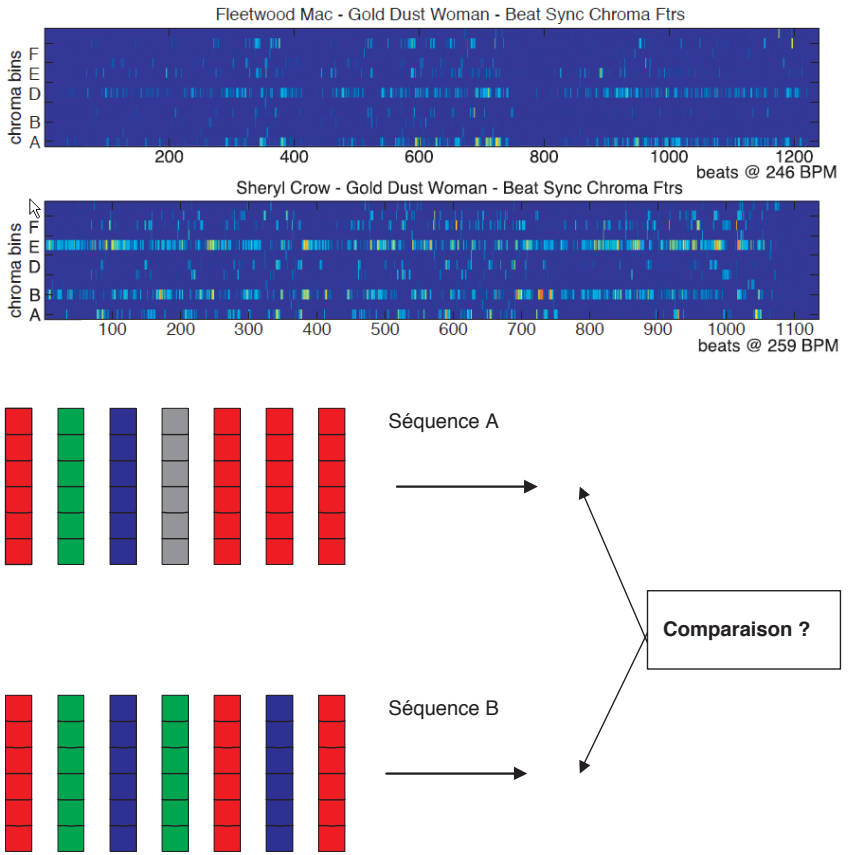


VIDEO



B. Similarité de l'évolution temporelle des descripteurs audio: Détection de "cover-version"

- Objectif: Détecter les multiples cover-versions O_l (reprises) d'un morceau cible O_t dans une base de donnée
- Caractéristiques d'une cover-version: généralement la même suite harmonique (même suite d'accords, même mélodie), (également même paroles)
- Extraction de la séquence temporelle des chromas/PCP de chaque morceau: $\underline{D}(O_l, m)$, m est le temps
- On va calculer le coût d'alignement entre $\underline{D}(O_t, m)$ et tous les $\underline{D}(O_l, m)$ des morceaux de la base
- Technique utilisée: l'Alignement Dynamique du Temps

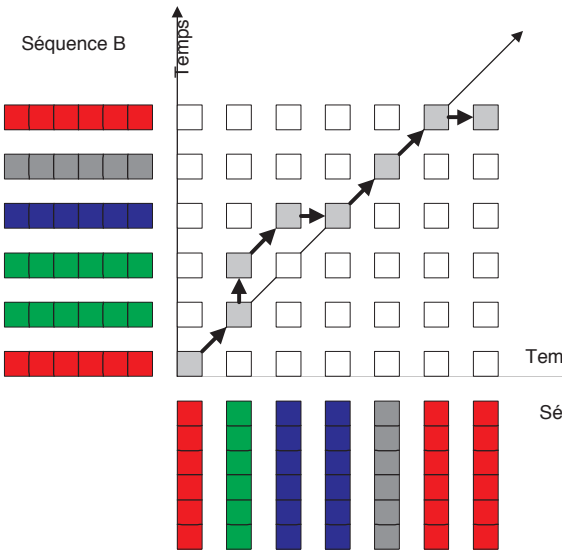


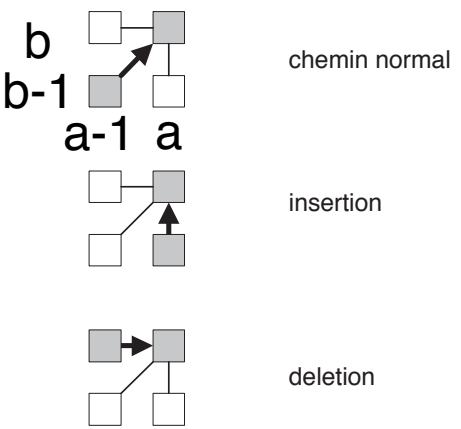
B. Alignement Dynamique du Temps

- Objectif: aligner deux séquences temporelles A et B et calculer le coût de leur alignement
- Hypothèse: Les points de début et de fin des deux séquences sont supposées en correspondances
- On parcours progressivement tous les points (a,b) de la matrice d'alignement
- En un point (a,b) on cherche le meilleur chemin (de coût local minimal) pour y arriver parmi les trois chemins:
  - $(a-1, b-1) \rightarrow (a, b)$: chemin normal
  - $(a, b-1) \rightarrow (a, b)$: insertion
  - $(a-1, b) \rightarrow (a, b)$: deletion

  Le coût local est calculer comme
  - Le coût du point précédent (cumul)
  - Plus un coût défavorable si on utilise les chemins insertion et deletion

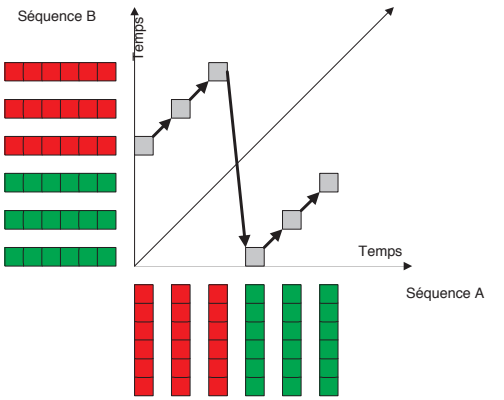
  On associe à (a,b) ce coût local plus son coût intrinsèque: la distance entre $\underline{D}(O_t, a)$ et $\underline{D}(O_l, b)$
- Arrivé à la fin, on obtient le coût de l'alignement globale

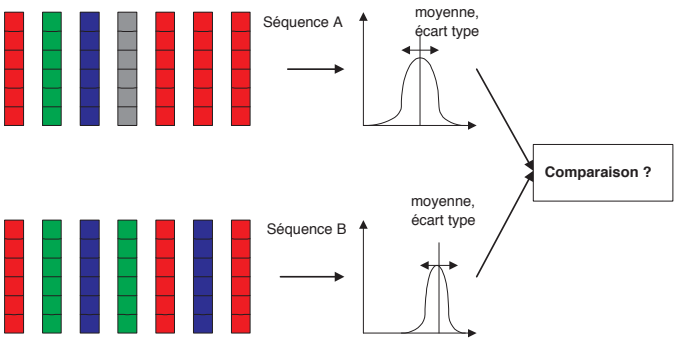




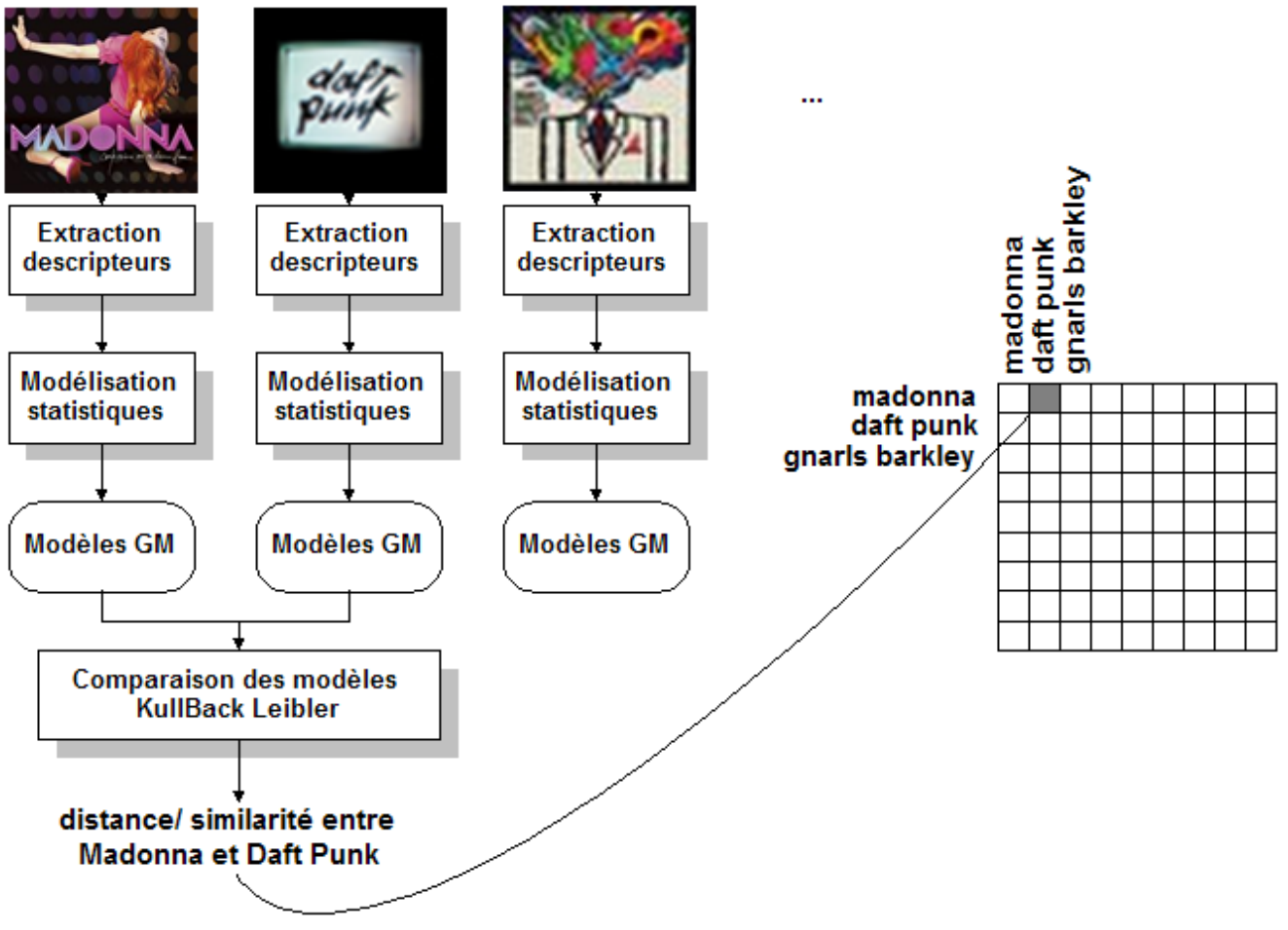
C. Similarité entre deux modèles statistiques d'un contenu audio

- Problème:
  - Morceau de musique= séquence de descripteurs audio $\underline{D}(m)$
  - Comparaison de séquences $\underline{D}(O_t, m)$ et $\underline{D}(O_l, m)$: pas nécessairement corrélée à la similarité (cnfr le timbre)
- Comment faire ?
  - Modélisation statistique de la séquence de descripteurs audio de O_t et O_l
  - Comparaison des modèles statistiques N_t et N_l
  - = Le modèle le plus utilisé en similarité musicale
- En pratique
  - $\underline{D}(O_t, m)$: les MFCCs (vecteur à 13dim à chaque trame m).
  - On modélise le paquet (et non la séquence) de $\underline{D}(O_t, m)$ par un modèle statistique: modèle gaussien $N(\underline{\mu}_t, \Sigma_t)$
  - Distance ? "divergence de Kullback-Leibler symétrisée" entre $N(\underline{\mu}_t, \Sigma_t)$ et $N(\underline{\mu}_l, \Sigma_l)$





C. Similarité entre deux modèles statistiques d'un contenu audio



VIDEO

C. La loi normale (gaussienne)

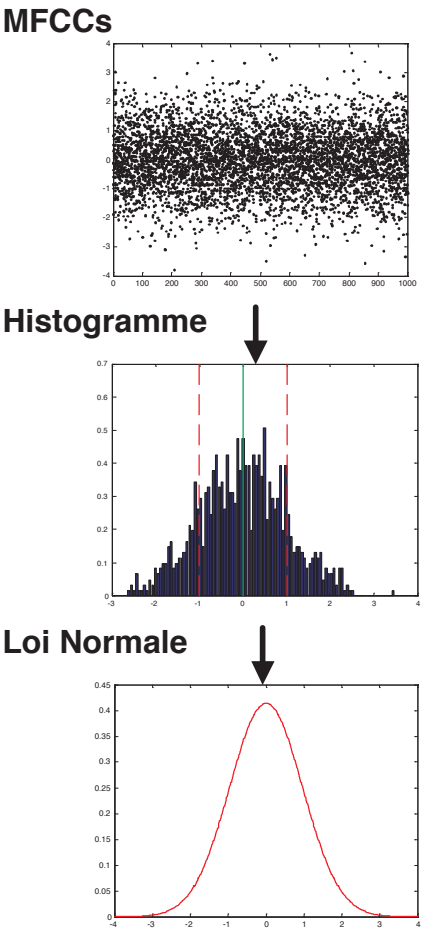
- On analyse la distribution (histogramme) de tous les MFCCs extrait au cours du temps (sans prendre en compte le temps)
- La distribution de chaque dimension (13 dim), suit une loi normale (gaussienne)
- On modélise le paquet de MFCCs par une loi normale à 13 dimensions
- Loi normale
  - Densité normale de probabilité à 1 dimension

$$p(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-1/2\left(\frac{x-\mu}{\sigma}\right)^2} \quad (2)$$

  - Densité normale de probabilité à d dimensions

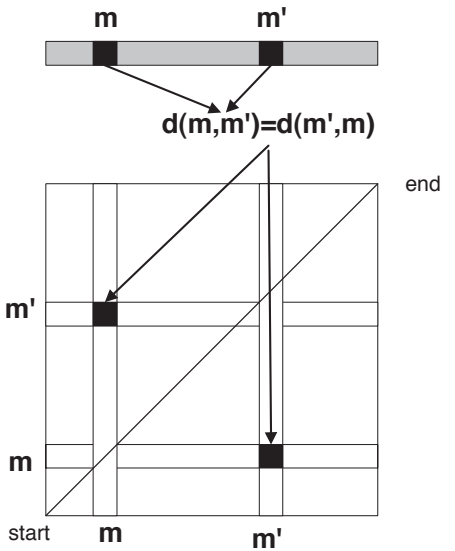
$$p(x) = \frac{1}{(2\pi)^{d/2}|\Sigma|^{1/2}} e^{-1/2(x-\mu)^t \Sigma^{-1}(x-\mu)} \quad (3)$$

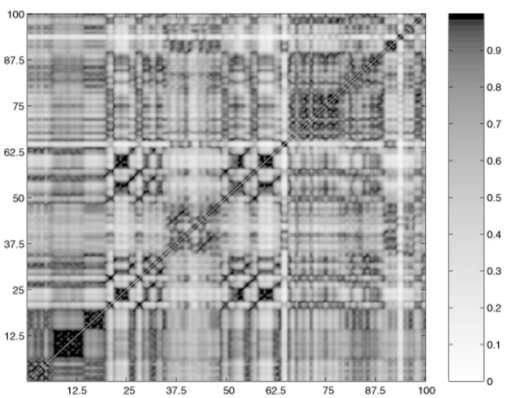
    - $\mu = E[x] = \int_x x p(x) dx$
    - $\Sigma = E[(x-\mu)(x-\mu)^t] = \int (x-\mu)(x-\mu)^t p(x) dx$
  - Différents types de matrice de covariance Σ: sphérique, diagonale, pleine



La similarité des événements é l'intérieur d'un morceau

- On cherche à analyser la structure d'un morceau de musique en terme de répétitions d'événements au cours du temps (couplets, refrains)
- On applique le calcul de mesure de similarité (distance euclidienne, cosinusoidale) entre les différents instants d'un même morceau
- Méthode
  - Soit $\underline{D}(m)$ un vecteur de descripteurs audio extraits à l'instant m
  - En pratique, on choisit $\underline{D}(m)$ comme une combinaison de MFCC, Chromas/PCP et SFM
  - On calcul les distances $d(m, m')$ entre tous les couples de temps (m, m') d'un même morceau
  - Représentation sous forme d'une matrice de similarité/ distance ou encore co-occurrence
- Utilisation: localisation des couplets, refrain

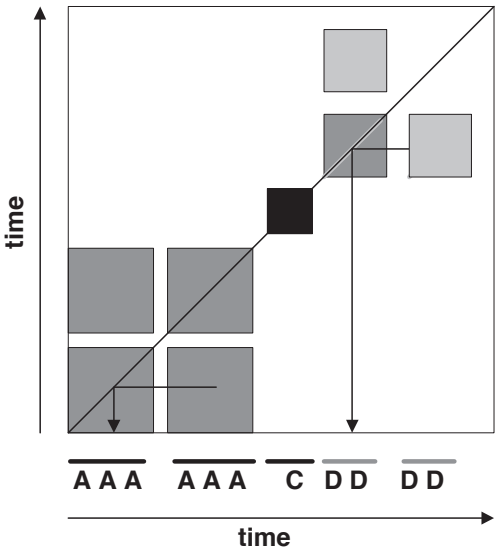




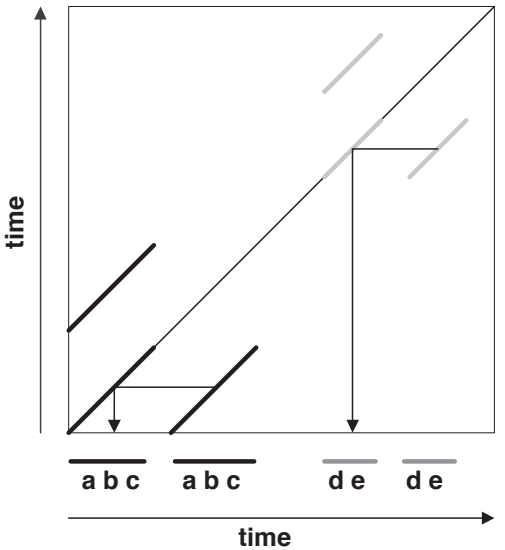
DEMO

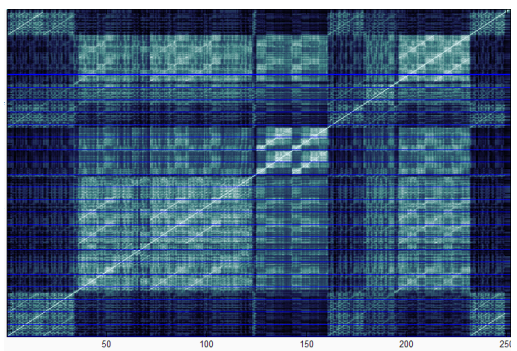
Lecture d'une matrice de similarité

Approche par états

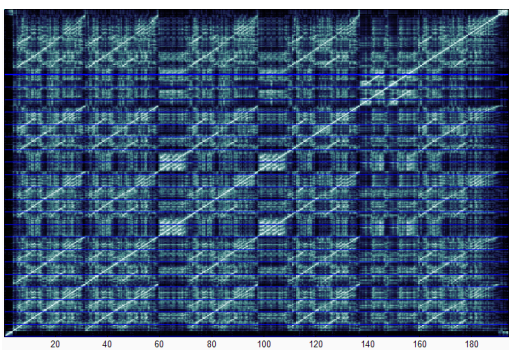


Approche par séquences



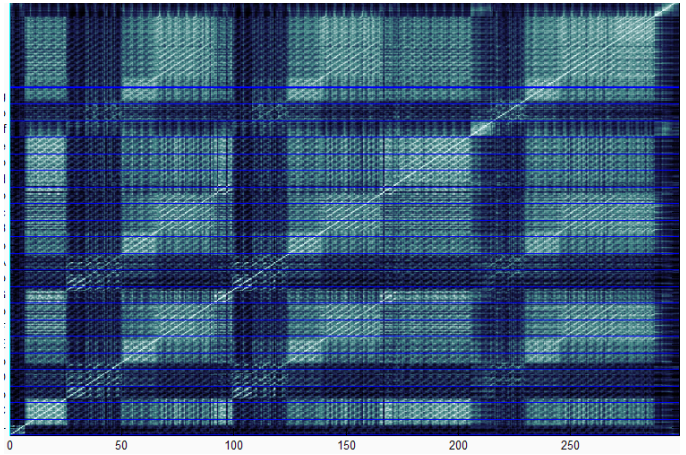


Moby "Natural Blues"

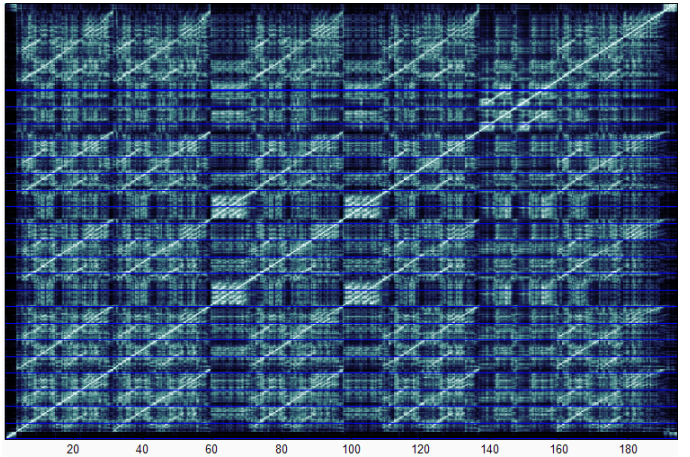


Chopin "Mazurka Op.6 No1"

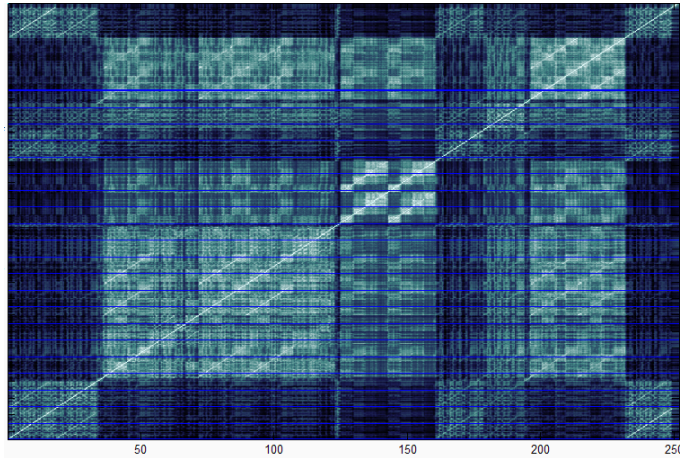
Exemples



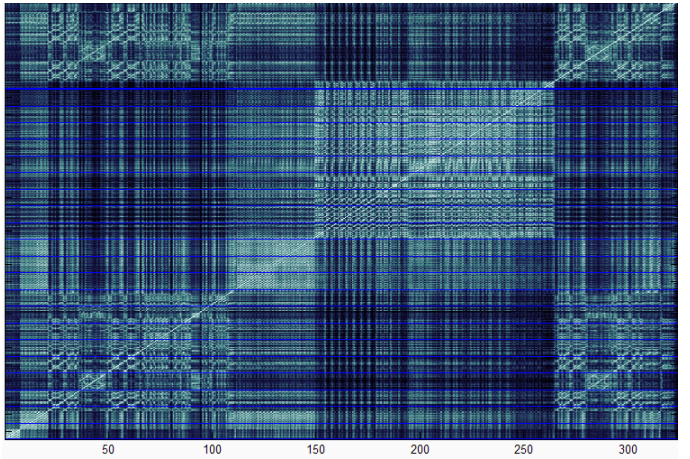
Nirvana "Smells like teen spirit"



Chopin "Mazurka Op.6 No1"

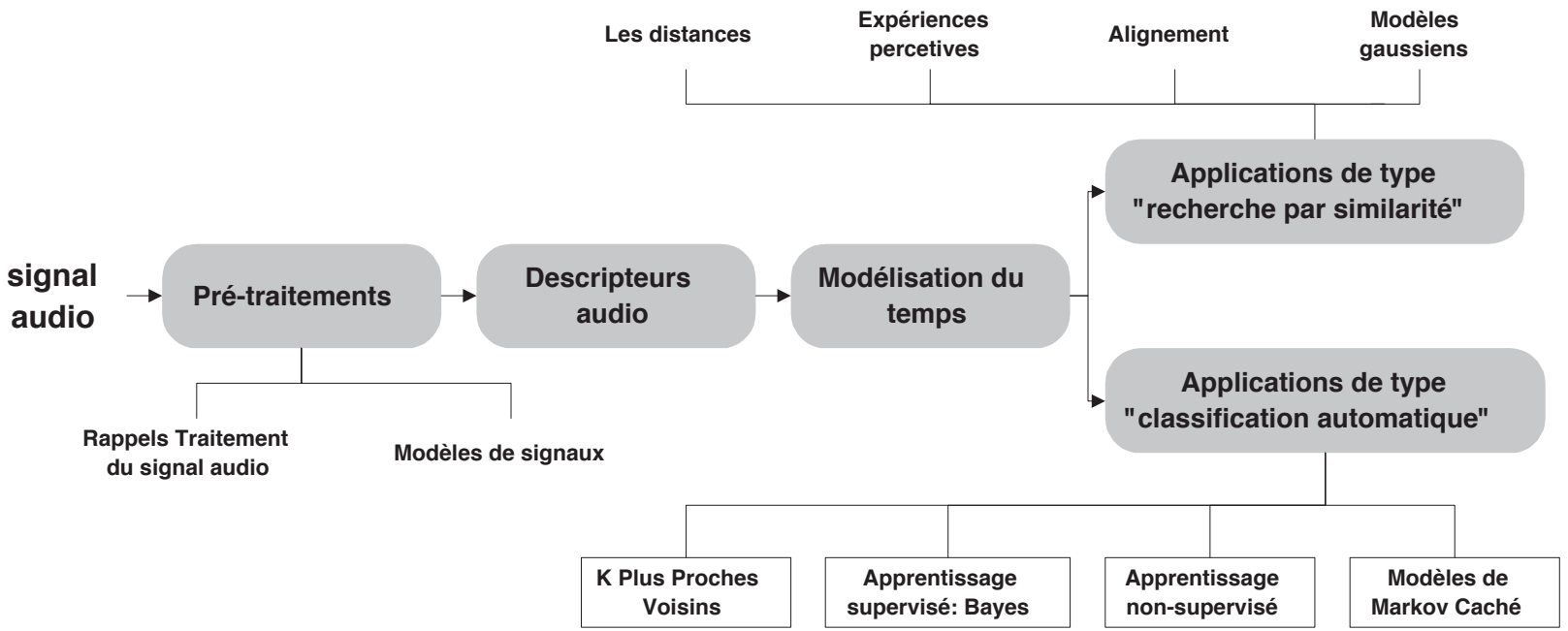


Moby "Natural Blues"



Dave Brubeck "Take Five"

Plans


Les distances
Expériences percetives
Alignement
Modèles gaussiens
Applications de type "recherche par similarité"
signal audio
Pré-traitements
Descripteurs audio
Modélisation du temps
Applications de type "classification automatique"
Rappels Traitement du signal audio
Modèles de signaux
K Plus Proches Voisins
Apprentissage supervisé: Bayes
Apprentissage non-supervisé
Modèles de Markov Caché

Indexation automatique: deux types d'approches

Approche apprentissage automatique

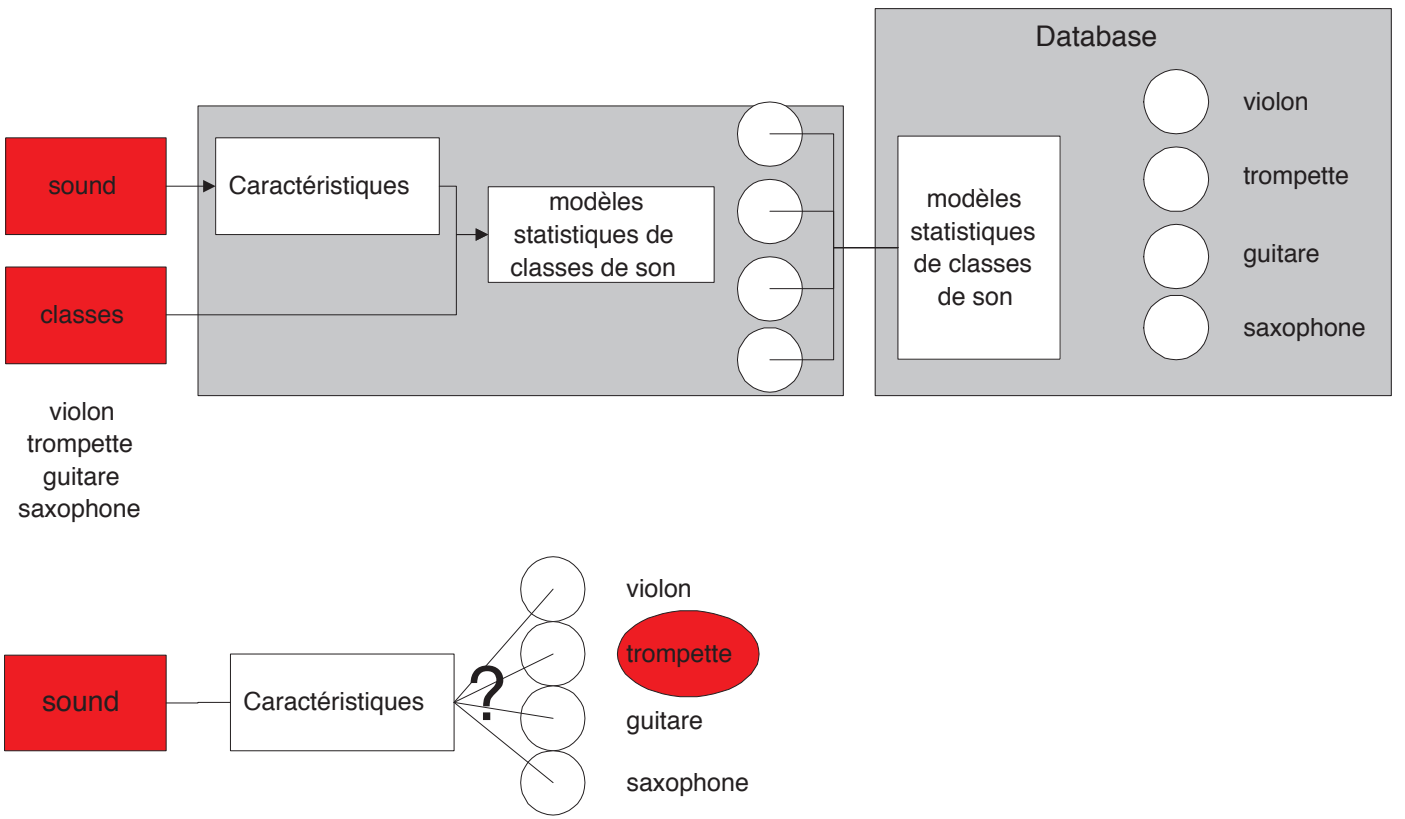
- Le probleme est
  - modélisé par des classes (classes parole/musique, classes de music-genre, classes de music-mood)
- Apprentissage:
  - Extraire du signal audio un nombre important d'information = descripteurs de contenu audio
  - Sélectionner automatiquement les descripteurs les plus pertinents pour un problème donné
  - Modéliser les propriétés statistiques des descripteurs audio pour chaque classe
- Indexation d'un signal inconnu:
  - Extraction de descripteurs
  - Évaluation de l'appartenance de ses descripteurs à un modèle statistique donné

Approche extraction supervisée

- Le problème est
  - trop complexe pour être résolu automatiquement
  - mais le concept à extraire est formalisable (exemples: hauteur, tempo)
- Apprentissage
  - Construction d'un algorithme spécifique pour résoudre un problème spécifique
  - Recherches en traitement du signal + reconnaissance de forme et mapping manuel
- Indexation d'un signal inconnu:
  - Faire tourner un algorithme spécifique pour extraire chaque paramètre inconnu du signal spécifique
- Avantage: performance importante
- Inconvénient: pas de généralisation

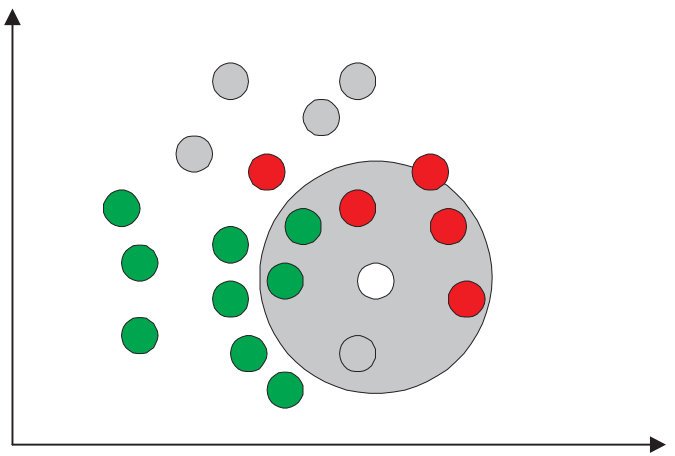
Système type de classification

- Phase d'apprentissage (training)
  - Choisir un ensemble d'observations (sons, morceaux de musique) représentatifs des classes à apprendre: annotation
  - Si possible classes équilibrées (même nombre d'exemples par classes), exemples non-ambigus
- Phase de classification (test, évaluation)

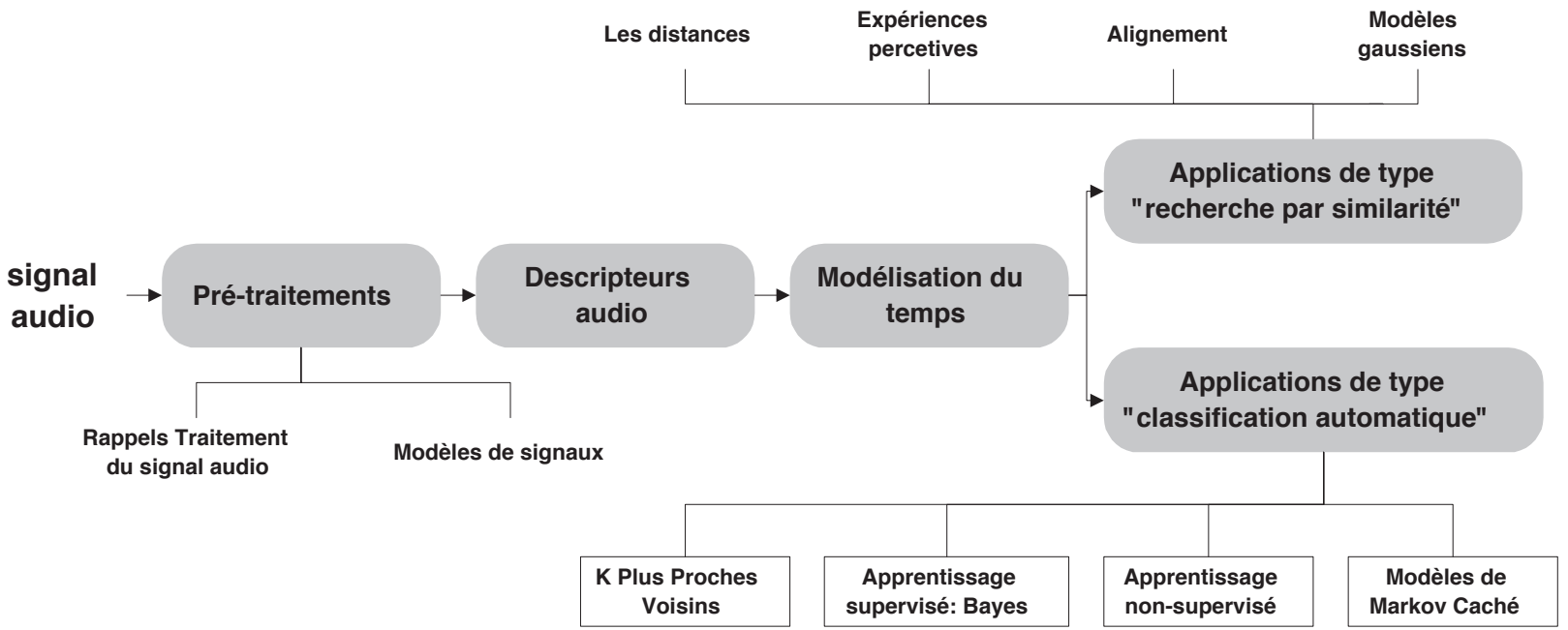


Les K Plus Proches Voisins

- Classificateur sans modèle; principe très simple
- Entraînement:
  - remplir un espace par l'ensemble des "points" d'apprentissage (les descripteurs audio à I dimensions)
  - à chaque point x est associé sa classe $\omega(x)$
- Évaluation
  - Soit y de classe $\omega(y)$ inconnue
  - Chercher dans l'espace de descriptions les K points les plus proches de y selon une distance euclidienne: x_k
  - On associe à y la classe majoritaire parmi les K plus proche voisins x_k $\omega(y) = argmax_{\omega \in \Omega} \sum_{k=1}^{K} \delta(\omega, \omega(x_k))$
  - Suppose:
    1. le nombre K optimal donné
    2. la distance euclidienne utilisable pour le type de données
    3. les descripteurs audio (axes) ont des échelles comparables $\rightarrow$ normalisation
  - Désavantage:
    1. demande le stockage et l'accès à toutes les données (très grand nombre de données)
    2. coût de calcul de la distance entre y et tous les point x

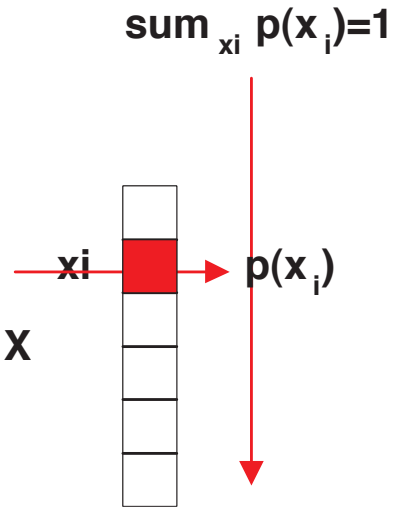


Plans


Les distances
Expériences percetives
Alignement
Modèles gaussiens
Applications de type "recherche par similarité"
signal audio
Pré-traitements
Descripteurs audio
Modélisation du temps
Applications de type "classification automatique"
Rappels Traitement du signal audio
Modèles de signaux
K Plus Proches Voisins
Apprentissage supervisé: Bayes
Apprentissage non-supervisé
Modèles de Markov Caché

Éléments de probabilité

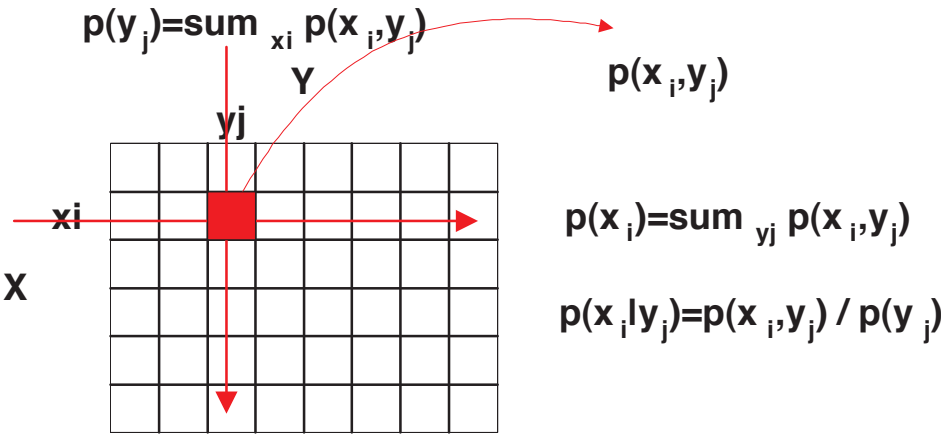
- Variable aléatoire X:
  - l'ensemble des résultats possibles d'une expérience aléatoire (gain d'un joueur dans un jeu de hasard)
- Loi de probabilité:
  - $P(X = x_i) = p(x_i)$
  - Exemples de loi discrètes: Bernoulli, binomiale, hypergéométrique, poisson
- Densité de probabilité $f_X(x)$:
  - $P(a < X < b) = \int_a^b f_X(x)dx$
  - Exemples de fonction $f_X(x)$: loi uniforme, loi normale, loi exponentielle
- Propriétés
  1. $P(x) \in [0, 1]$
  2. $\int_x P(x)dx = 1$



Éléments de probabilité (2 variables)

- A deux variables:
  - $P(X = x_i, Y = y_j) = p(x_i, y_j)$
- Si les variables sont indépendantes:
  - $p(x_i, y_j) = p(x_i)p(y_j)$
- Loi marginale de X de Y:
  - $p(x_i) = \sum_j p(x_i, y_j)$
  - $p(y_j) = \sum_i p(x_i, y_j)$
- Probabilité conditionnelle:
  - $p(x_i|y_j) = \frac{p(x_i,y_j)}{p(y_j)}$
  - $p(y_j|x_i) = \frac{p(x_i,y_j)}{p(x_i)}$

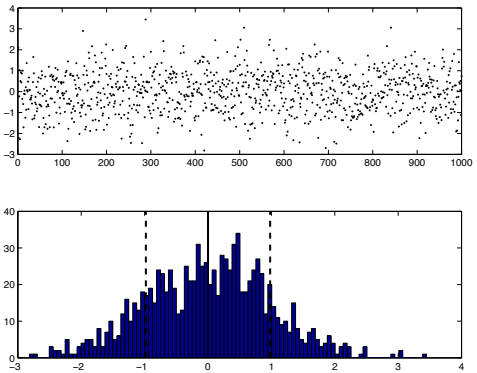
  Donc
  $p(x_i, y_j) = p(x_i|y_j)p(y_j) = p(y_j|x_i)p(x_i)$



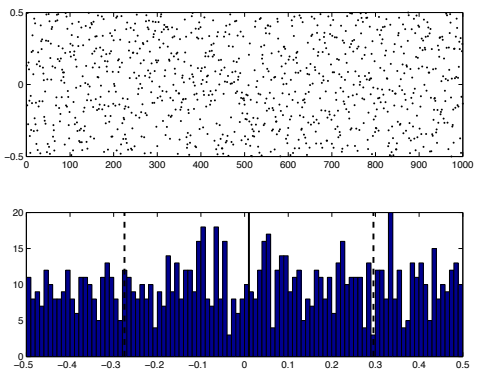
Éléments de probabilité

- Espérance: $E(X) = \sum_i p(x_i)x_i$
- La moyenne:
  - mesure la tendance centrale d'une variable X
  - $\mu_X = E(X) = \sum_i p(x_i)x_i$
- L'écart type:
  - mesure l'écart moyen de la variable X autour de sa moyenne
  - $\sigma_X^2 = E((X - \mu_X)^2) = \sum_i p(x_i)(x_i - \mu_X)^2$
  - écart-type $\sigma_X = \sqrt{\sigma_X^2}$
- L'histogramme
  - Soit un ensemble donné d'observations d'une variable X: x_o
  - On mesure combien de fois x_o est dans un intervalle donné
  - Si X prend ses valeurs dans l'intervalle $[a, b]$, un histogramme sur N points revient à calculer
  - $h(n) = \sum_o x_o \in [a + (n-1)\frac{b-a}{N}, a + n\frac{b-a}{N}]$

Exemple 1:



Exemple 2:

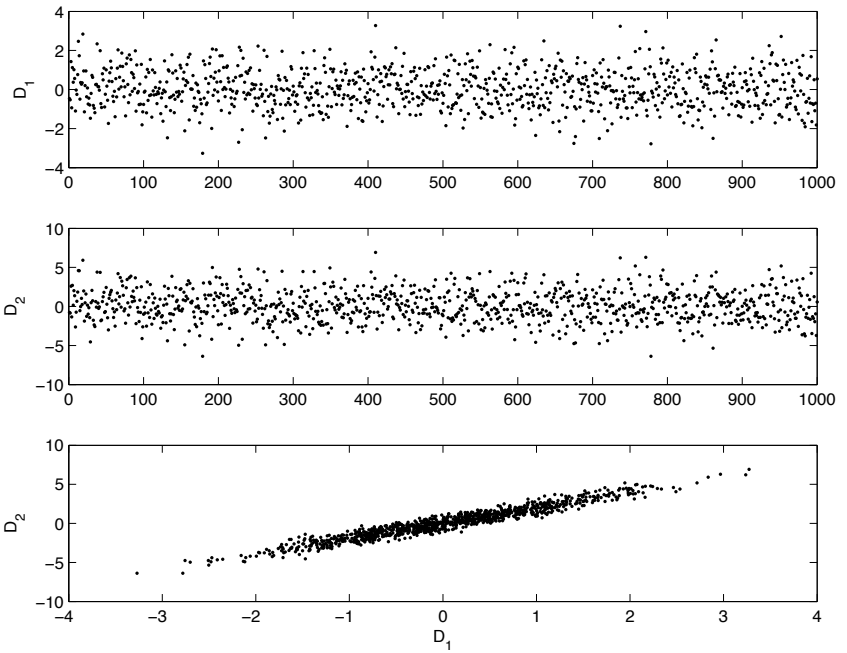


Éléments de probabilité (2 variables)

Vecteur aléatoire $\underline{Z} = [X, Y]$

- Vecteur de moyenne $\underline{\mu_Z} = [\mu_X, \mu_Y]$
  - $\mu_X = \sum_i p(x_i) x_i$
  - $\mu_Y = \sum_j p(y_j) y_j$
- La matrice de variance/ co-variance Σ
  - Variance $\sigma_X^2 = \sum_i p(x_i)(x_i - \mu_X)^2$
  - Variance $\sigma_Y^2 = \sum_j p(y_j)(y_j - \mu_Y)^2$
  - Covariance (x, y): mesure la variation simultanée de deux variables aléatoires $cov(x, y) = \sigma_{xy} = E((X - \mu_X)(Y - \mu_Y)) = \sum_i \sum_j p(x_i, y_j)(x_i - \mu_x)(y_j - \mu_y)$
    - variables non-corrélée: $cov(x, y) = 0$
    - variables corrélées: $cov(x, y) > 0$
  - Matrice de variance/ co-variance:

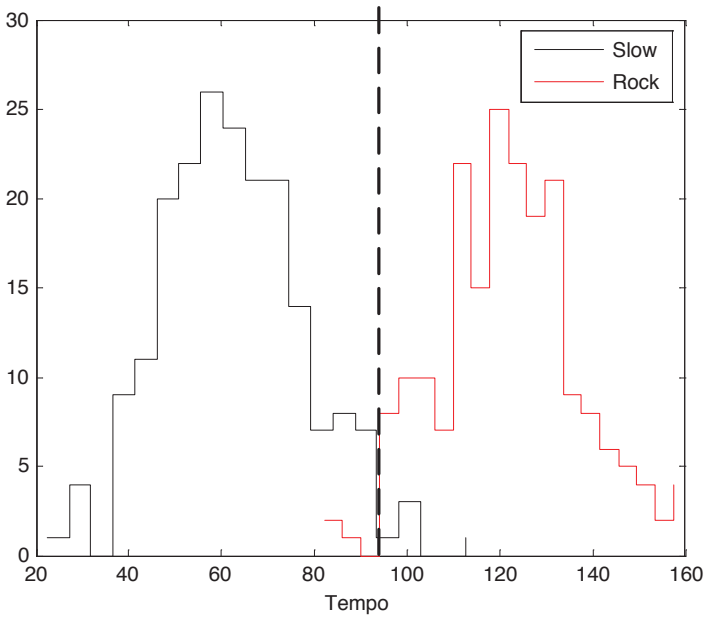
$$\Sigma = \begin{pmatrix} \sigma_x^2 & \sigma_{x,y} \\ \sigma_{y,x} & \sigma_y \end{pmatrix} \qquad (4)$$



La discrimination

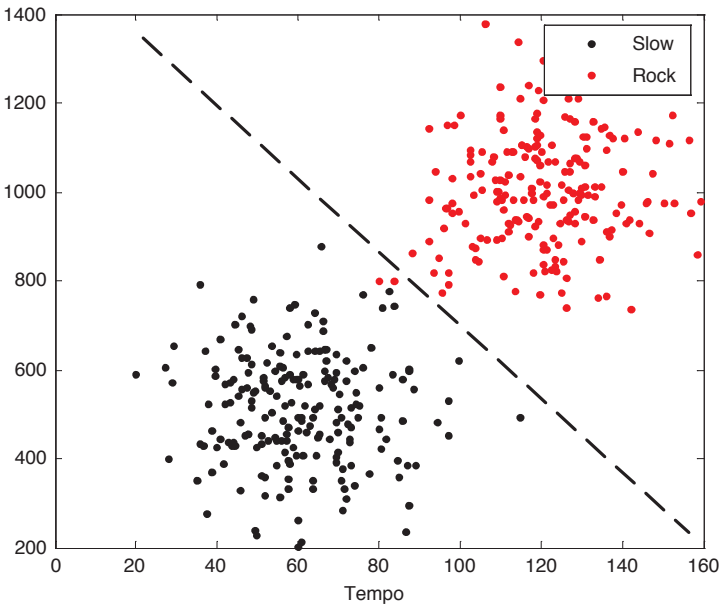
Exemple simple

- On cherche à créer un classificateur qui détecte deux classes de musique: "slow" et "rock"
- On observe un seul descripteur: la valeur de D = tempo pour un ensemble de morceaux (base d'apprentissage) de musique "slow" et "rock"
- On cherche la meilleure valeur de D (appelé seuil de décision $d*$) permettant de séparer la classe "slow" de la classe "rock"
- $d*$ (ligne en pointillée)
  - permet de minimiser le nombre d'erreur de classification en moyenne
  - il reste cependant des erreurs



La discrimination

- On observe maintenant deux descripteurs: la valeur de D_1 = tempo et de D_2 = centroid spectral
- On cherche la meilleure séparation (ligne de décision en pointillé) permettant de séparer la classe "slow" de la classe "rock"
- La ligne en pointillé fournie le meilleur compromis entre
  - minimisation du nombre d'erreurs de classification
  - simplicité du modèle (surface de décision linéaire), généralisation du modèle en dehors de l'ensemble d'apprentissage
  - il faut également limiter le nombre de paramètres d'observations (malédiction des grandes dimensions)
- Compromis entre performance sur la base d'apprentissage et généralisation (simplicité du modèle)



Règle de Bayes

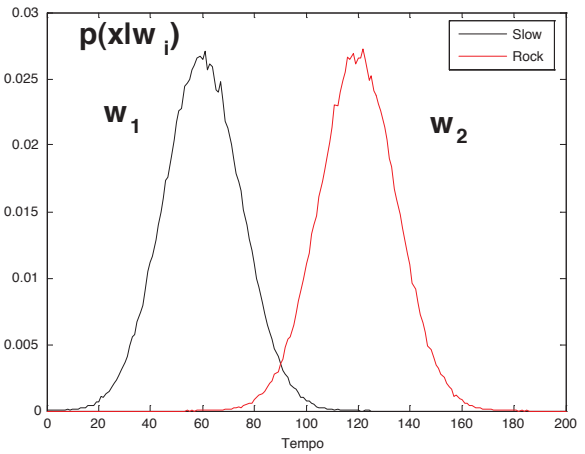
- Soit $\omega(O)$ l'état ou la classe d'une observation O
  - $\omega =?$ rock, $\omega =?$ slow
- $\omega(O)$ est une variable que l'on cherche à prédire
- Probabilité a priori: la connaissance qu'on a "a priori" d'observer un état dans un contexte donné
  - $P(\omega = rock) = P(\omega = slow) = 0.5$ si le morceau passe sur FIP
  - $P(\omega = rock) > P(\omega = slow)$ si le morceau passe passe sur "Sky-Rock"
- Si on ne dispose pas d'autres informations
  - on a "a priori" plus de chance d'observer la classe "rock" que la classe "slow" sur "Sky-Rock"
- Si on dispose d'autres informations (descripteurs audio)
  - on utilise les probabilités conditionnelles
- Probabilités conditionnelles ?
  - la probabilité d'observer un état ω étant donné qu'on a telle observation X: $P(\omega|X)$
  - la probabilité d'observer la classe "rock" étant donné les MFCCs $P(\omega = rock|X = MFCC)$

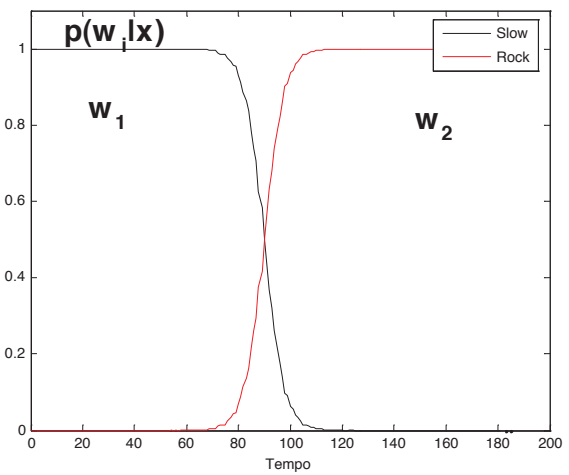
Règle de Bayes

- Pendant l'apprentissage on "apprend" les valeurs typiques des observations X pour chaque état ω: $P(X|\omega)$
  - on "apprend" les valeurs typiques des MFCCs pour tous les morceaux de la classe "rock"/"slow":
    $P(X = MFCC|\omega = rock)$,
    $P(X = MFCC|\omega = slow)$
- Règles de Bayes: permet de passer de $P(X|\omega)$ à $P(\omega|X)$

$$P(\omega|X) = \frac{P(X|\omega)P(\omega)}{P(X)} = \frac{vraisemblance * prior}{evidence} \quad (5)$$

- Prise de décision: on décide que la classe est
  - ω_1 si $P(\omega_1|X) > P(\omega_2|X)$
  - ω_2 si $P(\omega_2|X) > P(\omega_1|X)$





$P(X|\omega)$? La loi normale (gaussienne)

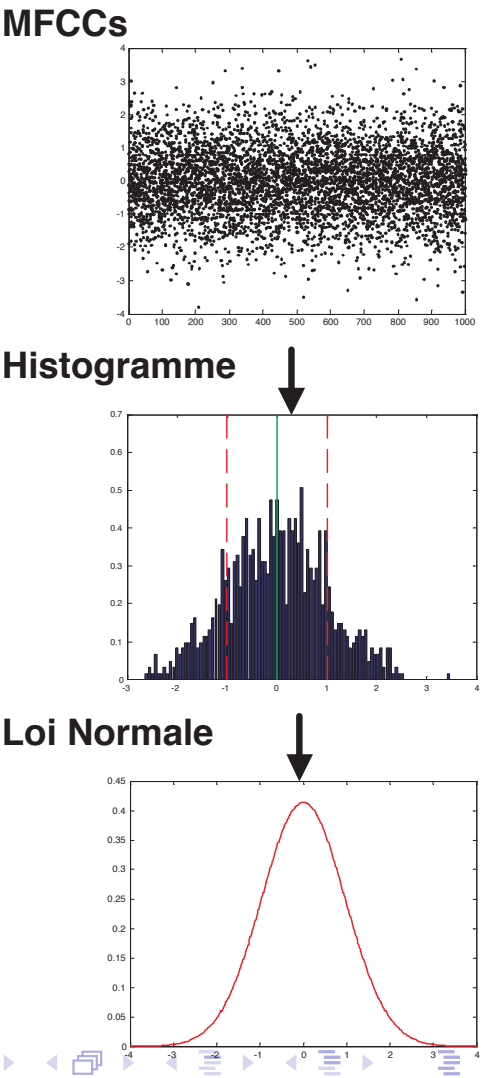
- On analyse la distribution (histogramme) de tous les MFCCs extrait au cours du temps (sans prendre en compte le temps)
- La distribution de chaque dimension (13 dim), suit une loi normale (gaussienne)
- On modélise le paquet de MFCCs par une loi normale à 13 dimensions
- Loi normale
  - Densité normale de probabilité à 1 dimension

$$p(X|\omega) = \frac{1}{\sqrt{2\pi}\sigma} e^{-1/2\left(\frac{x-\mu}{\sigma}\right)^2} \quad (6)$$

  - Densité normale de probabilité à d dimensions

$$p(X|\omega) = \frac{1}{(2\pi)^{d/2}|\Sigma|^{1/2}} e^{-1/2(x-\mu)^t \Sigma^{-1}(x-\mu)} \quad (7)$$

    - $\mu = E[x] = \int_x x p(x) dx$
    - $\Sigma = E[(x-\mu)(x-\mu)^t] = \int (x-\mu)(x-\mu)^t p(x) dx$
  - Différents types de matrice de covariance Σ

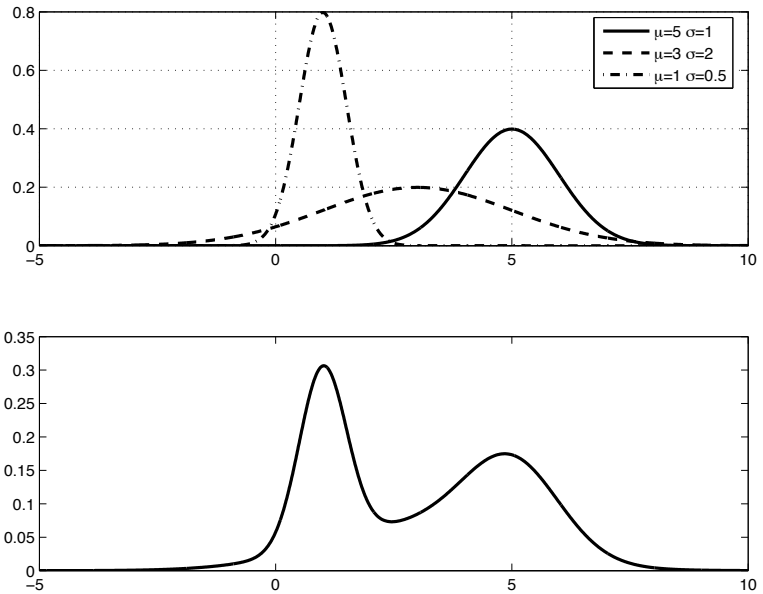


$P(X|\omega)$? Le modéles de mélange de gaussiennes (GMM)

- Si la distribution de X ne peut pas être approximée par une loi normale
- On approxime la distribution par une somme pondérée de fonction gaussienne:
  - $p(X|\omega) = \sum_{m=1}^{M} \pi_m f_m(x)$
  - On additionne M fonction gaussienne $f_m(x)$, poids respectifs π_m
  - $\sum_{i=m}^{M} \pi_m = 1$
- Chaque fonction gaussienne $f_m(x)$ à ses propres paramètres μ_m, Σ_m

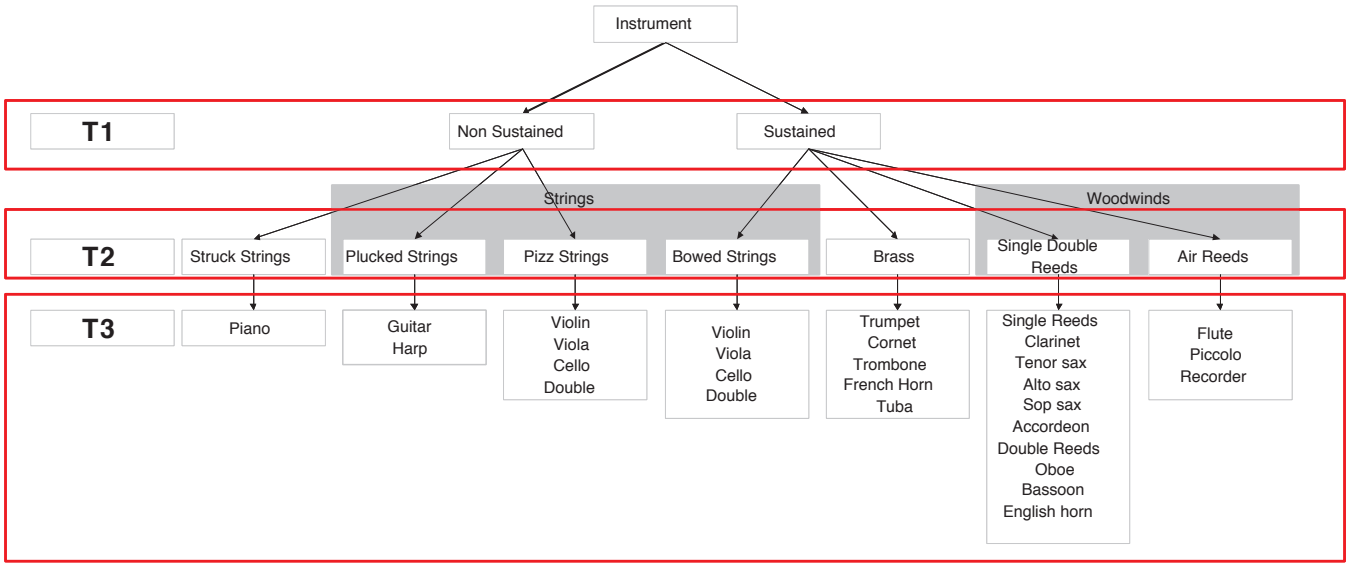
$$f_m(x) = \frac{1}{(2\pi)^{d/2}|\Sigma_m|^{1/2}} e^{-1/2(x-\mu_m)^t \Sigma_m^{-1}(x-\mu_m)} \quad (8)$$

- Estimation ? Algorithme "Expectation/ Maximization"

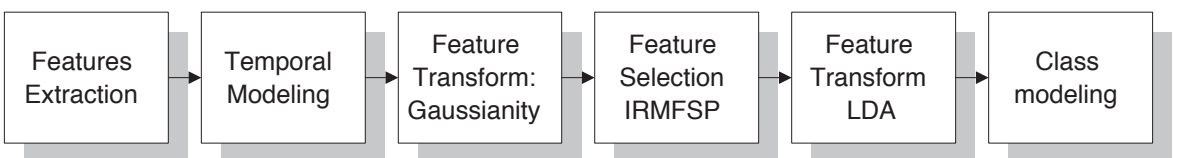


Exemple d'utilisation: reconnaissance des instruments de musique

- Reconnaissance de $\omega = 27$ instruments de musique



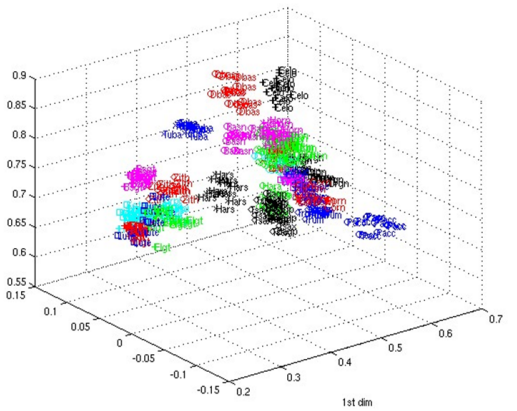
- Système utilisé



- Plusieurs base de données d'apprentissage (généralisation):
  - Ircam Studio OnLine (1323 sons), Iowa University database (816), McGill (585), Microsoft (216), Pro (532) and Vi (691)

Exemple d'utilisation: reconnaissance des instruments

- Représentation des descripteurs dans un espace 3D
- Matrice de confusion
- VIDEO: Arie Livshin: reconnaissance en temps-réel



		original class																						
		piano	guitar	harp	viola-pizz	bass-pizz	cello-pizz	violin-pizz	viola	bass	cello	violin	french-horn	cornet	trombone	trumpet	tuba	flute	piccolo	recorder	bassoon	clarinet	english-horn	oboe
recognized class	piano	36	3	4	4	2	2	1				1					2							
	guitar	29	48	12		1	8																	
	harp	24	22	68	2	3	5	2																
	viola-pizz	1		6	85		1	9											4		2			
	bass-pizz	1	3			76	12																	
	cello-pizz	2	20	2	4	18	71				1						1							
	violin-pizz	3	1		6		1	88										2	4		3			
	viola								44		5	14												
	bass			2						93	4	6					1	2			1	3		
	cello	1							37	5	68	16	1					2			1	1	5	
	violin								14		3	55						1					10	1
	french-horn	1										1	50		13	1	15		4	5	2			
	cornet										2		1	30	3	13		2	1					4
	trombone												15	15	49	7		1	1		2			
	trumpet										1			47	10	61				3		2		2
	tuba	2		2									23		7		79							
	flute										1	2	5	2	3	4	1	77	10	10	1	23	2	4
	piccolo															1		4	71	5		5		8
	recorder									1								2	4	59				
	bassoon	1	4	5							1		2	2	12					3	81		12	1
	clarinet	1				1			2	1	7	2				4		5	1	10		46	10	20
	english-horn										1	1	1							3	3	1	12	4
	oboe										4	1		4		9		3	1	3	1	14	49	58
	number of sounds	146	159	130	54	186	170	97	225	280	356	264	242	53	202	157	140	323	83	39	203	212	41	184

Évaluation des résultats d'un système de classification

- Critère de généralisation: ensemble d'évaluation différents de celui d'apprentissage
  - Ensemble d'apprentissage et d'évaluation différents
  - Validation croisée (N-fold cross validation)= sub-division de la base en N partie indépendantes: (N-1) pour l'entrainement, 1 pour l'évaluation.
- Mesures
  - TP: True Positif: Positif $\rightarrow \hat{Positif}$
  - FP: False Positif: Negatif $\rightarrow \hat{Positif}$
  - TN: True Negatif: Negatif $\rightarrow \hat{Negatif}$
  - FN: False Negatif: Positif $\rightarrow \hat{Negatif}$
- Matrice de confusion
- Résumé
  - Recall $= \frac{TP}{TP+FN}$: $\frac{nombre-de-documents-correctement-detectes}{nombre-total-de-documents-a-detecter}$
  - Precision $= \frac{TP}{TP+FP}$: $\frac{nombre-de-document-correctement-detectes}{nombre-total-de-documents-detectes}$
  - F-measure $= \frac{2Recall \cdot Precision}{Recall+Precision}$

Detected

True	Positif	Negatif
Positif	TP	FN
Negatif	FP	TN

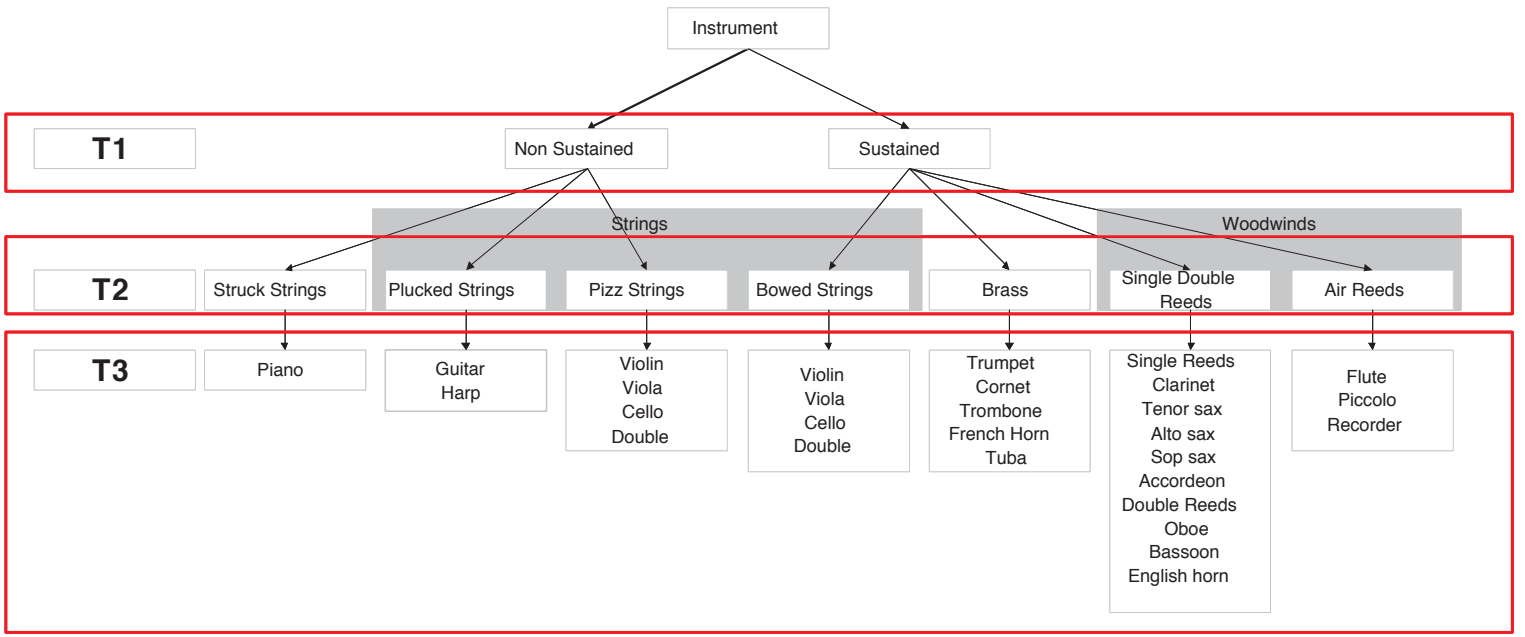
Recall=TP/(TP+FN)

Precision=TP/(TP+FP)

Différentes stratégies de classification

Classification à plat:

Classification hiérarchique (meilleurs résultats, introduction de connaissance sur l'organisation des classes)



Example d'utilisation: reconnaissance du genre musical

6 Classes:

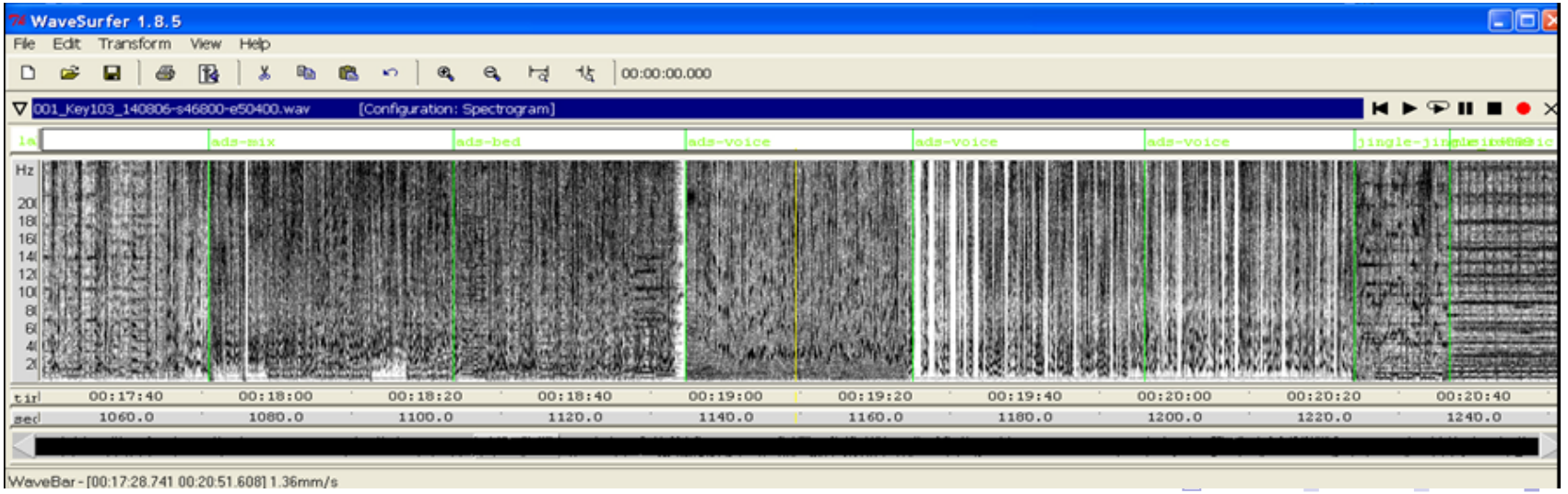
		Found						
		classical	electronic	jazz_blues	metal_punk	rock_pop	world	
Real	classical	90,6	0,0	0,3	0,0	0,0	**9,1**	
	electronic	1,8	73,7	0,9	2,6	**9,6**	11,4	
	jazz_blues	0,0	0,0	96,2	0,0	3,8	0,0	
	metal_punk	0,0	0,0	0,0	84,4	**15,6**	0,0	
	rock_pop	0,0	4,9	2,9	**16,7**	67,6	7,8	
	world	**16,4**	4,9	4,9	0,8	13,1	59,8	
								78,7

21 Classes: le taux de reconnaissance décroit généralement avec le nombre de classes, présence de classes ambiguës

'real/found'	'Pop'	'Blues'	'Rock'	'Hip-Hop'	'Chanson'	'randb'	'Folk'	'Soul'	'Jazz'	'Reggae'	'Rap'	'Classical'	'Alternative'	'Trip-Hop'	'Electronic'	'House'	'Synthpop'	'Dance'	'Latin'	'Disco'	'Metal'
'Pop'	**1,97**	0,478	9,791	2,687	2,09	6,448	4,299	5,612	10,51	1,612	1,313	14,45	8	3,284	2,03	0,716	3,045	1,851	2,209	2,866	14,75
'Blues'	0	**4,225**	15,49	2,817	2,113	2,113	11,27	7,042	9,155	2,817	3,521	16,2	3,521	1,409	0,704	0	2,817	0,704	4,225	1,409	8,451
'Rock'	0,205	1,489	**10,93**	0,77	2,926	2,361	4,312	4,363	4,826	0,667	1,027	12,53	12,99	2,515	1,181	0,975	2,721	0,513	1,848	1,232	29,62
'Hip-Hop'	0	0,98	2,941	**20,59**	0	5,882	0,98	2,941	0,98	9,804	39,22	1,961	0	0,98	3,922	0	0	0,98	3,922	2,941	0,98
'Chanson'	0,773	1,468	7,883	0,927	**19,09**	2,859	8,578	2,473	4,946	1,932	3,014	18,24	3,4	2,164	0,696	0,541	1,855	1,855	4,173	1,777	11,36
'randb'	1,254	0,94	3,762	7,837	0,627	**42,32**	1,881	9,718	6,583	3,135	3,135	4,389	1,254	0,627	1,881	0,94	1,567	0,94	3,762	1,881	1,567
'Folk'	0	4	10,4	0,8	6,4	5,6	**26,4**	4	6,4	2,4	1,6	17,6	1,6	0,8	1,6	0,8	0,8	0	6,4	0,8	1,6
'Soul'	1,333	2,667	5,333	1,333	1,333	14,67	1,333	**28**	21,33	1,333	0	4	4	2,667	0	0	2,667	2,667	1,333	1,333	2,667
'Jazz'	0,481	2,404	6,25	0,481	2,885	4,808	4,808	5,769	**44,71**	1,923	1,923	10,1	0,481	2,404	2,404	0	0,481	0,481	1,442	0,481	5,289
'Reggae'	0	1,508	4,02	9,045	3,015	7,538	4,02	3,518	0,503	**28,64**	13,57	1,508	2,01	1,508	0,503	1,508	0	0,503	8,04	1,508	7,538
'Rap'	0	0,709	0,473	8,747	1,655	1,891	0,236	0,236	0	3,783	**75,89**	0	0,946	0,473	0,473	0,709	0	0,236	0,946	0	2,601
'Classical'	0	1,429	0,714	0	0,714	0	0,714	0,714	4,286	1,429	0	**82,86**	0	0	0,714	0	0,714	0,714	5	0	0
'Alternative'	0	1,818	10	4,546	0,909	0	0	0	1,818	2,727	1,818	6,364	**31,82**	0,909	3,636	0,909	2,727	0,909	1,818	1,818	25,46
'Trip-Hop'	0	0	0	0	2,817	1,409	1,409	5,634	5,634	0	4,225	4,225	8,451	**33,8**	11,27	0	2,817	2,817	0	0	15,49
'Electronic'	0	0	3,401	6,122	1,361	2,041	2,041	2,721	2,041	4,082	2,041	6,122	5,442	11,57	**14,97**	8,844	2,721	3,401	0,68	2,721	17,69
'House'	0	0	2,326	2,326	0	2,326	1,163	1,163	0	0	0	0	1,163	3,488	8,14	**34,88**	11,63	19,77	0	8,14	3,488
'Synthpop'	0	0	2,299	0	1,149	0	0	0	0	0	1,149	5,747	5,747	2,299	3,448	0	**49,43**	4,598	3,448	1,149	19,54
'Dance'	1,242	0	0,621	3,106	0	0,621	0	0	3,106	0,621	3,727	3,106	0,621	0,621	3,106	11,8	3,106	**54,04**	0,621	4,348	5,59
'Latin'	1,053	5,263	0	1,053	5,263	7,368	6,316	2,105	4,211	7,368	2,105	5,263	4,211	0	0	0	0	1,053	**46,32**	0	1,053
'Disco'	1,515	1,515	1,515	3,03	0	1,515	0	10,61	0	0	1,515	3,03	6,061	0	12,12	6,061	1,515	6,061	0	**42,42**	1,515
'Metal'	0	0	2,857	4,286	1,429	0	1,429	0	0	0	0	8,571	10	0	1,429	0	4,286	0	0	0	**65,71**

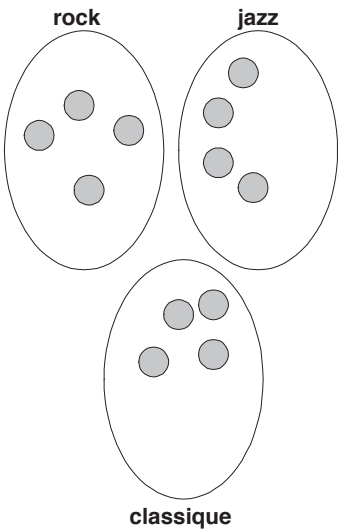
Exemple d'utilisation: segmentation parole/ musique d'un flux audio

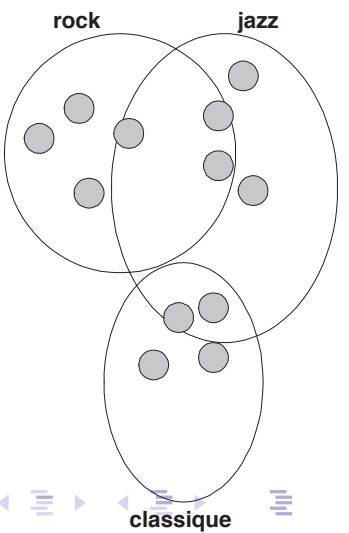
- Objectif: on cherche à segmenter un flux audio (radio, télé) en catégories $\omega =$ parole et $\omega =$ musique.
- Méthode
  1. a) On classe chaque seconde du flux en catégorie "parole" ou "musique" $\omega(O(m))$, b) on lisse $\omega(O(m))$ au cours du temps afin d'éviter les changements de classes parasites (parole dans la musique Rap), c) on segmente aux instants de changements de classes
  2. a) On segmente le flux dés qu'il y a un changement important de contenu, b) on classe ensuite les segments en "parole" ou "musique"
- VIDEO reconnaissance en temps-réel



Le multi-label

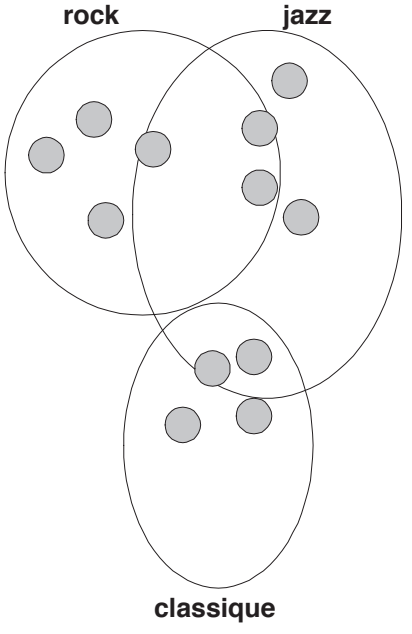
- Mono-Label:
  - Une observation O (un morceau de musique) appartient à une seule classe (les classes sont mutuellement excusives)
  - Apprentissage: on cherche à maximiser la discrimination entre les classes
  - Evaluation: on assigne à O la classe ω_i pour lesquelles $P(\omega_i|X)$ est maximale
  - Remarque: On peut éventuellement avoir plusieurs systèmes de classes orthogonaux ($\Omega = rock, classique, jazz$, $\Omega = studio, live$)
- Multi-label:
  - Une observation O peut appartenir simultanément à plusieurs classes (les classes ne sont pas exclusives)
  - Exemple: un morceau peut être un mélange de rock et de jazz, il appartient aux deux classes





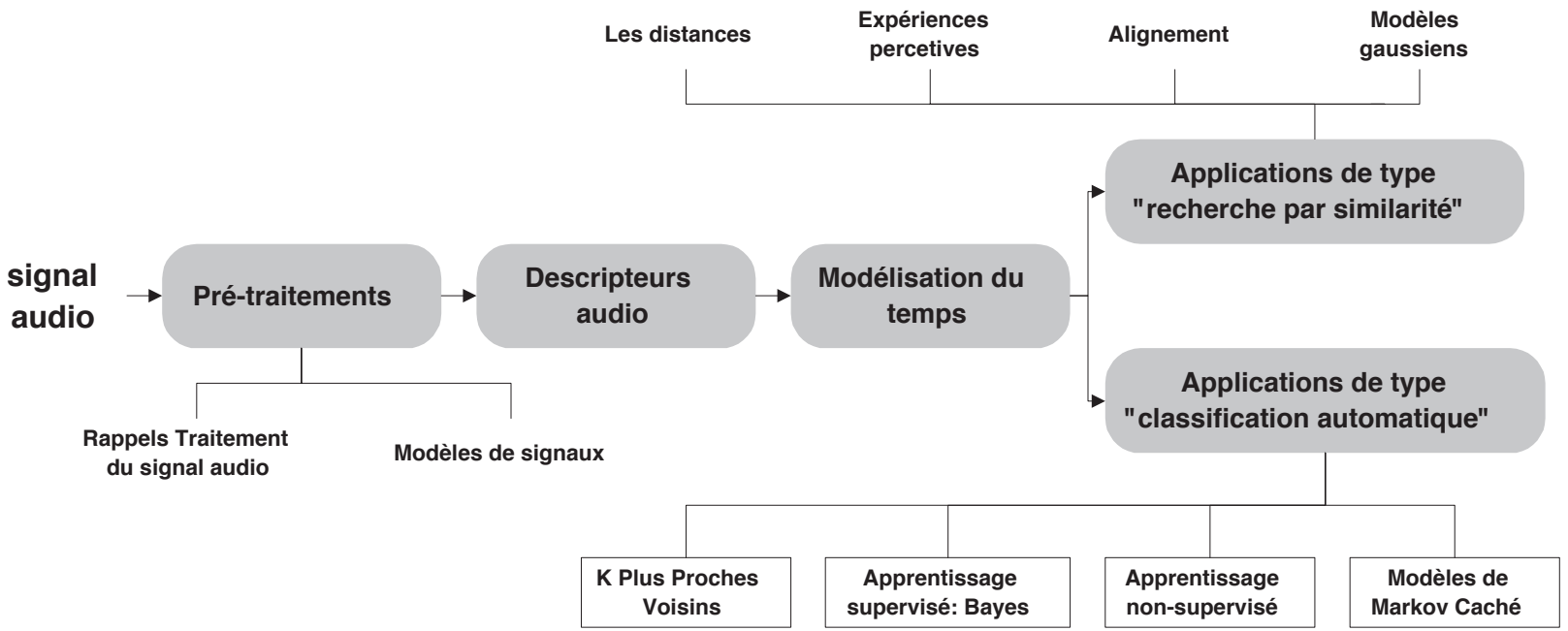
Le multi-label

- Différentes stratégies d'apprentissage
  - Binary Relevance:
    - on considère chaque label comme un sous problème indépendant: on entraine des classificateurs pour détecter ω_i versus $\neg\omega_i$ (un morceau est rock ou pas rock)
    - on combine l'ensemble des décisions individuelles (un morceau est rock, jazz, pas-classique)
  - Label powersets:
    - On crée des méta-classes correspondant à toutes les combinaisons possibles des classes: (rock/pas-jazz/pas-classique, rock/jazz/pas-classique, rock/jazz/classique, pas-rock/jazz/pas-classique, ...)
- Applications:
  - Reconnaissance d'instruments multiples
  - Music multi-label (cnfr nuage de tags de Last-FM): genre, mood



d'après Burred

Plans


Les distances
Expériences percetives
Alignement
Modèles gaussiens
Applications de type "recherche par similarité"
signal audio
Pré-traitements
Descripteurs audio
Modélisation du temps
Applications de type "classification automatique"
Rappels Traitement du signal audio
Modèles de signaux
K Plus Proches Voisins
Apprentissage supervisé: Bayes
Apprentissage non-supervisé
Modèles de Markov Caché

Classification supervisée/ non-supervisée

L'apprentissage supervisé: Un expert (ou oracle) est employé pour étiqueter correctement des exemples. L'apprenant doit alors trouver ou approximer la fonction qui permet d'affecter la bonne étiquette à ces exemples.

L'apprentissage non-supervisé: Aucun expert n'est disponible. L'algorithme doit découvrir par lui-même la structure des données. Le clustering et les modèles de mélanges de gaussiennes sont des algorithmes d'apprentissage non supervisés.

from wikipedia

Clustering: introduction

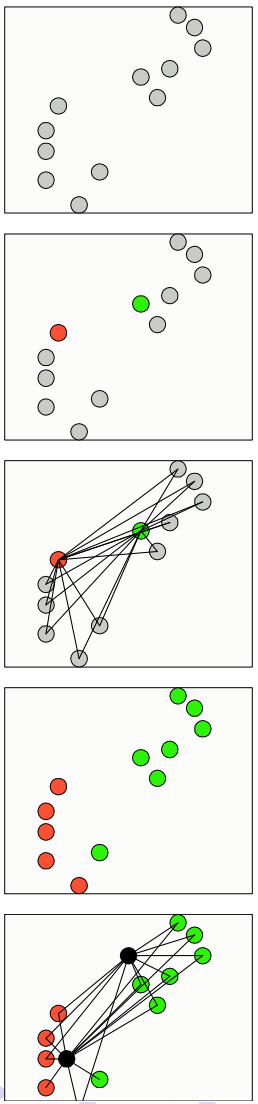
Algorithmes de clustering

- Clustering ?
  - Processus qui partitionne un ensemble de données en sous-classes (clusters) ayant du sens
  - Algorithme permettant de trouver la structure sous-jacente à un ensemble de données
  - Apprentissage non-supervisé (par opposition à l'apprentissage supervisé: Bayes, ...)
- Deux grandes classes d'algorithmes:
  - Algorithmes de **partitionnement**: divise un ensemble de N items en K clusters, tous les clusters sont considérés simultanément
    - K-means, Fuzzy-K-Means
  - Algorithme **hiérarchiques**:
    - par **agglomération**: les paires d'objets ou de clusters sont successivement liés pour produire des clusters plus grand (bottom-up)
    - par **division**: les clusters sont successivement diviser en de plus en plus petits clusters (top-down)

Algorithmes de partitionnement

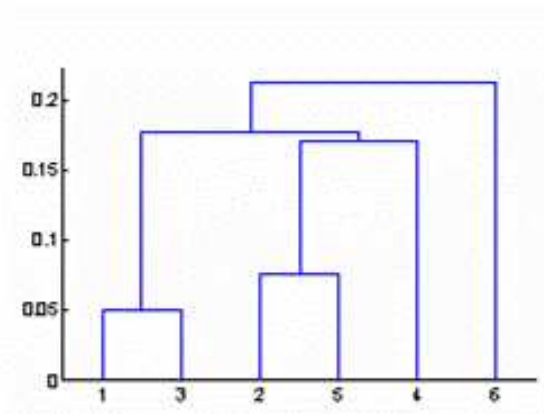
Algorithme K-Means (nuées dynamiques)

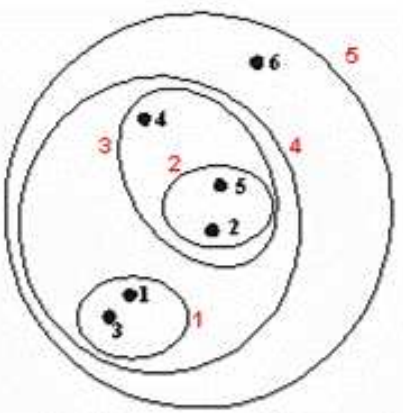
- **Initialisation:**
  - choisir K sous-ensembles (clusters) initiaux
  - calculer le centroide de chaque sous-ensemble (cluster) à partir des objets attribués à ce sous-ensemble (cluster)
  - différentes méthodes pour l'initialisation: random, KD-tree, ...
- **Etape E:** attribuer chaque objet au sous-ensemble (cluster) dont il est le plus proche (distance euclidienne)
- **Etape M:** étant donné la nouvelle attribution des objets aux sous-ensemble (clusters) recalculer les centroides (moyenne arithmétique)
- **Itération:** réitérer jusqu'à ce que les objets ne bougent plus (ou que la valeur des centroides ne bougent plus)

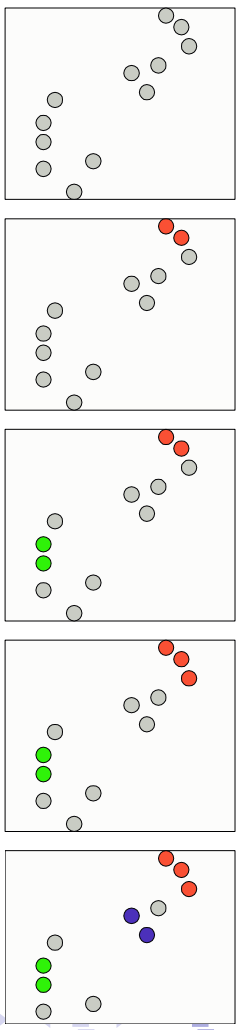


Algorithmes hiérarchiques par agglomération (Hierarchical Agglomerative Clustering)

- **Initialisation:** chaque objet constitue un cluster
- **Itération:** regroupement des objets ou clusters les plus proches
- **Condition d'arrêt:** on arrive au sommet de l'arbre, ou bien on a obtenu K clusters

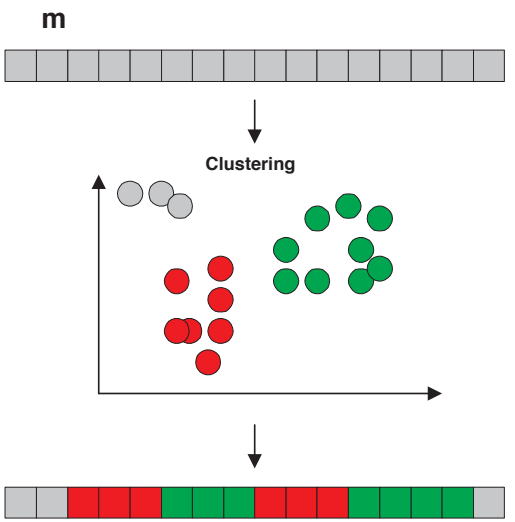




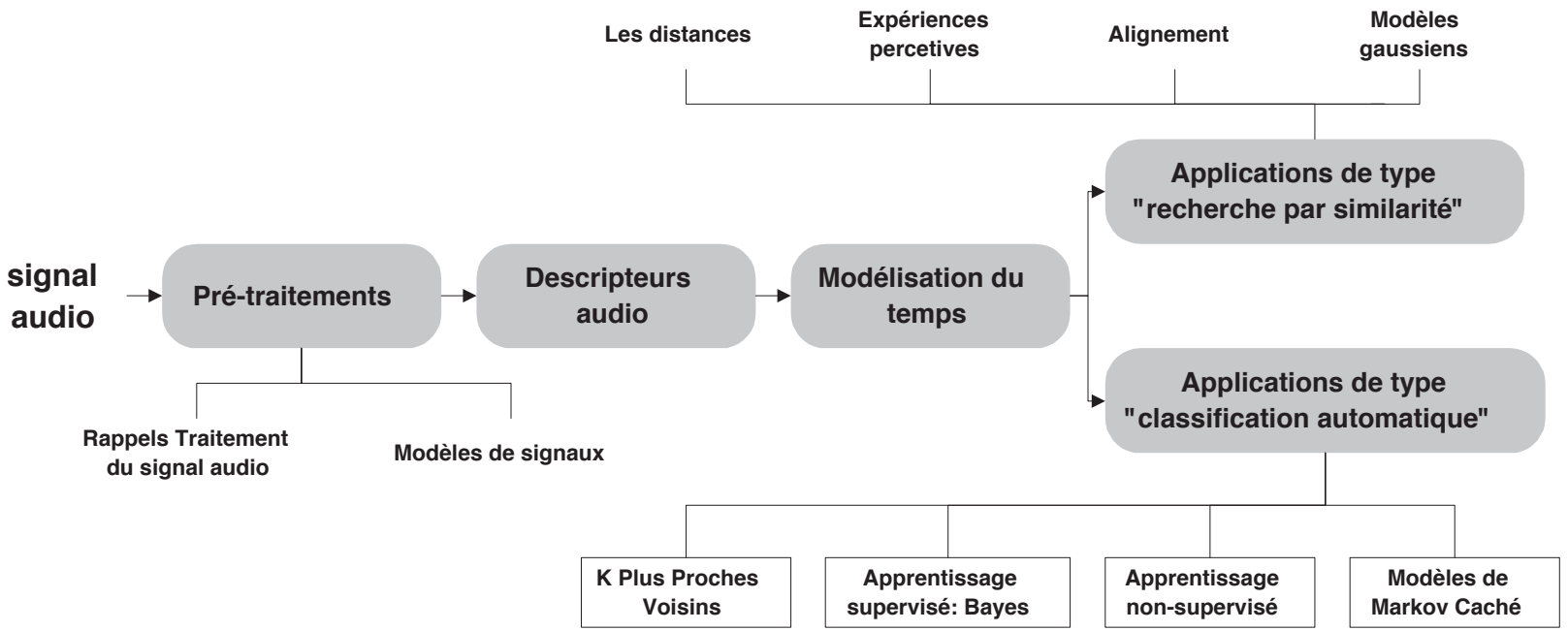


Exemple: détection de la structure musicale d'un morceau

- Objectif: on cherche à détecter la structure d'un morceau de musique en termes de répétitions de parties (couplet, refrain)
- On ne peut pas faire d'entrainement statistique
  - les éléments créant la structure sont propres à chaque morceau
- Méthode:
  - On extrait des observations du signal audio $O(m)$ représentant les différents instants m du morceau
  - On choisit typiquement les MFCCs + Chroma/PCP + SFMs
  - On utilise un algorithme de clustering afin de regrouper les observations $O(m)$ en "groupes"
  - Idéalement, à la sortie de l'algorithme les différents "groupes" représentent chacun soit les couplets, soit les refrains

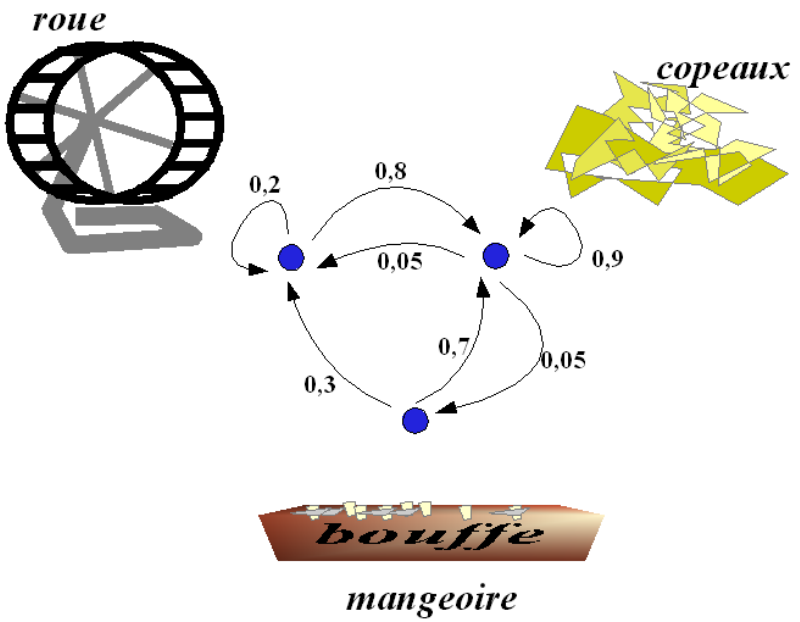


Plans


Les distances
Expériences percetives
Alignement
Modèles gaussiens
Applications de type "recherche par similarité"
signal audio
Pré-traitements
Descripteurs audio
Modélisation du temps
Applications de type "classification automatique"
Rappels Traitement du signal audio
Modèles de signaux
K Plus Proches Voisins
Apprentissage supervisé: Bayes
Apprentissage non-supervisé
Modèles de Markov Caché

Les modèles de Markov Cachés (Hidden Markov Model)

- Markov: un mathématicien Russe
- Chaîne de Markov:
  - un processus stochastique à temps discret pouvant être dans des états discrets,
  - la prédiction de l'état actuel ne dépend que de l'instant précédent.
- Les états discrets: $S_0, S_1, ..., S_k$;
- $S(n)$ la valeur de l'état à l'instant n
- Exemple: Doudou le hamster
  - 3 états: dormir, manger, exercice
  - Matrice transition

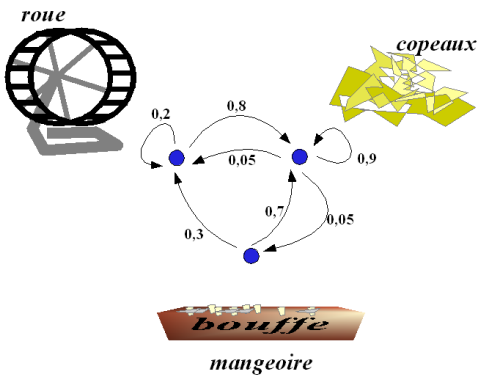


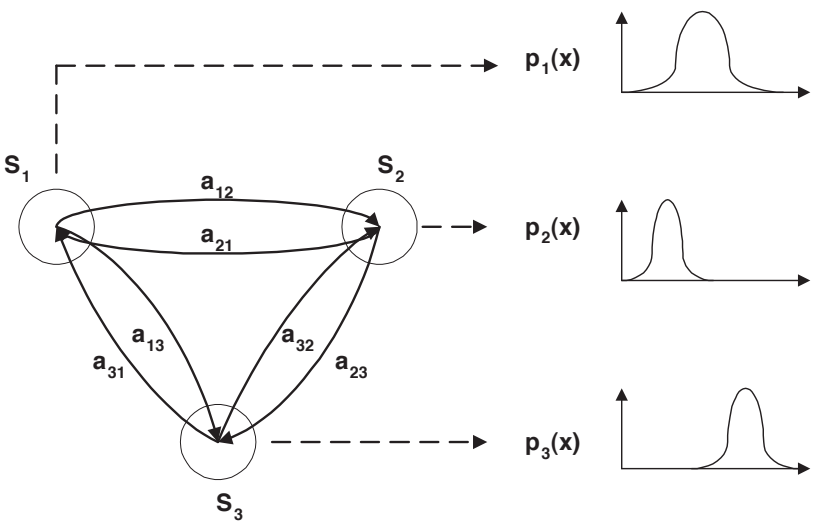
$$T = P(S_i(n-1), S_j(n)) \begin{pmatrix} 0.9 & 0.05 & 0.05 \\ 0.7 & 0 & 0.3 \\ 0.8 & 0 & 0.2 \end{pmatrix} \quad (9)$$

from wikipedia

Les modèles de Markov Cachés (Hidden Markov Model)

- Chaîne de Markov **cachée** ?
  - On observe pas directement les états (cachés) mais une émission des états (exemple le bruit fait par Doudou le hamster)
- Définition d'un modèle de Markov caché, λ:
  - Définition des états S_i
  - π_i la probabilité initiale (a priori) d'observer chaque état au temps $n = 0$
  - $a_{i,j}$ les probabilités de transitions entre états (matrice de transition): $S_i(n-1) \rightarrow S_j(n)$
  - p_i: la probabilité d'émission de chaque état S_i (quel est le son émis quand Doudou dort, quand il mange, quand il tourne dans la roue)

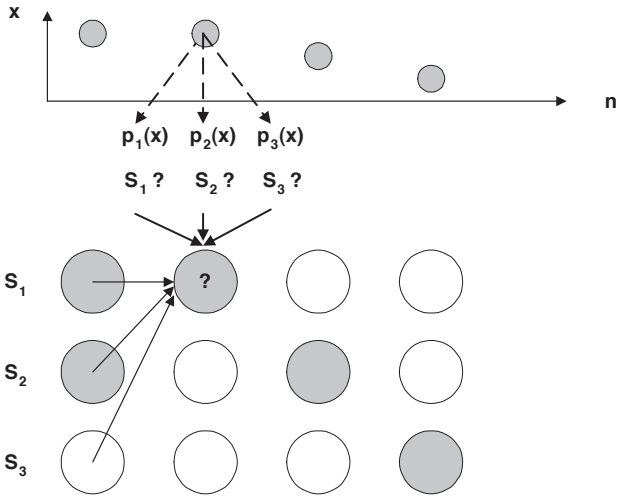




Les modèles de Markov Cachés (Hidden Markov Model)

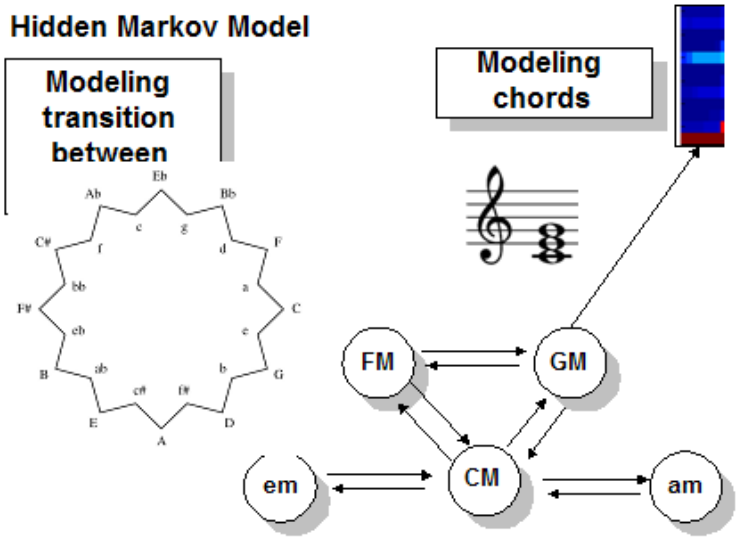
Trois problèmes:

- **Décodage:**
  - Étant donné une séquence d'observations $O(m)$ et un modèle λ, trouver la meilleur séquence d'états au cours du temps
  - si on observe la séquence du son de Doudou et étant donné son modèle dormir/manger/exercice, quel est la séquence d'activités de Doudou ?
- **Reconnaissance:**
  - Quelle est la probabilité qu'une séquence d'observation $O(m)$ ai été émise par un modèle λ: $P(O|\lambda)$
  - comment déterminer si une séquence d'observation du son correspond au modèle dormir/manger/exercice de Doudou le hamster, ou à un modèle dormir/manger/travailler de Bill le salarié
- **Entrainement:**
  - Étant donné un ensemble de séquences d'observations $O(m)$, comment trouver les paramètres du modèle λ: maximiser la probabilité $P(O|\lambda)$
  - comment créer le modèle de Doudou le hamster ?



Exemple (décodage): reconnaissance d'accords

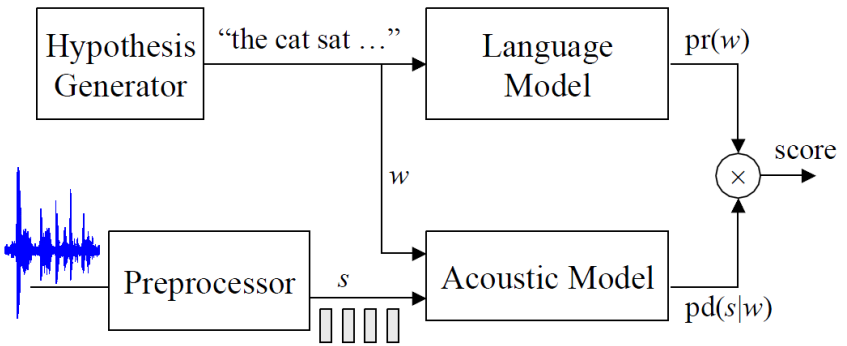
- Objectif:
  - On veut estimer la suite d'accords (C-Major, C#-Majeur, ..., C-mineur, ...) d'un morceau de musique à partir de l'observation de son signal audio
- Méthode
  - Observation: on extrait la séquence de descripteurs audio chroma/PCP
  - Probabilité d'émission de chaque accord: $p(S = C - majeur|chroma)$, $p(S = Db - majeur|chroma)$, ...
  - Transitions entre accords: suivent la théorie musicale (cercle des quintes, relatifs majeur-mineur): Sol-Majeur vers Do-Majeur (consonance), Sol-Majeur vers Do# Majeur (dissonance)
- Solution: on estime la suite d'accords par décodage d'un modèle de Markov Caché
  - États (=24 accords), Probabilité d'émission, Probabilité de transition, Probabilité initiale

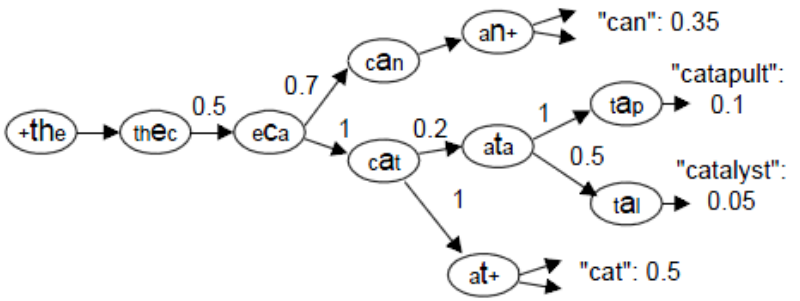


Exemple (reconnaissance): système simple de reconnaissance de parole

Quatre grandes parties:

- 1) Modèle de langage:
  - Estimation de la probabilité d'une séquence de mots
- 2) Phonétiseur:
  - Transformation des mots en séquence de phonèmes
    - Phonème: plus petite unité distinctive que l'on puisse isoler: cote (/k?t/) et côte (/kot/)
    - Pour une même langue, accent $\rightarrow$ plusieurs prononciations: petit, p'tit
    - 37 phonèmes en français
  - Transformation des mots en séquence de tri-phones
    - 37^3 tri-phone en français

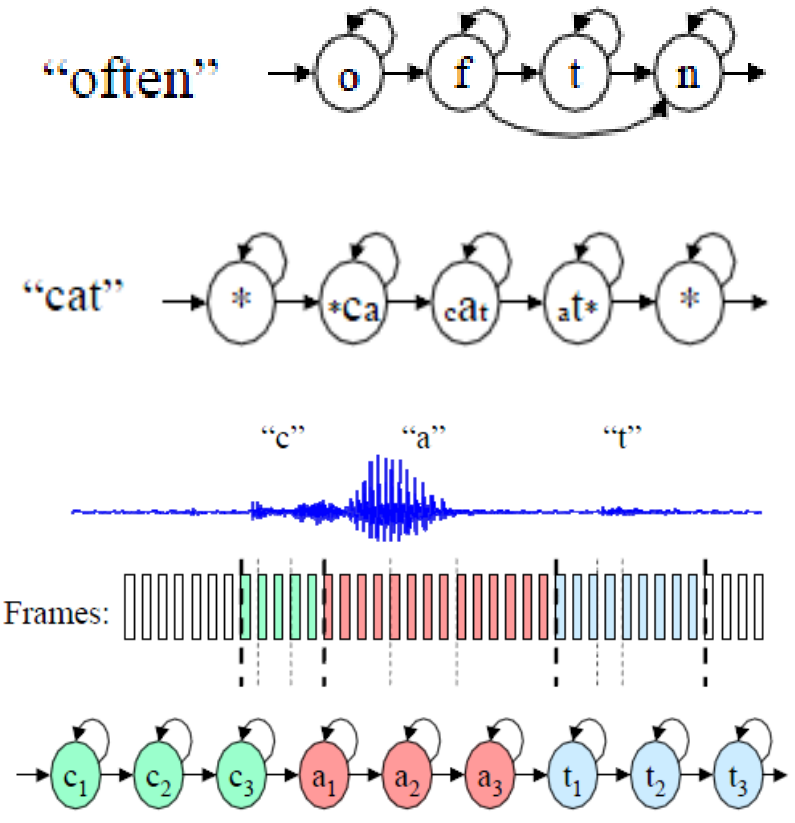




d'après Brookes

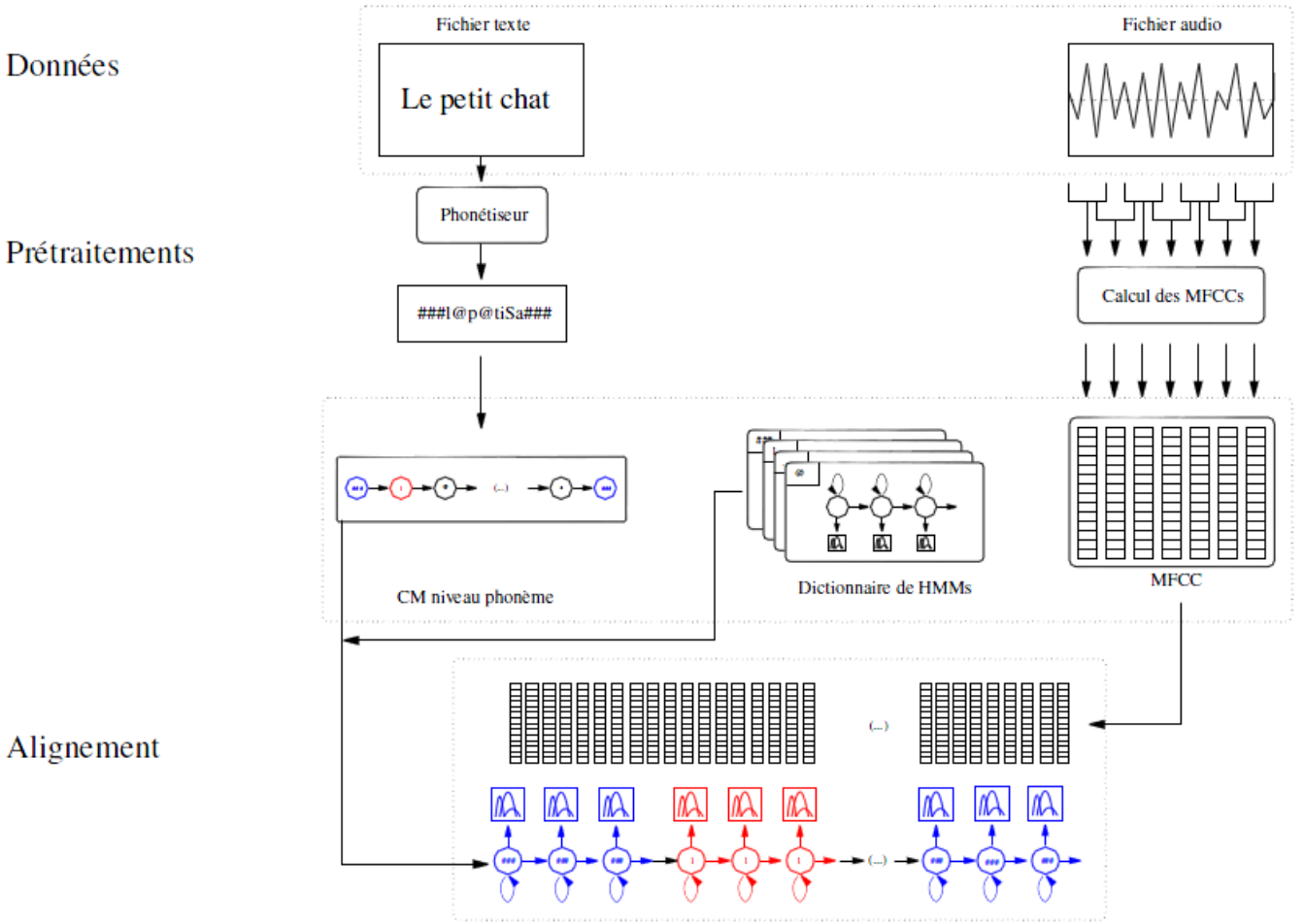
Exemple (reconnaissance): système simple de reconnaissance de parole

- 3) Pré traitement:
  - extraction de MFCC + Δ MFCC + ΔΔ MFCC (L=25ms, 40 bandes, 3*39 coefficients)
- 4) Modèle acoustique:
  - HMM permettant la jonction entre un phonème / tri-phone et les différentes occurrences acoustiques (différents locuteurs)



d'après Brookes

Exemple (decodage): alignement de parole



d'après Lanchantin