

# Text-Independent Speaker Recognition Using Gaussian Mixture Models

Eduardo Martins Barros de Albuquerque Tenório

Centro de Informática  
Universidade Federal de Pernambuco  
Trabalho de Graduação em Engenharia da Computação  
*embat@cin.ufpe.br*

Recife, 25 de Junho de 2015

# Conteúdo

- 1 Introdução
- 2 Sistemas de Reconhecimento de Locutor
- 3 Extração de Características
- 4 Modelos de Mistura Gaussianas
- 5 Experimentos
- 6 Conclusão

# Conteúdo

- 1 Introdução
- 2 Sistemas de Reconhecimento de Locutor
- 3 Extração de Características
- 4 Modelos de Mistura Gaussianas
- 5 Experimentos
- 6 Conclusão

# Reconhecimento de ...

Fala **O que** está sendo dito

# Reconhecimento de ...

Fala **O que** está sendo dito

- Conteúdo da mensagem

# Reconhecimento de ...

Fala **O que** está sendo dito

- Conteúdo da mensagem
- Estado emocional do locutor

# Reconhecimento de ...

Fala **O que** está sendo dito

- Conteúdo da mensagem
- Estado emocional do locutor
- Sotaque ou dificuldade de articulação

# Reconhecimento de ...

Fala **O que** está sendo dito

- Conteúdo da mensagem
- Estado emocional do locutor
- Sotaque ou dificuldade de articulação

Locutor **Quem** está falando



# Reconhecimento de ...

Fala **O que** está sendo dito

- Conteúdo da mensagem
- Estado emocional do locutor
- Sotaque ou dificuldade de articulação

Locutor **Quem** está falando

- Identificar uma pessoa num grupo

# Reconhecimento de ...

Fala **O que** está sendo dito

- Conteúdo da mensagem
- Estado emocional do locutor
- Sotaque ou dificuldade de articulação

Locutor **Quem** está falando

- Identificar uma pessoa num grupo
- Autenticar um usuário

# Reconhecimento de ...

Fala **O que** está sendo dito

- Conteúdo da mensagem
- Estado emocional do locutor
- Sotaque ou dificuldade de articulação

Locutor **Quem** está falando

- Identificar uma pessoa num grupo
- Autenticar um usuário

Este trabalho é focado em reconhecimento de **locutor**

# Reconhecimento de Locutor

**Identificação** Determina a identidade de um locutor dentro de um conjunto não unitário

# Reconhecimento de Locutor

**Identificação** Determina a identidade de um locutor dentro de um conjunto não unitário

- 1 para N

# Reconhecimento de Locutor

**Identificação** Determina a identidade de um locutor dentro de um conjunto não unitário

- 1 para N
- Problema de **conjunto fechado**

# Reconhecimento de Locutor

**Identificação** Determina a identidade de um locutor dentro de um conjunto não unitário

- 1 para N
- Problema de **conjunto fechado**

**Verificação** Determina se o locutor é quem diz ser

# Reconhecimento de Locutor

**Identificação** Determina a identidade de um locutor dentro de um conjunto não unitário

- 1 para N
- Problema de **conjunto fechado**

**Verificação** Determina se o locutor é quem diz ser

- 1 para 1



# Reconhecimento de Locutor

**Identificação** Determina a identidade de um locutor dentro de um conjunto não unitário

- 1 para N
- Problema de **conjunto fechado**

**Verificação** Determina se o locutor é quem diz ser

- 1 para 1
- Problema de **conjunto aberto**

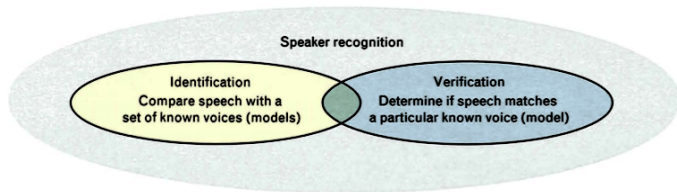
# Reconhecimento de Locutor

**Identificação** Determina a identidade de um locutor dentro de um conjunto não unitário

- 1 para N
- Problema de **conjunto fechado**

**Verificação** Determina se o locutor é quem diz ser

- 1 para 1
- Problema de **conjunto aberto**



# Dependência de texto

Dependente Teste  $\in$  Treinamento

# Dependência de texto

Dependente Teste  $\in$  Treinamento

- Diversos graus de dependência

# Dependência de texto

Dependente Teste  $\in$  Treinamento

- Diversos graus de dependência
- Teste  $\notin$  Treinamento  $\implies$  Retreinamento

# Dependência de texto

Dependente Teste  $\in$  Treinamento

- Diversos graus de dependência
- Teste  $\notin$  Treinamento  $\implies$  Retreinamento

Independente Teste  $\neq$  Treinamento

# Dependência de texto

Dependente Teste  $\in$  Treinamento

- Diversos graus de dependência
- Teste  $\notin$  Treinamento  $\implies$  Retreinamento

Independente Teste  $\neq$  Treinamento

- Características não textuais

# Dependência de texto

Dependente Teste  $\in$  Treinamento

- Diversos graus de dependência
- Teste  $\notin$  Treinamento  $\implies$  Retreinamento

Independente Teste  $\neq$  Treinamento

- Características não textuais
- Presentes em diferentes sotaques e até *gibberish*



# Dependência de texto

Dependente Teste  $\in$  Treinamento

- Diversos graus de dependência
- Teste  $\notin$  Treinamento  $\implies$  Retreinamento

Independente Teste  $\neq$  Treinamento

- Características não textuais
- Presentes em diferentes sotaques e até *gibberish*

Este trabalho é focado em reconhecimento de locutor  
**independente de texto**

# Modelos de Mistura Gaussiana

## GMM **Combinação** de Gaussianas

# Modelos de Mistura Gaussiana

**GMM** **Combinação** de Gaussianas

**UBM** GMM gerado por diversas **locuções de fundo**

# Modelos de Mistura Gaussiana

**GMM** **Combinação** de Gaussianas

**UBM** GMM gerado por diversas **locuções de fundo**

**AGMM** GMM **adaptado** a partir de um UBM

# Modelos de Mistura Gaussiana

**GMM** **Combinação** de Gaussianas

**UBM** GMM gerado por diversas **locuções de fundo**

**AGMM** GMM **adaptado** a partir de um UBM

**FGMM** GMM utilizando **Fractional Covariance Matrix** (FCM)

# Objetivos

Implementar sistemas de reconhecimento de locutor e analisar:

# Objetivos

Implementar sistemas de reconhecimento de locutor e analisar:

- Taxas de **sucesso** para identificação

# Objetivos

Implementar sistemas de reconhecimento de locutor e analisar:

- Taxas de **sucesso** para identificação
  - Diferentes tamanhos de mistura ( $M$ )



# Objetivos

Implementar sistemas de reconhecimento de locutor e analisar:

- Taxas de **sucesso** para identificação
  - Diferentes tamanhos de mistura ( $M$ )
  - Diferentes tamanhos de características ( $\Delta$ )

# Objetivos

Implementar sistemas de reconhecimento de locutor e analisar:

- Taxas de **sucesso** para identificação
  - Diferentes tamanhos de mistura ( $M$ )
  - Diferentes tamanhos de características ( $\Delta$ )
- Comparar identificações utilizando GMM e FGMM

# Objetivos

Implementar sistemas de reconhecimento de locutor e analisar:

- Taxas de **sucesso** para identificação
  - Diferentes tamanhos de mistura ( $M$ )
  - Diferentes tamanhos de características ( $\Delta$ )
- Comparar identificações utilizando GMM e FGMM
- Taxas de **falsa detecção** e **falsa rejeição** para verificação

# Objetivos

Implementar sistemas de reconhecimento de locutor e analisar:

- Taxas de **sucesso** para identificação
  - Diferentes tamanhos de mistura ( $M$ )
  - Diferentes tamanhos de características ( $\Delta$ )
- Comparar identificações utilizando GMM e FGMM
- Taxas de **falsa detecção** e **falsa rejeição** para verificação
  - Diferentes tamanhos de mistura ( $M$ )

# Objetivos

Implementar sistemas de reconhecimento de locutor e analisar:

- Taxas de **sucesso** para identificação
  - Diferentes tamanhos de mistura ( $M$ )
  - Diferentes tamanhos de características ( $\Delta$ )
- Comparar identificações utilizando GMM e FGMM
- Taxas de **falsa detecção** e **falsa rejeição** para verificação
  - Diferentes tamanhos de mistura ( $M$ )
  - Diferentes tamanhos de características ( $\Delta$ )

# Objetivos

Implementar sistemas de reconhecimento de locutor e analisar:

- Taxas de **sucesso** para identificação
  - Diferentes tamanhos de mistura ( $M$ )
  - Diferentes tamanhos de características ( $\Delta$ )
- Comparar identificações utilizando GMM e FGMM
- Taxas de **falsa detecção** e **falsa rejeição** para verificação
  - Diferentes tamanhos de mistura ( $M$ )
  - Diferentes tamanhos de características ( $\Delta$ )
- Comparar verificações utilizando GMM e AGMM

# Conteúdo

- 1 Introdução
- 2 Sistemas de Reconhecimento de Locutor**
- 3 Extração de Características
- 4 Modelos de Mistura Gaussianas
- 5 Experimentos
- 6 Conclusão

# Identificação

Modelagem Para cada locutor  $S_j \in \mathcal{S}$



# Identificação

Modelagem Para cada locutor  $\mathcal{S}_j \in \mathcal{S}$

- Extrair  $\mathbf{X}_k$  dos sinais  $\mathbf{Y}_k$  falados por  $\mathcal{S}_j$

# Identificação

Modelagem Para cada locutor  $\mathcal{S}_j \in \mathcal{S}$

- Extrair  $\mathbf{X}_k$  dos sinais  $\mathbf{Y}_k$  falados por  $\mathcal{S}_j$
- Treinar um  $\lambda_j$  para cada  $\mathcal{S}_j$  através dos  $\mathbf{X}_k$

# Identificação

**Modelagem** Para cada locutor  $\mathcal{S}_j \in \mathcal{S}$

- Extrair  $\mathbf{X}_k$  dos sinais  $\mathbf{Y}_k$  falados por  $\mathcal{S}_j$
- Treinar um  $\lambda_j$  para cada  $\mathcal{S}_j$  através dos  $\mathbf{X}_k$

**Teste** Para um locutor desconhecido  $\mathcal{S}$

# Identificação

**Modelagem** Para cada locutor  $\mathcal{S}_j \in \mathcal{S}$

- Extrair  $\mathbf{X}_k$  dos sinais  $\mathbf{Y}_k$  falados por  $\mathcal{S}_j$
- Treinar um  $\lambda_j$  para cada  $\mathcal{S}_j$  através dos  $\mathbf{X}_k$

**Teste** Para um locutor desconhecido  $\mathcal{S}$

- Extrair  $\mathbf{X}$  do sinal  $\mathbf{Y}$  falado por  $\mathcal{S}$

# Identificação

**Modelagem** Para cada locutor  $\mathcal{S}_j \in \mathcal{S}$

- Extrair  $\mathbf{X}_k$  dos sinais  $\mathbf{Y}_k$  falados por  $\mathcal{S}_j$
- Treinar um  $\lambda_j$  para cada  $\mathcal{S}_j$  através dos  $\mathbf{X}_k$

**Teste** Para um locutor desconhecido  $\mathcal{S}$

- Extrair  $\mathbf{X}$  do sinal  $\mathbf{Y}$  falado por  $\mathcal{S}$
- $i = \arg_j \max p(\mathbf{X}|\lambda_j) \implies \mathcal{S} \leftarrow \mathcal{S}_i$

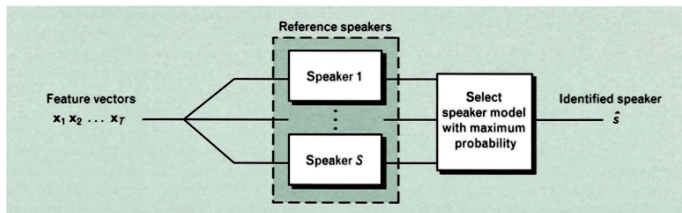
# Identificação

**Modelagem** Para cada locutor  $\mathcal{S}_j \in \mathcal{S}$

- Extrair  $\mathbf{X}_k$  dos sinais  $\mathbf{Y}_k$  falados por  $\mathcal{S}_j$
- Treinar um  $\lambda_j$  para cada  $\mathcal{S}_j$  através dos  $\mathbf{X}_k$

**Teste** Para um locutor desconhecido  $\mathcal{S}$

- Extrair  $\mathbf{X}$  do sinal  $\mathbf{Y}$  falado por  $\mathcal{S}$
- $i = \arg_j \max p(\mathbf{X}|\lambda_j) \Rightarrow \mathcal{S} \leftarrow \mathcal{S}_i$



# Verificação

Modelagem Para todos os  $S_j \in \mathcal{S}$

# Verificação

Modelagem Para todos os  $\mathcal{S}_j \in \mathcal{S}$

- Extrair  $\mathbf{X}_k$  dos sinais  $\mathbf{Y}_k$  falados por cada  $\mathcal{S}_j$



# Verificação

Modelagem Para todos os  $\mathcal{S}_j \in \mathcal{S}$

- Extrair  $\mathbf{X}_k$  dos sinais  $\mathbf{Y}_k$  falados por cada  $\mathcal{S}_j$
- Treinar um  $\lambda_{bkg}$  através dos  $\mathbf{X}_k$  de todos os  $\mathcal{S}_j$

# Verificação

Modelagem Para todos os  $\mathcal{S}_j \in \mathcal{S}$

- Extrair  $\mathbf{X}_k$  dos sinais  $\mathbf{Y}_k$  falados por cada  $\mathcal{S}_j$
- Treinar um  $\lambda_{bkg}$  através dos  $\mathbf{X}_k$  de todos os  $\mathcal{S}_j$
- Modelar um  $\lambda_j$  para cada  $\mathcal{S}_j$

# Verificação

**Modelagem** Para todos os  $\mathcal{S}_j \in \mathcal{S}$

- Extrair  $\mathbf{X}_k$  dos sinais  $\mathbf{Y}_k$  falados por cada  $\mathcal{S}_j$
- Treinar um  $\lambda_{bkg}$  através dos  $\mathbf{X}_k$  de todos os  $\mathcal{S}_j$
- Modelar um  $\lambda_j$  para cada  $\mathcal{S}_j$

**Teste**  $\mathcal{S}$  diz ser  $\mathcal{S}_C \in \mathcal{S}$

# Verificação

**Modelagem** Para todos os  $\mathcal{S}_j \in \mathcal{S}$

- Extrair  $\mathbf{X}_k$  dos sinais  $\mathbf{Y}_k$  falados por cada  $\mathcal{S}_j$
- Treinar um  $\lambda_{bkg}$  através dos  $\mathbf{X}_k$  de todos os  $\mathcal{S}_j$
- Modelar um  $\lambda_j$  para cada  $\mathcal{S}_j$

**Teste**  $\mathcal{S}$  diz ser  $\mathcal{S}_C \in \mathcal{S}$

- Extrair  $\mathbf{X}$  do sinal  $\mathbf{Y}$  falado por  $\mathcal{S}_C$

# Verificação

**Modelagem** Para todos os  $\mathcal{S}_j \in \mathcal{S}$

- Extrair  $\mathbf{X}_k$  dos sinais  $\mathbf{Y}_k$  falados por cada  $\mathcal{S}_j$
- Treinar um  $\lambda_{bkg}$  através dos  $\mathbf{X}_k$  de todos os  $\mathcal{S}_j$
- Modelar um  $\lambda_j$  para cada  $\mathcal{S}_j$

**Teste**  $\mathcal{S}$  diz ser  $\mathcal{S}_C \in \mathcal{S}$

- Extrair  $\mathbf{X}$  do sinal  $\mathbf{Y}$  falado por  $\mathcal{S}_C$
- $\Lambda(\mathbf{X}) = \log p(\mathbf{X}|\lambda_C) - \log p(\mathbf{X}|\lambda_{bkg})$

# Verificação

**Modelagem** Para todos os  $\mathcal{S}_j \in \mathcal{S}$

- Extrair  $\mathbf{X}_k$  dos sinais  $\mathbf{Y}_k$  falados por cada  $\mathcal{S}_j$
- Treinar um  $\lambda_{bkg}$  através dos  $\mathbf{X}_k$  de todos os  $\mathcal{S}_j$
- Modelar um  $\lambda_j$  para cada  $\mathcal{S}_j$

**Teste**  $\mathcal{S}$  diz ser  $\mathcal{S}_C \in \mathcal{S}$

- Extrair  $\mathbf{X}$  do sinal  $\mathbf{Y}$  falado por  $\mathcal{S}_C$
- $\Lambda(\mathbf{X}) = \log p(\mathbf{X}|\lambda_C) - \log p(\mathbf{X}|\lambda_{bkg})$
- $\Lambda(\mathbf{X}) \geq \theta \implies \textit{aceita}$

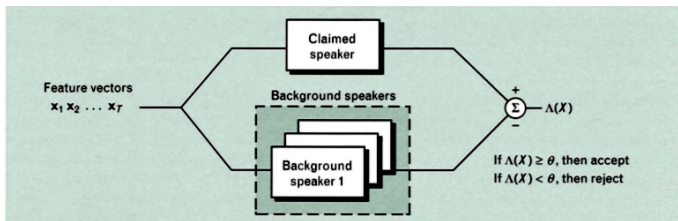
# Verificação

**Modelagem** Para todos os  $\mathcal{S}_j \in \mathcal{S}$

- Extrair  $\mathbf{X}_k$  dos sinais  $\mathbf{Y}_k$  falados por cada  $\mathcal{S}_j$
- Treinar um  $\lambda_{bkg}$  através dos  $\mathbf{X}_k$  de todos os  $\mathcal{S}_j$
- Modelar um  $\lambda_j$  para cada  $\mathcal{S}_j$

**Teste**  $\mathcal{S}$  diz ser  $\mathcal{S}_C \in \mathcal{S}$

- Extrair  $\mathbf{X}$  do sinal  $\mathbf{Y}$  falado por  $\mathcal{S}_C$
- $\Lambda(\mathbf{X}) = \log p(\mathbf{X}|\lambda_C) - \log p(\mathbf{X}|\lambda_{bkg})$
- $\Lambda(\mathbf{X}) \geq \theta \implies \text{aceita}$



# Conteúdo

- 1 Introdução
- 2 Sistemas de Reconhecimento de Locutor
- 3 Extração de Características**
- 4 Modelos de Mistura Gaussianas
- 5 Experimentos
- 6 Conclusão



# Características Ideais

- Natural e frequente na fala

# Características Ideais

- Natural e frequente na fala
- Facilmente mensurável

# Características Ideais

- Natural e frequente na fala
- Facilmente mensurável
- $\uparrow$  variação inter-locutor e  $\downarrow$  variação intra-locutor

# Características Ideais

- Natural e frequente na fala
- Facilmente mensurável
- $\uparrow$  variação inter-locutor e  $\downarrow$  variação intra-locutor
- Constante no tempo e não afetável pela saúde

# Características Ideais

- Natural e frequente na fala
- Facilmente mensurável
- $\uparrow$  variação inter-locutor e  $\downarrow$  variação intra-locutor
- Constante no tempo e não afetável pela saúde
- Robusta a ruído razoável e a transmissão

# Características Ideais

- Natural e frequente na fala
- Facilmente mensurável
- $\uparrow$  variação inter-locutor e  $\downarrow$  variação intra-locutor
- Constante no tempo e não afetável pela saúde
- Robusta a ruído razoável e a transmissão
- Difícil de ser produzido artificialmente

# Características Ideais

- Natural e frequente na fala
- Facilmente mensurável
- $\uparrow$  variação inter-locutor e  $\downarrow$  variação intra-locutor
- Constante no tempo e não afetável pela saúde
- Robusta a ruído razoável e a transmissão
- Difícil de ser produzido artificialmente
- Não ser facilmente modificável pelo locutor

# Mel-Frequency Cepstrum Coefficients

Simula a função da **cóclea**



# Mel-Frequency Cepstrum Coefficients

Simula a função da **cóclea**

Escala Mel Logaritmica

# Mel-Frequency Cepstrum Coefficients

Simula a função da **cóclea**

Escala Mel Logaritmica

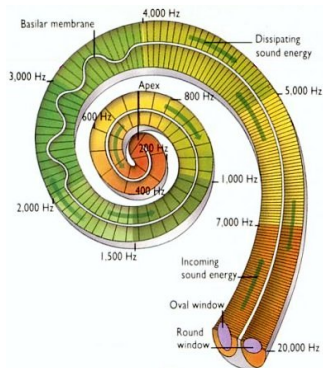
- $f_{mel} = 2595 \log_{10}(1 + \frac{f}{700})$

# Mel-Frequency Cepstrum Coefficients

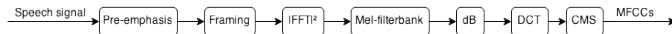
Simula a função da **cóclea**

Escala Mel Logaritmica

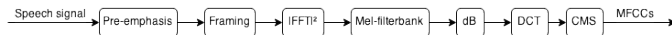
- $f_{mel} = 2595 \log_{10}\left(1 + \frac{f}{700}\right)$



# MFCC - Extração

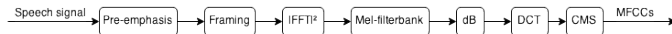


# MFCC - Extração



Pré-ênfase **Realça** as altas frequências (opcional)

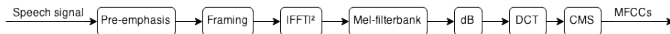
# MFCC - Extração



Pré-ênfase **Realça** as altas frequências (opcional)

- $s_{emph}[n] = s[n] - \alpha \cdot s[n - 1]$

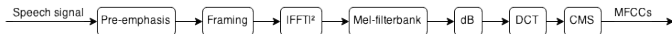
# MFCC - Extração



Pré-ênfase **Realça** as altas frequências (opcional)

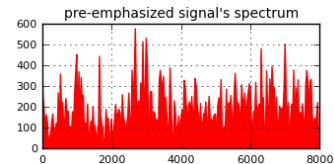
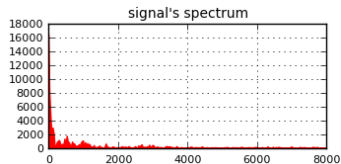
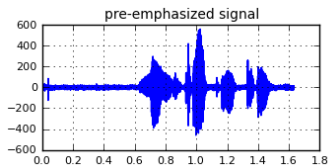
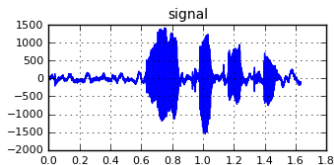
- $s_{emph}[n] = s[n] - \alpha \cdot s[n - 1]$
- $\alpha \in [0.95, 0.98]$

# MFCC - Extração



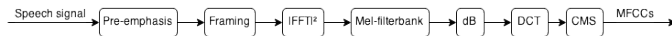
Pré-ênfase **Realça** as altas frequências (opcional)

- $s_{emph}[n] = s[n] - \alpha \cdot s[n - 1]$
- $\alpha \in [0.95, 0.98]$

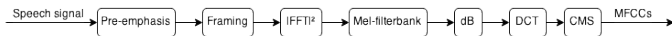




# MFCC - Extração

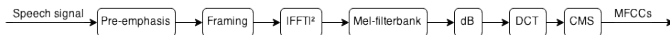


# MFCC - Extração



**Janelamento** Divide o sinal em janelas **superpostas**

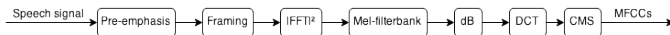
# MFCC - Extração



**Janelamento** Divide o sinal em janelas **superpostas**

- Largura de 20 milissegundos

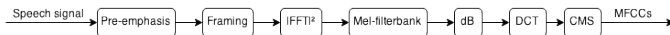
# MFCC - Extração



**Janelamento** Divide o sinal em janelas **superpostas**

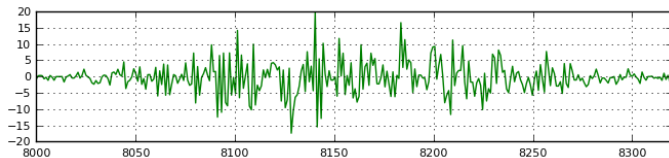
- Largura de 20 milissegundos
- Deslocamento de 10 milissegundos

# MFCC - Extração

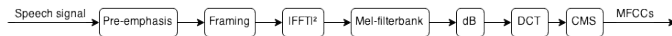


**Janelamento** Divide o sinal em janelas **superpostas**

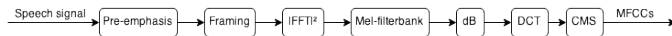
- Largura de 20 milissegundos
- Deslocamento de 10 milissegundos



# MFCC - Extração

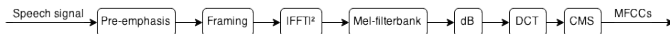


# MFCC - Extração

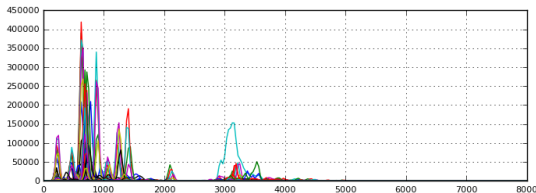
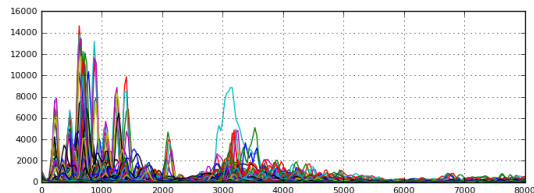


$|FFT|^2$  Calcula o **espectro de potência**

# MFCC - Extração

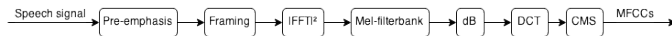


$|FFT|^2$  Calcula o **espectro de potência**

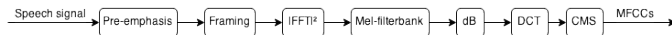




# MFCC - Extração

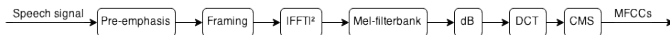


# MFCC - Extração

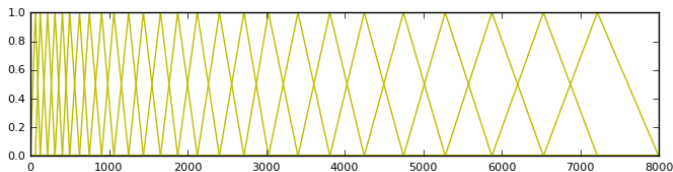


Filtros Espectro em Hz  $\Rightarrow$  espectro em **mels**

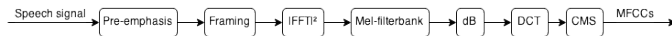
# MFCC - Extração



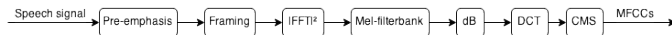
Filtros Espectro em Hz  $\Rightarrow$  espectro em **mels**



# MFCC - Extração

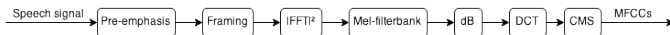


# MFCC - Extração

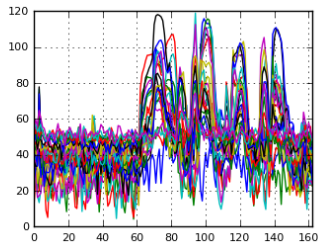
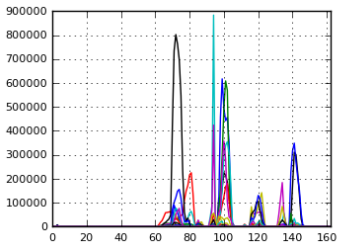


**dB** Calcula a **sonoridade**

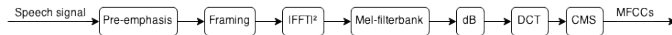
# MFCC - Extração



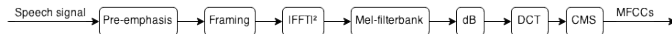
**dB** Calcula a **sonoridade**



# MFCC - Extração



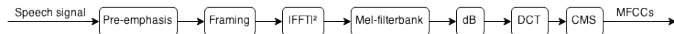
# MFCC - Extração



DCT Coeficientes espectrais  $\Rightarrow$  coeficientes **cepstrais**



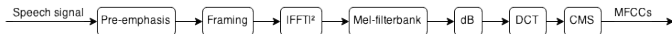
# MFCC - Extração



**DCT** Coeficientes espectrais  $\implies$  coeficientes **cepstrais**

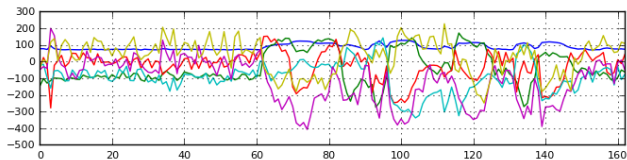
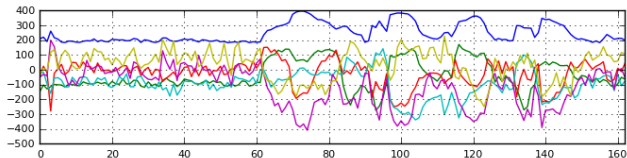
- $$c_n = \sum_{k=1}^K S_k \cdot \cos \left[ n \left( k - \frac{1}{2} \right) \frac{\pi}{K} \right], n = 1, 2, \dots, L$$

# MFCC - Extração

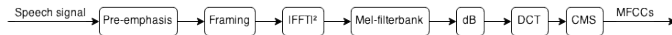


**DCT** Coeficientes espectrais  $\Rightarrow$  coeficientes **cepstrais**

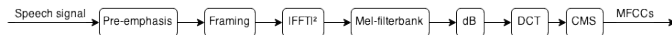
$$\bullet c_n = \sum_{k=1}^K S_k \cdot \cos \left[ n \left( k - \frac{1}{2} \right) \frac{\pi}{K} \right], n = 1, 2, \dots, L$$



# MFCC - Extração

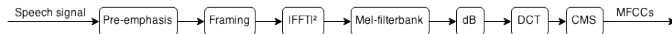


# MFCC - Extração



**CMS** Normaliza os MFCCs para reduzir perturbações

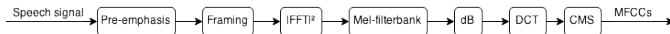
# MFCC - Extração



**CMS** Normaliza os MFCCs para reduzir perturbações

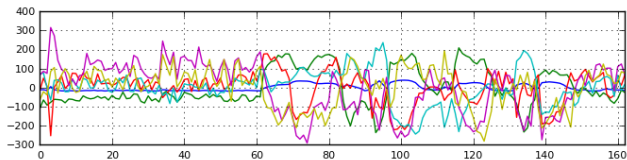
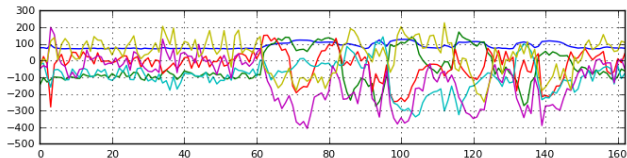
- $$c_n = c_n - \frac{1}{T} \sum_{t=1}^T c_{n,t}$$

# MFCC - Extração

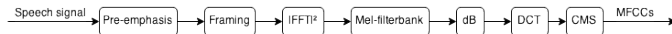


**CMS** Normaliza os MFCCs para reduzir perturbações

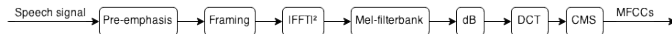
$$\bullet c_n = c_n - \frac{1}{T} \sum_{t=1}^T c_{n,t}$$



# MFCC - Extração



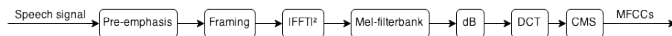
# MFCC - Extração



Δs Novos  $c_n$  **derivados** dos antigos  $c_n$  (opcional)



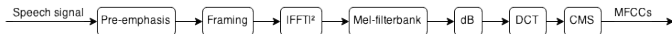
# MFCC - Extração



**Δs** Novos  $c_n$  **derivados** dos antigos  $c_n$  (opcional)

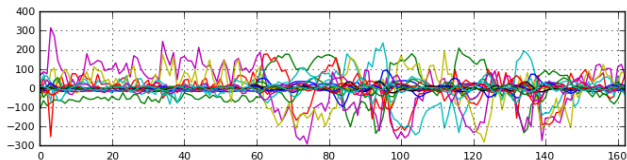
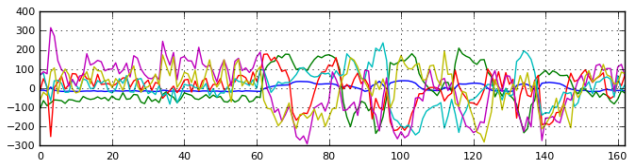
$$\bullet \Delta_t = \frac{\sum_{n=1}^N n(c_{t+n} - c_{t-n})}{2 \sum_{n=1}^N n^2}$$

# MFCC - Extração



**Δs** Novos  $c_n$  **derivados** dos antigos  $c_n$  (opcional)

$$\bullet \Delta_t = \frac{\sum_{n=1}^N n(c_{t+n} - c_{t-n})}{2 \sum_{n=1}^N n^2}$$



# Conteúdo

- 1 Introdução
- 2 Sistemas de Reconhecimento de Locutor
- 3 Extração de Características
- 4 Modelos de Mistura Gaussianas**
- 5 Experimentos
- 6 Conclusão

# Definição

$$\text{GMM } p(\mathbf{x}|\lambda) = \sum_{i=1}^M w_i p_i(\mathbf{x})$$

# Definição

GMM  $p(\mathbf{x}|\lambda) = \sum_{i=1}^M w_i p_i(\mathbf{x})$

Gaussiana  $p(\mathbf{x}) = \frac{1}{(2\pi)^{D/2} |\boldsymbol{\Sigma}|^{1/2}} e^{-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})' \boldsymbol{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu})}$

# Definição

**GMM**  $p(\mathbf{x}|\lambda) = \sum_{i=1}^M w_i p_i(\mathbf{x})$

**Gaussiana**  $p(\mathbf{x}) = \frac{1}{(2\pi)^{D/2} |\boldsymbol{\Sigma}|^{1/2}} e^{-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})' \boldsymbol{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu})}$

$$\lambda = \{(w_i, \boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i)\}, i = 1, \dots, M$$

# Definição

**GMM**  $p(\mathbf{x}|\lambda) = \sum_{i=1}^M w_i p_i(\mathbf{x})$

**Gaussiana**  $p(\mathbf{x}) = \frac{1}{(2\pi)^{D/2} |\mathbf{\Sigma}|^{1/2}} e^{-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})' \mathbf{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu})}$

$$\lambda = \{(w_i, \boldsymbol{\mu}_i, \mathbf{\Sigma}_i)\}, i = 1, \dots, M$$

$$\mathbf{\Sigma} \text{ diagonal} \implies \sigma^2$$

# Definição

**GMM**  $p(\mathbf{x}|\lambda) = \sum_{i=1}^M w_i p_i(\mathbf{x})$

**Gaussiana**  $p(\mathbf{x}) = \frac{1}{(2\pi)^{D/2} |\mathbf{\Sigma}|^{1/2}} e^{-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})' \mathbf{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu})}$

$$\lambda = \{(w_i, \boldsymbol{\mu}_i, \mathbf{\Sigma}_i)\}, i = 1, \dots, M$$

$$\mathbf{\Sigma} \text{ diagonal} \implies \sigma^2$$

Dada uma sequência  $\mathbf{X}$



# Definição

**GMM**  $p(\mathbf{x}|\lambda) = \sum_{i=1}^M w_i p_i(\mathbf{x})$

**Gaussiana**  $p(\mathbf{x}) = \frac{1}{(2\pi)^{D/2} |\mathbf{\Sigma}|^{1/2}} e^{-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})' \mathbf{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu})}$

$$\lambda = \{(w_i, \boldsymbol{\mu}_i, \mathbf{\Sigma}_i)\}, i = 1, \dots, M$$

$$\mathbf{\Sigma} \text{ diagonal} \implies \sigma^2$$

Dada uma sequência  $\mathbf{X}$

- $p(\mathbf{X}|\lambda) = \prod_{t=1}^T p(\mathbf{x}_t|\lambda).$

# Definição

**GMM**  $p(\mathbf{x}|\lambda) = \sum_{i=1}^M w_i p_i(\mathbf{x})$

**Gaussiana**  $p(\mathbf{x}) = \frac{1}{(2\pi)^{D/2} |\mathbf{\Sigma}|^{1/2}} e^{-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})' \mathbf{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu})}$

$$\lambda = \{(w_i, \boldsymbol{\mu}_i, \mathbf{\Sigma}_i)\}, i = 1, \dots, M$$

$$\mathbf{\Sigma} \text{ diagonal} \implies \sigma^2$$

Dada uma sequência  $\mathbf{X}$

- $p(\mathbf{X}|\lambda) = \prod_{t=1}^T p(\mathbf{x}_t|\lambda)$ .
- Função não linear de  $\lambda$

# Definição

**GMM**  $p(\mathbf{x}|\lambda) = \sum_{i=1}^M w_i p_i(\mathbf{x})$

**Gaussiana**  $p(\mathbf{x}) = \frac{1}{(2\pi)^{D/2} |\mathbf{\Sigma}|^{1/2}} e^{-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})' \mathbf{\Sigma}^{-1}(\mathbf{x}-\boldsymbol{\mu})}$

$$\lambda = \{(w_i, \boldsymbol{\mu}_i, \mathbf{\Sigma}_i)\}, i = 1, \dots, M$$

$$\mathbf{\Sigma} \text{ diagonal} \implies \sigma^2$$

Dada uma sequência  $\mathbf{X}$

- $p(\mathbf{X}|\lambda) = \prod_{t=1}^T p(\mathbf{x}_t|\lambda)$ .
- Função não linear de  $\lambda$
- Estimar com o Expectation-Maximization (EM)

# Expectation-Maximization

Estimar  $\lambda^{(k+1)}$  a partir de  $\lambda^k$

# Expectation-Maximization

Estimar  $\lambda^{(k+1)}$  a partir de  $\lambda^k$

Obedecer  $p(\mathbf{X}|\lambda^{(k+1)}) \geq p(\mathbf{X}|\lambda^{(k)})$

# Expectation-Maximization

Estimar  $\lambda^{(k+1)}$  a partir de  $\lambda^k$

Obedecer  $p(\mathbf{X}|\lambda^{(k+1)}) \geq p(\mathbf{X}|\lambda^{(k)})$

Calcular *E-Step* e *M-Step* para cada  $k$  até convergir

# Expectation-Maximization

Estimar  $\lambda^{(k+1)}$  a partir de  $\lambda^k$

Obedecer  $p(\mathbf{X}|\lambda^{(k+1)}) \geq p(\mathbf{X}|\lambda^{(k)})$

Calcular *E-Step* e *M-Step* para cada  $k$  até convergir

E-Step 
$$P(i|\mathbf{x}_t) = \frac{w_i p_i(\mathbf{x}_t)}{\sum_{k=1}^M w_k p_k(\mathbf{x}_t)}$$

# Expectation-Maximization

Estimar  $\lambda^{(k+1)}$  a partir de  $\lambda^k$

Obedecer  $p(\mathbf{X}|\lambda^{(k+1)}) \geq p(\mathbf{X}|\lambda^{(k)})$

Calcular *E-Step* e *M-Step* para cada  $k$  até convergir

**E-Step** 
$$P(i|\mathbf{x}_t) = \frac{w_i p_i(\mathbf{x}_t)}{\sum_{k=1}^M w_k p_k(\mathbf{x}_t)}$$

**M-Step** Adaptar os parâmetros



# Expectation-Maximization

Estimar  $\lambda^{(k+1)}$  a partir de  $\lambda^k$

Obedecer  $p(\mathbf{X}|\lambda^{(k+1)}) \geq p(\mathbf{X}|\lambda^{(k)})$

Calcular *E-Step* e *M-Step* para cada  $k$  até convergir

**E-Step**  $P(i|\mathbf{x}_t) = \frac{w_i p_i(\mathbf{x}_t)}{\sum_{k=1}^M w_k p_k(\mathbf{x}_t)}$

**M-Step** Adaptar os parâmetros

**Pesos**  $\bar{w}_i = \frac{1}{T} \sum_{t=1}^T P(i|\mathbf{x}_t, \lambda)$

# Expectation-Maximization

Estimar  $\lambda^{(k+1)}$  a partir de  $\lambda^k$

Obedecer  $p(\mathbf{X}|\lambda^{(k+1)}) \geq p(\mathbf{X}|\lambda^{(k)})$

Calcular *E-Step* e *M-Step* para cada  $k$  até convergir

**E-Step**  $P(i|\mathbf{x}_t) = \frac{w_i p_i(\mathbf{x}_t)}{\sum_{k=1}^M w_k p_k(\mathbf{x}_t)}$

**M-Step** Adaptar os parâmetros

**Pesos**  $\bar{w}_i = \frac{1}{T} \sum_{t=1}^T P(i|\mathbf{x}_t, \lambda)$

**Médias**  $\bar{\mu}_i = \frac{\sum_{t=1}^T P(i|\mathbf{x}_t, \lambda) \mathbf{x}_t}{\sum_{t=1}^T P(i|\mathbf{x}_t, \lambda)}$

# Expectation-Maximization

Estimar  $\lambda^{(k+1)}$  a partir de  $\lambda^k$

Obedecer  $p(\mathbf{X}|\lambda^{(k+1)}) \geq p(\mathbf{X}|\lambda^{(k)})$

Calcular *E-Step* e *M-Step* para cada  $k$  até convergir

**E-Step**  $P(i|\mathbf{x}_t) = \frac{w_i p_i(\mathbf{x}_t)}{\sum_{k=1}^M w_k p_k(\mathbf{x}_t)}$

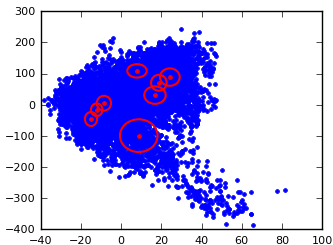
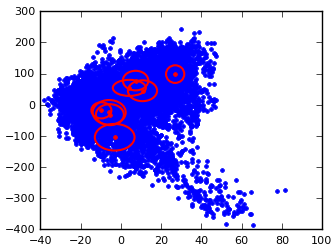
**M-Step** Adaptar os parâmetros

**Pesos**  $\bar{w}_i = \frac{1}{T} \sum_{t=1}^T P(i|\mathbf{x}_t, \lambda)$

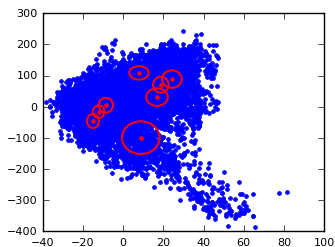
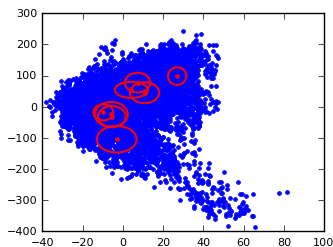
**Médias**  $\bar{\mu}_i = \frac{\sum_{t=1}^T P(i|\mathbf{x}_t, \lambda) \mathbf{x}_t}{\sum_{t=1}^T P(i|\mathbf{x}_t, \lambda)}$

**Variâncias**  $\bar{\sigma}_i^2 = \frac{\sum_{t=1}^T P(i|\mathbf{x}_t, \lambda) \mathbf{x}_t^2}{\sum_{t=1}^T P(i|\mathbf{x}_t, \lambda)} - \bar{\mu}_i^2$

# Expectation-Maximization

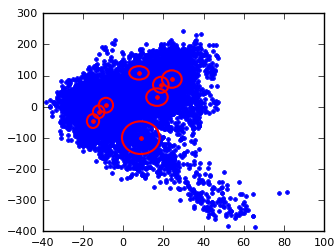
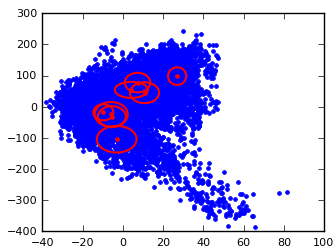


# Expectation-Maximization



$$M = 8$$

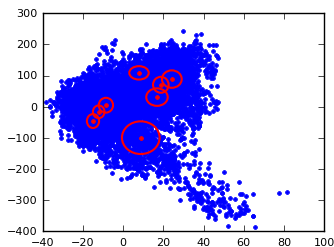
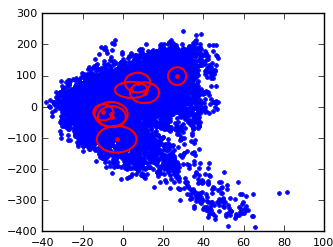
# Expectation-Maximization



$$M = 8$$

$$\Delta = 0$$

# Expectation-Maximization

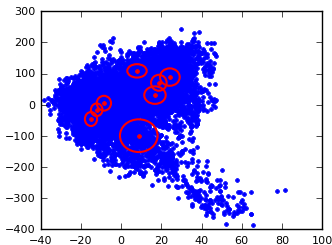
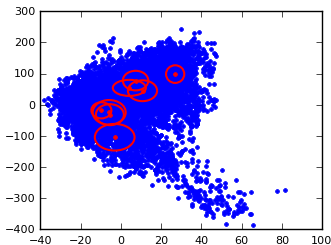


$$M = 8$$

$$\Delta = 0$$

Inicialização *k-means* com 1 iteração

# Expectation-Maximization



$$M = 8$$

$$\Delta = 0$$

Inicialização *k-means* com 1 iteração

Limiar  $10^{-3}$



# Universal Background Model

Utiliza locuções de todos os locutores registrados

# Universal Background Model

Utiliza locuções de todos os locutores registrados

Realça características comuns

# Universal Background Model

Utiliza locuções de todos os locutores registrados

Realça características comuns

**X** específico  $\implies \uparrow \Lambda(\mathbf{X})$

# Universal Background Model

Utiliza locuções de todos os locutores registrados

Realça características comuns

$\mathbf{X}$  específico  $\implies \uparrow \Lambda(\mathbf{X})$

Escolhido o tipo ( $b$ )

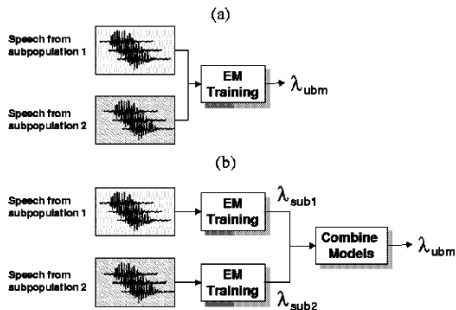
# Universal Background Model

Utiliza locuções de todos os locutores registrados

Realça características comuns

$\mathbf{X}$  específico  $\implies \uparrow \Lambda(\mathbf{X})$

Escolhido o tipo ( $b$ )



# Adapted Gaussian Mixture Model

Adaptação  $\lambda_{bkg}$  treinado  $\implies \lambda_j$  para cada  $\mathcal{S}_j$

# Adapted Gaussian Mixture Model

Adaptação  $\lambda_{bkg}$  treinado  $\implies \lambda_j$  para cada  $\mathcal{S}_j$   
Modelagem mais rápida que EM

# Adapted Gaussian Mixture Model

Adaptação  $\lambda_{bkg}$  treinado  $\implies \lambda_j$  para cada  $\mathcal{S}_j$

Modelagem mais rápida que EM

Composto de *M-Step* e *MAP-Step*



# Adapted Gaussian Mixture Model

Adaptação  $\lambda_{bkg}$  treinado  $\implies \lambda_j$  para cada  $\mathcal{S}_j$

Modelagem mais rápida que EM

Composto de *M-Step* e *MAP-Step*

E-Step Semelhante ao *E-Step* do EM

# Adapted Gaussian Mixture Model

Adaptação  $\lambda_{bkg}$  treinado  $\implies \lambda_j$  para cada  $\mathcal{S}_j$

Modelagem mais rápida que EM

Composto de *M-Step* e *MAP-Step*

E-Step Semelhante ao *E-Step* do EM

- $n_i = \sum_{t=1}^T P(i|\mathbf{x}_t)$

# Adapted Gaussian Mixture Model

Adaptação  $\lambda_{bkg}$  treinado  $\implies \lambda_j$  para cada  $\mathcal{S}_j$

Modelagem mais rápida que EM

Composto de *M-Step* e *MAP-Step*

E-Step Semelhante ao *E-Step* do EM

- $n_i = \sum_{t=1}^T P(i|\mathbf{x}_t)$
- $E_i(\mathbf{x}) = \frac{1}{n_i} \sum_{t=1}^T P(i|\mathbf{x}_t) \mathbf{x}_t$

# Adapted Gaussian Mixture Model

Adaptação  $\lambda_{bkg}$  treinado  $\implies \lambda_j$  para cada  $\mathcal{S}_j$

Modelagem mais rápida que EM

Composto de *M-Step* e *MAP-Step*

E-Step Semelhante ao *E-Step* do EM

- $n_i = \sum_{t=1}^T P(i|\mathbf{x}_t)$
- $E_i(\mathbf{x}) = \frac{1}{n_i} \sum_{t=1}^T P(i|\mathbf{x}_t) \mathbf{x}_t$
- $E_i(\mathbf{x}^2) = \frac{1}{n_i} \sum_{t=1}^T P(i|\mathbf{x}_t) \mathbf{x}_t^2$

# Adapted Gaussian Mixture Model

**Adaptação**  $\lambda_{bkg}$  treinado  $\implies \lambda_j$  para cada  $\mathcal{S}_j$

Modelagem mais rápida que EM

Composto de *M-Step* e *MAP-Step*

**E-Step** Semelhante ao *E-Step* do EM

- $n_i = \sum_{t=1}^T P(i|\mathbf{x}_t)$
- $E_i(\mathbf{x}) = \frac{1}{n_i} \sum_{t=1}^T P(i|\mathbf{x}_t) \mathbf{x}_t$
- $E_i(\mathbf{x}^2) = \frac{1}{n_i} \sum_{t=1}^T P(i|\mathbf{x}_t) \mathbf{x}_t^2$

**MAP-Step** Adapta os parâmetros

# Adapted Gaussian Mixture Model

**Adaptação**  $\lambda_{bkg}$  treinado  $\implies \lambda_j$  para cada  $\mathcal{S}_j$

Modelagem mais rápida que EM

Composto de *M-Step* e *MAP-Step*

**E-Step** Semelhante ao *E-Step* do EM

- $n_i = \sum_{t=1}^T P(i|\mathbf{x}_t)$
- $E_i(\mathbf{x}) = \frac{1}{n_i} \sum_{t=1}^T P(i|\mathbf{x}_t) \mathbf{x}_t$
- $E_i(\mathbf{x}^2) = \frac{1}{n_i} \sum_{t=1}^T P(i|\mathbf{x}_t) \mathbf{x}_t^2$

**MAP-Step** Adapta os parâmetros

**Pesos**  $\hat{w}_i = [\alpha_i n_i / T + (1 - \alpha_i) w_i] \gamma$

# Adapted Gaussian Mixture Model

**Adaptação**  $\lambda_{bkg}$  treinado  $\implies \lambda_j$  para cada  $\mathcal{S}_j$

Modelagem mais rápida que EM

Composto de *M-Step* e *MAP-Step*

**E-Step** Semelhante ao *E-Step* do EM

- $n_i = \sum_{t=1}^T P(i|\mathbf{x}_t)$
- $E_i(\mathbf{x}) = \frac{1}{n_i} \sum_{t=1}^T P(i|\mathbf{x}_t) \mathbf{x}_t$
- $E_i(\mathbf{x}^2) = \frac{1}{n_i} \sum_{t=1}^T P(i|\mathbf{x}_t) \mathbf{x}_t^2$

**MAP-Step** Adapta os parâmetros

**Pesos**  $\hat{w}_i = [\alpha_i n_i / T + (1 - \alpha_i) w_i] \gamma$

**Médias**  $\hat{\mu}_i = \alpha_i E_i(\mathbf{x}) + (1 - \alpha_i) \mu_i$

# Adapted Gaussian Mixture Model

**Adaptação**  $\lambda_{bkg}$  treinado  $\implies \lambda_j$  para cada  $\mathcal{S}_j$

Modelagem mais rápida que EM

Composto de *M-Step* e *MAP-Step*

**E-Step** Semelhante ao *E-Step* do EM

- $n_i = \sum_{t=1}^T P(i|\mathbf{x}_t)$
- $E_i(\mathbf{x}) = \frac{1}{n_i} \sum_{t=1}^T P(i|\mathbf{x}_t) \mathbf{x}_t$
- $E_i(\mathbf{x}^2) = \frac{1}{n_i} \sum_{t=1}^T P(i|\mathbf{x}_t) \mathbf{x}_t^2$

**MAP-Step** Adapta os parâmetros

**Pesos**  $\hat{w}_i = [\alpha_i n_i / T + (1 - \alpha_i) w_i] \gamma$

**Médias**  $\hat{\mu}_i = \alpha_i E_i(\mathbf{x}) + (1 - \alpha_i) \mu_i$

**Variâncias**  $\hat{\sigma}_i^2 = \alpha_i E_i(\mathbf{x}^2) + (1 - \alpha_i)(\sigma_i^2 + \mu_i^2) - \hat{\mu}_i^2$



# Adapted Gaussian Mixture Model

$\gamma$  normaliza os pesos

# Adapted Gaussian Mixture Model

$\gamma$  normaliza os pesos

Coeficiente  $\alpha_i = \frac{n_i}{n_i + r}$

# Adapted Gaussian Mixture Model

$\gamma$  normaliza os pesos

Coeficiente  $\alpha_i = \frac{n_i}{n_i + r}$

- $\alpha_i \rightarrow 0 \implies$  manter os antigos parâmetros

# Adapted Gaussian Mixture Model

$\gamma$  normaliza os pesos

Coeficiente  $\alpha_i = \frac{n_i}{n_i + r}$

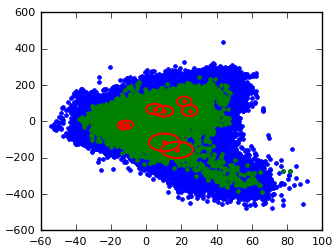
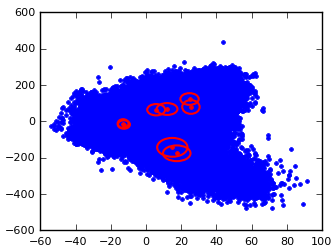
- $\alpha_i \rightarrow 0 \implies$  manter os antigos parâmetros
- $\alpha_i \rightarrow 1 \implies$  adaptar para os novos parâmetros

# Adapted Gaussian Mixture Model

$\gamma$  normaliza os pesos

Coeficiente  $\alpha_i = \frac{n_i}{n_i + r}$

- $\alpha_i \rightarrow 0 \implies$  manter os antigos parâmetros
- $\alpha_i \rightarrow 1 \implies$  adaptar para os novos parâmetros

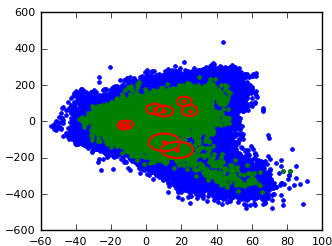
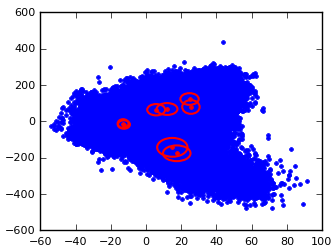


# Adapted Gaussian Mixture Model

$\gamma$  normaliza os pesos

Coeficiente  $\alpha_i = \frac{n_i}{n_i + r}$

- $\alpha_i \rightarrow 0 \implies$  manter os antigos parâmetros
- $\alpha_i \rightarrow 1 \implies$  adaptar para os novos parâmetros



Pesos, médias e variâncias adaptados

# Fractional Gaussian Mixture Model

GMM com  $\Sigma$  fracionário

# Fractional Gaussian Mixture Model

GMM com  $\Sigma$  fracionário

- $\sigma^2 = E[(X^r - \mu^r)^2]$



# Fractional Gaussian Mixture Model

GMM com  $\Sigma$  fracionário

- $\sigma^2 = E[(X^r - \mu^r)^2]$
- $\bar{\sigma}_i^2 = \frac{\sum_{t=1}^T P(i|\mathbf{x}_t, \lambda) (\mathbf{x}_t^r - \bar{\mu}_i^r)^2}{\sum_{t=1}^T P(i|\mathbf{x}_t, \lambda)}$

# Fractional Gaussian Mixture Model

GMM com  $\Sigma$  fracionário

- $\sigma^2 = E[(X^r - \mu^r)^2]$
- $\bar{\sigma}_i^2 = \frac{\sum_{t=1}^T P(i|\mathbf{x}_t, \lambda) (\mathbf{x}_t^r - \bar{\mu}_i^r)^2}{\sum_{t=1}^T P(i|\mathbf{x}_t, \lambda)}$

Problema  $\mathbb{C}$

# Fractional Gaussian Mixture Model

GMM com  $\Sigma$  fracionário

- $\sigma^2 = E[(X^r - \mu^r)^2]$
- $\bar{\sigma}_i^2 = \frac{\sum_{t=1}^T P(i|\mathbf{x}_t, \lambda) (\mathbf{x}_t^r - \bar{\mu}_i^r)^2}{\sum_{t=1}^T P(i|\mathbf{x}_t, \lambda)}$

Problema  $\mathbb{C}$

Antes  $c_n = c_n + (1 - \min_t c_{n,t})$

# Fractional Gaussian Mixture Model

GMM com  $\Sigma$  fracionário

- $\sigma^2 = E[(X^r - \mu^r)^2]$
- $\bar{\sigma}_i^2 = \frac{\sum_{t=1}^T P(i|\mathbf{x}_t, \lambda) (\mathbf{x}_t^r - \bar{\mu}_i^r)^2}{\sum_{t=1}^T P(i|\mathbf{x}_t, \lambda)}$

Problema  $\mathbb{C}$

Antes  $c_n = c_n + (1 - \min_t c_{n,t})$

Depois  $\mu_n = \mu_n - (1 - \min_t c_{n,t})$

# Fractional Gaussian Mixture Model

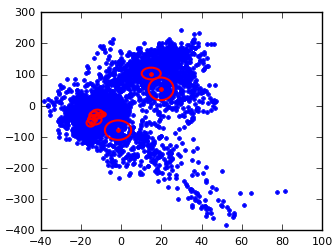
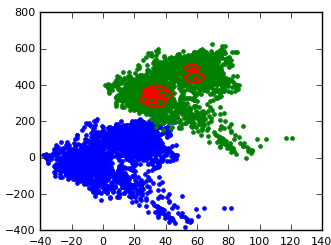
GMM com  $\Sigma$  fracionário

- $\sigma^2 = E[(X^r - \mu^r)^2]$
- $\bar{\sigma}_i^2 = \frac{\sum_{t=1}^T P(i|\mathbf{x}_t, \lambda) (\mathbf{x}_t^r - \bar{\mu}_i^r)^2}{\sum_{t=1}^T P(i|\mathbf{x}_t, \lambda)}$

Problema  $\mathbb{C}$

Antes  $c_n = c_n + (1 - \min_t c_{n,t})$

Depois  $\mu_n = \mu_n - (1 - \min_t c_{n,t})$



# Conteúdo

- 1 Introdução
- 2 Sistemas de Reconhecimento de Locutor
- 3 Extração de Características
- 4 Modelos de Mistura Gaussianas
- 5 Experimentos**
- 6 Conclusão

# Experimentos

# Conteúdo

- 1 Introdução
- 2 Sistemas de Reconhecimento de Locutor
- 3 Extração de Características
- 4 Modelos de Mistura Gaussianas
- 5 Experimentos
- 6 Conclusão**



# Conclusão

# Obrigado