UNIVERSIDAD AUTÓNOMA DE MADRID

ESCUELA POLITÉCNICA SUPERIOR



PROYECTO FIN DE CARRERA

# EXTRACCIÓN DE INFORMACIÓN DE SEÑALES DE VOZ PARA EL AGRUPAMIENTO POR LOCUTORES DE LOCUCIONES ANÓNIMAS

Ingeniería de Telecomunicación

Iván Gómez Piris

Junio 2014

# EXTRACCIÓN DE INFORMACIÓN DE SEÑALES DE VOZ PARA EL AGRUPAMIENTO POR LOCUTORES DE LOCUCIONES ANÓNIMAS

AUTOR: Iván Gómez Piris
TUTOR: Joaquín González Rodríguez

Área de Tratamiento de Voz y Señales
Dpto. de Tecnología Electrónica y de las Comunicaciones
Escuela Politécnica Superior
Universidad Autónoma de Madrid
Junio 2014

# Resumen

En este proyecto de fin de carrera se presenta una solución, mediante agrupamiento de locutores, a la situación realista de desconocimiento de etiquetas de identidad de locutor para el entrenamiento de técnicas de compensación de variabilidad como LDA (*Linear Discriminant Analysis*) y PLDA (*Probabilistic Linear Discriminant Analysis),* tecnologías en el estado del arte del reconocimiento de locutor.

En primer lugar, se presenta el estado del arte de los sistemas de reconocimiento de locutor, así como diferentes herramientas necesarias para la evaluación del rendimiento del sistema, como son el Equal Error Rate (EER,%), el mínimo de la Función de Coste de Detección (minDCF) y el mínimo Logarithmic Likelihood Ratio Cost ($minC_{llr}$).

En segundo lugar, se presenta una descripción del algoritmo de agrupamiento que se ha implementado (AHC, *Agglomerative Hierarchical Clustering*). El algoritmo AHC presenta configuraciones en función de parámetros como la métrica de distancia a utilizar, el método de *linkage* y el criterio de parada. Además, se describen diferentes técnicas de validación de las soluciones de agrupamiento obtenidas mediante el algoritmo.

A continuación, se presenta una descripción del sistema de reconocimiento de locutor, de las soluciones de agrupamiento y de las medidas de rendimiento implementadas. Partiendo de los archivos de audio, se detalla desde cómo se han extraído los vectores de características de cada locución (MFCC) hasta la obtención de los *i-vectors*, modelo de locutor utilizado en este proyecto y observación sobre la que se realizará el agrupamiento.

Posteriormente, se llevan a cabo diferentes experimentos con el fin de obtener resultados que permitan evaluar el impacto de los diferentes agrupamientos de locutores realizados cuando se desconocen las etiquetas de identidad de locutor y si es posible alcanzar rendimientos similares a los que se obtendrían haciendo uso de ellas. Los experimentos se llevarán a cabo siguiendo el protocolo de evaluaciones NIST (National Institute of Standards and Techonology).

Finalmente, se presentan las conclusiones extraídas a lo largo del proyecto junto con las propuestas del trabajo futuro.

# Palabras claves

Reconocimiento de locutor, MFCC, GMM-UBM, *Factor Analysis, i-vector, AHC, LDA, PLDA*.

# Abstract

In this M.Sc. Thesis we present a solution, performed by speaker clustering, to the situation of lack of speaker identity labels for training variability compensation state-of-the-art techniques in the speaker recognition field, such as LDA (Linear Discriminant Analysis) or PLDA (Probabilistic Linear Discriminant Analysis).

First of all, we present the state of the art of the speaker recognition field as well as different tools needed for evaluating the performance of a speaker recognition system, such as the Equal Error Rate (EER,%), the minimum of the cost function Detection (minDCF) and the minimum Cost Logarithmic Likelihood Ratio (minCllr).

Secondly, we report a description of the clustering algorithm that has been implemented (AHC, agglomerative Hierarchical Clustering). This algorithm presents configurations depending on parameters such as the distance metric, the method of linkage and the stopping criterion. Furthermore, different clustering-validation techniques are described.

Then, we report the system description -taking into account the speaker recognition system, the clustering algorithm and the performance measures- that has been implemented. We detail how the features vectors (MFCC) have been extracted from each utterance up to obtain i-vectors, the speaker-models used in this project and segments over the clustering has been performed.

Subsequently, we carry out different experiments in order to obtain results that allow us to evaluate the impact of different speaker-cluster structures when the speaker identity labels are unknown and as far as possible achieve similar yields to those obtained when the speaker identity labels are known and used. The experiments were performed following the protocol of NIST evaluation plans (National Institute of Standards and Techonology).

Finally, conclusions are drawn, and future lines of work are proposed.

# Keywords

Speaker recognition, MFCC, GMM-UBM, *Factor Analysis, i-vector, AHC, LDA, PLDA*.

A mis padres, a mi hermana y a mi novia.

*La calidad nunca es un accidente; siempre es el resultado de un esfuerzo de la inteligencia.*

John Ruskin

# Agradecimientos

Con la realización de este proyecto finaliza una etapa muy importante de mi vida. No ha sido un camino fácil y me he enfrentado a muchos retos pero, esfuerzo y sacrificio, los he ido superando poco a poco.

En primer lugar, me gustaría dar las gracias a mi tutor, Joaquín González Rodríguez, por haberme dado la oportunidad de realizar el presente proyecto en el Área de Tratamiento de Voz y Señal (ATVS), así como por su esfuerzo y su ayuda.

En segundo lugar, mi sincero agradecimiento a todas las personas del ATVS: Javier Franco, Alicia, Alfredo, Ricardo, Junchen Xu, Fer, Daniel Ramos, Javier González, Marta, Esther, Pedro, Rubén Vera, Álvaro, Ram y Ruifang. He sido muy afortunado al haber tenido la oportunidad de trabajar, en mayor o menor medida, con todos vosotros. Me habéis enseñado muchísimo y me habéis hecho sentir como en casa. Muchas gracias a todos.

Si bien ya lo he mencionado, me gustaría recalcar la importancia de una persona, compañera de fatigas en todos estos años, Alfredo (Alf/Rubal). Después de tantas historias y tantas batallitas con diferentes asignaturas, ¡ya terminamos tío! ¡Bye bye EPS!

Agradecer también a mis compañeros de carrera todos los momentos por los que hemos pasado. He hecho muchas amistades durante toda esta etapa, hemos luchado juntos y hemos vencido juntos.

También quiero agradecer a toda mi familia por el apoyo que me han dado durante todo este tiempo, con especial dedicación a mis padres y a mi hermana. Me habéis apoyado en todo momento, ayudándome siempre que os ha sido posible.

Por último, pero no menos importante, agradecer a mi novia Bea su apoyo. Hemos tenido épocas difíciles debido a la dedicación que ha requerido esta carrera, pero has sabido y querido pasarlos a mi lado. Has demostrado una fortaleza grandiosa en momentos en los que yo necesitaba una referencia. Muchas gracias por todo tu apoyo.

Gracias a todos.

Iván Gómez Piris
Junio 2014


Este proyecto ha sido realizado en el Área de Tratamiento de Voz y Señales (ATVS) en la Escuela Politécnica Superior de la UAM. El proyecto ha sido financiado parcialmente por el Ministerio de Economía y Competitividad a través del proyecto TEC 2012-37585-C02-01.

# Índice de contenidos

# Índice de figuras

# Índice de tablas

# Glosario de términos

**ASR** *Automatic Speech Recognition* (reconocimiento automático de habla).

**AHC** *Agglomerative Hierarchical Clustering* (agrupamiento jerárquico aglomerativo).

**CMN** *Cepstral Mean Normalization* (normalización de la media cepstral). Técnica de compensación de los efectos del canal de transmisión sobre la señal de voz que se aplica en el dominio de los coeficientes cepstrales.

**Cluster** Grupo o conjunto de elementos que guardan alguna relación entre sí.

**CSS** *Cosine similarity score* (puntuación similitud coseno).

**DCF** *Detection Cost Function* (función de coste de detección). Función definida para la evaluación del rendimiento de los sistemas de reconocimiento de locutor.

**DCT** *Discrete Cosine Transform* (transformada discreta del coseno). Función de transformación basada en la DFT pero que utiliza únicamente números reales.

**DFT** *Discrete Fourier Transform* (transformada discreta de Fourier). Función de transformación ampliamente empleada en tratamiento de señales y campos afines para analizar las frecuencias presentes en una señal muestreada.

**DET** *Detection Error Trade-off* (compensación por error de detección). La curva DET se emplea para representar de forma gráfica el rendimiento de un sistema de reconocimiento biométrico para los distintos puntos de trabajo posibles. Se obtiene mediante un cambio de escala en los ejes de la curva ROC.

**EER** *Equal Error Rate* (tasa de error igual). Tasa de error, en los sistemas de reconocimiento biométrico, en que se igualan las tasas de falsa aceptación y falso rechazo.

**FA** *Factor Analysis* (análisis de factores). Técnica empleada en reconocimiento de locutor para el modelado explícito de la variabilidad inter-sesión en el entrenamiento de los modelos de locutor.

**FAR** *False Acceptance Rate* (tasa de falsa aceptación). Porcentaje de errores, respecto al total de comparaciones realizadas, en los que se considera

que el rasgo de test se corresponde con el patrón de referencia cuando en realidad corresponde a una identidad distinta.

**FFT** *Fast Fourier Transform* (transformada rápida de Fourier). Algoritmo para la implementación rápida de la DFT.

**FRR** *False Rejection Rate* (tasa de falso rechazo). Porcentaje de errores, respecto al total de comparaciones realizadas, en los que se considera que el rasgo de test se corresponde con el patrón de referencia cuando en realidad corresponde a una identidad distinta.

**GMM** *Gaussian Mixture Model* (modelo de mezcla de gaussianas). Técnica para el modelado de la identidad de un sujeto por medio del ajuste de un conjunto de gaussianas multivariadas a su distribución de características.

**GMM-UBM** Técnica de modelado basado en GMM pero entrenando un modelo *universal* UBM para la posterior adaptación del modelo de locutor vía adaptación MAP.

**JFA** Joint Factor Analysis. Técnica de compensación de variabilidad empleada para el modelado de la variabilidad intra-locutor como la debida al canal. Se aplica sobre supervectores.

**LDA** *Linear Discriminant Analysis.* Técnica de compensación de inter-variabilidad de locutor.

**Linkage** Metodología a seguir en algoritmos de *clustering* basados en medidas de distancia o similitud cuando se desean unir dos o más *clusters* que contienen dos o más segmentos.

**Locución** En el ámbito del reconocimiento de locutor, se emplea con el significado de "señal de audio utilizada como rasgo biométrico para la obtención de un patrón de referencia o como rasgo de test en el proceso de identificación o verificación".

**MAP** *Maximum A Posteriori*. Técnica empleada para la adaptación de modelos de locutor a partir de un UBM en los sistemas basados en GMM.

**MFCC** *Mel Frequency Cepstral Coefficients* (coeficientes cepstrales en escala de frecuencias Mel). Coeficientes para la representación del habla basados en la percepción auditiva humana.

**NIST** *National Institute of Standards and Technology* (Instituto Nacional de Estándares y Tecnología de los Estados Unidos de América).

**PLDA** *Probabilistic Linear Discriminant Analysis.* Técnica de compensación de variabilidad empleada para el modelado de la variabilidad intra-locutor y de la variabilidad inter-locutor. Se aplica sobre i-vectors.

**RASTA** *RelAtiveSpecTrAl* (espectral relativo). El filtrado RASTA es una técnica de compensación de los efectos del canal de transmisión sobre la señal de voz que se aplica en el dominio de los coeficientes cepstrales.

**Score** Puntuación obtenida por un sistema de reconocimiento biométrico en la comparación entre un patrón de referencia y un rasgo biométrico de test.

**Scoring** Proceso de obtención de scores.

**SRE** *Speaker Recognition Evaluation* (evaluación de reconocimiento de locutor). Serie de evaluaciones organizadas por el NIST para fomentar el avance en las técnicas de reconocimiento de locutor.

**Trial** Juicio o comparación entre un rasgo de test y un patrón de referencia.

**Trial de impostor** Comparación entre un patrón de referencia y un rasgo de test cuya identidad NO se corresponde con la de aquél.

**Trial genuino** Comparación entre un patrón de referencia y un rasgo de test cuya identidad SÍ se corresponde con la de aquél.

**UBM** *Universal Background Model* (modelo de fondo universal). GMM independiente de locutor utilizado para adaptar modelos de locutor vía MAP en sistemas de reconocimiento de locutor independiente de texto GMM-UBM.

**UPGMA** Unweighted Pair Group Method With Averaging.

**VQ** *Vector Quantization* (cuantificación vectorial). Técnica de compresión de datos utilizada como sistema de reconocimiento de locutor independiente de texto o para la reducción del conjunto de observaciones en sistemas dependientes de texto.

**WPGMA** Weighted Pair Group Method With Averaging.

# 1. Introducción

## 1.1 Motivación del proyecto

En la actualidad, el reconocimiento de voz es una de las tecnologías biométricas más utilizadas debido, principalmente, a los siguientes motivos: un gran avance de las Tecnologías de la Información y las Comunicaciones (TIC); una mayor aceptación social dado que se trata de una técnica no invasiva y cuya adquisición no supone grandes costes; la aparición de aplicaciones móviles que basan su funcionamiento en la voz y, especialmente, a la gran cantidad de información que contiene la voz como rasgo biométrico (identidad del locutor, idioma, edad, estado emocional, etc.).

Se puede hablar de dos grandes áreas en ámbito del reconocimiento de voz: el reconocimiento del habla y el reconocimiento de locutor. En el área reconocimiento del habla, el objetivo es extraer el contenido lingüístico de una locución. Por otra parte, en el área de reconocimiento de locutor el objetivo se centra en la extracción de la identidad del locutor que pronuncia una locución. Es en este último ámbito donde se enmarca el presente proyecto.

En los sistemas de reconocimiento de locutor se usan diversas características extraídas a partir de la voz, usualmente clasificadas en función de su interpretación física: nivel alto, nivel prosódico, nivel prosódico y acústico, y nivel acústico (2.3.1). En las dos últimas décadas, los sistemas de reconocimiento de locutor se han centrado en el nivel acústico/espectral debido a la precisión y eficiencia en reconocimiento mediante técnicas de modelado como GMM-UBM (2.6.2.1) e *i-vectors* (2.6.3.2) para representar los modelos de locutor. Además, han prosperado diferentes tecnologías de compensación de variabilidad como JFA (2.6.3.1), LDA (2.7.2) y PLDA (2.7.1.1) que han permitido mejorar el rendimiento de los sistemas de reconocimiento de locutor.

Uno de los puntos críticos de muchas de estas tecnologías es que necesitan el conocimiento de etiquetas de identidad de locutor de todos los archivos de audio de la base de datos utilizada para su entrenamiento, pues aprovechan dicho conocimiento. Históricamente, en las evaluaciones NIST SRE (*National Institute of Standars and Technology Speaker Recognition Evaluation)* los miles de archivos de audio que se entregaban para el desarrollo de los sistemas de locutor estaban etiquetados con la identidad del locutor.

Sin embargo, no siempre va a existir dicha disponibilidad de una base de datos de audio etiquetada. Para el desarrollo de nuevas aplicaciones en las que se necesite recoger una gran cantidad de audio (por ejemplo, realizar la tarea de reconocimiento de locutor a partir de grandes cantidades de archivos de audio extraídos de videos en *youtube*) el trabajo que implica etiquetar todo el audio de manera manual puede alargar extremadamente el tiempo y los costes necesarios para la implementación de dichas aplicaciones.

## 1.2 Objetivos del proyecto

Como se menciona en la sección anterior, los sistemas de reconocimiento de locutor del estado del arte hacen uso de técnicas de compensación de variabilidad que necesitan el conocimiento de etiquetas de identidad de locutor. Sin embargo, si se desea generar una base de datos de audio (decenas de miles de archivos de audio para varias centenas o miles de locutores) para una aplicación de reconocimiento de locutor que aproveche dichas técnicas el coste de obtener manualmente las etiquetas de identidad de locutor puede ser muy perjudicial.

Este proyecto se centra en la realista situación de no tener audio con etiquetas de identidad de locutor y en su extracción de manera automática para el entrenamiento de los sistemas de reconocimiento de locutor.

La solución propuesta se basa en la obtención de agrupaciones por locutores, mediante *clustering* (tomando como observaciones los *i-vectors* de las locuciones), que permitan una correcta estimación de los subespacios de variabilidad, permitiendo así alcanzar rendimientos similares a los que se obtendrían si se dispusieran de etiquetas de locutor conocidas.

Por tanto, el proyecto enfoca varios objetivos:

- Estudiar los sistemas de reconocimiento de locutor basados en *i-vectors*, así como diferentes técnicas de compensación de variabilidad.

- Estudiar diferentes métodos de *clustering* y técnicas de validación de las soluciones obtenidas a partir de dichos métodos.

- Evaluación del sistema mediante la realización de experimentos, analizando el impacto de las diferentes agrupaciones de locutores encontradas en el sistema de reconocimiento de locutor desarrollado.

## 1.3 Metodología

El desarrollo del proyecto se divide en las siguientes fases:

- **Documentación**: En una primera fase, el alumno ha estudiado la literatura sobre el estado del arte actual en técnicas de reconocimiento de locutor independiente de texto y diferentes técnicas de *clustering,* así como documentación sobre las bases de datos de audio que se utilizarán en el proyecto (NIST SRE 2010,NIST SRE 2008, NIST SRE 2006 y NIST SRE 2004).

- **Estudio del software:** En una segunda fase, el alumno se ha familiarizado con los sistemas desarrollados por el grupo ATVS para el reconocimiento de locutor,

la estructura de servidores de la que dispone el grupo y los *toolkits* de Matlab necesarios para el desarrollo de los experimentos.

- **Experimentos y desarrollo de software:** Posteriormente, se han realizado experimentos haciendo uso de las bases de audio anteriormente mencionadas. Todo el código desarrollado se ha organizado para su uso posterior.

- **Evaluación de resultados y elaboración de la memoria:** Se ha realizado un análisis de los resultados obtenidos a partir de las pruebas realizadas. Con los resultados obtenidos y los respectivos análisis realizados, se ha procedido a redactar la presente memoria.

## 1.4 Estructura de la memoria

El presente trabajo se estructura en seis capítulos:

- **Capítulo 1: Introducción.** Este capítulo presenta la motivación para el desarrollo de este proyecto, así como los objetivos a cumplir durante la ejecución del proyecto.

- **Capítulo 2: Estado del arte en reconocimiento biométrico de locutor.** En este capítulo se presenta el estado del arte actual en reconocimiento de locutor independiente de texto. Se presenta la descripción del modo de operación de estos sistemas, como también la descripción de la información presente en la señal de voz. Posteriormente, se describen los tipos de parametrización de las características extraídas de la voz. Adicionalmente, se presentan las herramientas que permiten la medida del rendimiento del sistema, así como diferentes técnicas para mejorar dicho rendimiento. Finalmente, se describen las diferentes técnicas empleadas en el estado del arte del reconocimiento de locutor independiente de texto.

- **Capítulo 3: Agrupamiento jerárquico de locutores.** En este capítulo se presenta la aproximación propuesta para la realización del *clustering*, etapa que tiene como objetivo la búsqueda de agrupaciones de locutores que permitan hacer uso de técnicas supervisadas de compensación de variabilidad como LDA y PLDA.

- **Capítulo 4: Descripción del sistema.** En este capítulo se presenta la descripción del sistema implementado. Además, se detallan los tipos de parametrización de características empleados y las técnicas empleadas en el reconocimiento de locutor.

- **Capítulo 5: Experimentos y resultados.** En este capítulo se describe el entorno experimental del sistema, así como los resultados obtenidos a través de la secuencia de experimentos realizados.

- **Capítulo 6: Conclusiones y trabajo futuro.** En este capítulo se presentan las conclusiones extraídas del proyecto realizado y las futuras líneas a seguir en este ámbito.

# 2. Estado del arte en reconocimiento biométrico de locutor

## 2.1 Introducción

En este capítulo se presenta el estado del arte en los sistemas de reconocimiento biométrico, prestando mayor atención a aquellos basados en la señal de voz. Además, se describe el estado actual de las técnicas empleadas en los sistemas de reconocimiento de locutor, así como aquellas que se utilizan para la extracción de características a partir de la señal de voz. Finalmente, se describen las herramientas comúnmente utilizadas para la evaluación de este tipo de sistemas.

## 2.2 Sistemas de reconocimiento biométrico

Los sistemas de reconocimiento biométrico se basan en el reconocimiento de patrones, datos biométricos o *rasgos biométricos* extraídos a partir de rasgos físicos o conductuales de una persona. Mediante la comparación entre los rasgos biométricos extraídos y los rasgos biométricos registrados en un sistema se puede reconocer la identidad de una persona.

### 2.2.1 Características de los rasgos biométricos

Una característica física o conductual puede ser usada como identificador biométrico mientras satisfaga los siguientes requisitos [Maltoni et al., 2003]:

- **Universal:** todo el mundo debe poseer dicha característica.
- **Distintivo:** los individuos deberán ser suficientemente diferentes en términos de ese rasgo. Es decir, tiene que poseer capacidad de diferenciación.
- **Estable:** la característica debe permanecer invariable a lo largo de un periodo de tiempo aceptable.
- **Evaluable:** el rasgo debe de poder ser medido cuantitativamente.

En los sistemas de reconocimiento biométrico también se deben cumplir los siguientes requerimientos:

- **Rendimiento:** el rasgo debe permitir alcanzar una precisión de reconocimiento, una velocidad y una robustez aceptables.

- **Aceptable:** las personas deben estar dispuestas a aceptar el uso de ese rasgo como identificador biométrico. Existen rasgos biométricos que son demasiado intrusivos, por lo que no son aceptados por la sociedad.

- **Seguro:** los sistemas basados en ese rasgo deben ser suficientemente robustos frente a posibles intentos de acceso fraudulentos.

Existen una gran variedad de características que cumplen estos requisitos y que son usados por sistemas reales de reconocimiento como rasgos biométricos. A continuación se enumeran algunos de los más empleados:

- ADN
- Cara
- Escáner de retina
- Firma manuscrita
- Forma de andar
- Geometría de la mano
- Huella dactilar
- Iris
- Venas del dorso de la mano
- Voz

Cada rasgo posee ciertas ventajas y desventajas que favorecen su uso en ciertas aplicaciones y lo imposibilitan en otras. Por tanto, la elección del rasgo biométrico a emplear por el sistema de reconocimiento estará condicionada por el tipo de aplicación final (necesidad de seguridad, aceptabilidad del rasgo, etc.).

Por otra parte, en función del tipo de característica empleada por el sistema biométrico, se puede clasificar el tipo de sistema en dos grandes categorías:

- **Sistema biométrico estático**: Engloba todas las medidas de características corporales o físicas del individuo. En este primer grupo se encuentran, por ejemplo, la huella dactilar, el ADN y el iris.

- **Sistema biométrico dinámico**: Recoge todas las características conductuales del individuo. En este segundo grupo se encuentran, por ejemplo, la firma manuscrita, la forma de andar y la dinámica de tecleo.

El caso de la voz es especial, pues los sistemas de reconocimiento hacen uso de características estáticas (contenido espectral de la voz o características acústicas) y de la evolución temporal de estas características, e incluso otras determinadas por la forma de hablar, como cambios de entonación, uso de las pausas, de determinadas secuencias de fonemas, … dependientes del comportamiento del individuo.

Respecto al nivel de cumplimiento de la voz como rasgo biométrico, ésta presenta un alto grado de ***aceptabilidad***, dado que no requiere un método de adquisición intrusivo para la persona. Además, aunque no se trate de un rasgo tan distintivo como el ADN,

el iris o la huella dactilar, sí que proporciona un alto grado de seguridad en aplicaciones de verificación y puede servir como ayuda para simplificar el trabajo de expertos forenses. Como principal desventaja resalta la baja ***estabilidad***, dado que las características de la voz pueden variar en función de diversos factores como la edad, el estado emocional, la salud, etc.

En la Tabla 1 se muestra el nivel de cumplimiento de las características citadas anteriormente por parte de la voz como rasgo biométrico.

| | Aceptabilidad | Seguridad | Universalidad | Evaluabilidad | Distintividad | Estabilidad | Rendimiento |
|---|---|---|---|---|---|---|---|
| Voz | A | A | M | M | B | B | B |

**Tabla 1.** Cumplimiento de requisitos por parte de la voz como rasgo biométrico. A, M y B denotan los niveles Alto, Medio y Bajo, respectivamente. Tabla adaptada a partir de [Maltoni et al., 2003].

### 2.2.2 Funcionamiento de un sistema de reconocimiento biométrico

Como se ha comentado en la sección 2.2, un sistema de reconocimiento biométrico basa su funcionamiento en el reconocimiento de patrones. De esta forma, el sistema clasifica a los usuarios en función de unos rasgos biométricos preestablecidos.

La Figura 1 muestra el esquema de funcionamiento de un sistema de reconocimiento biométrico. Los módulos dibujados en línea continua son parte indispensable del sistema mientras que los dibujados en línea discontinua son opcionales. La línea vertical dibujada en puntos marca la separación entre la interfaz de usuario y el propio sistema biométrico.

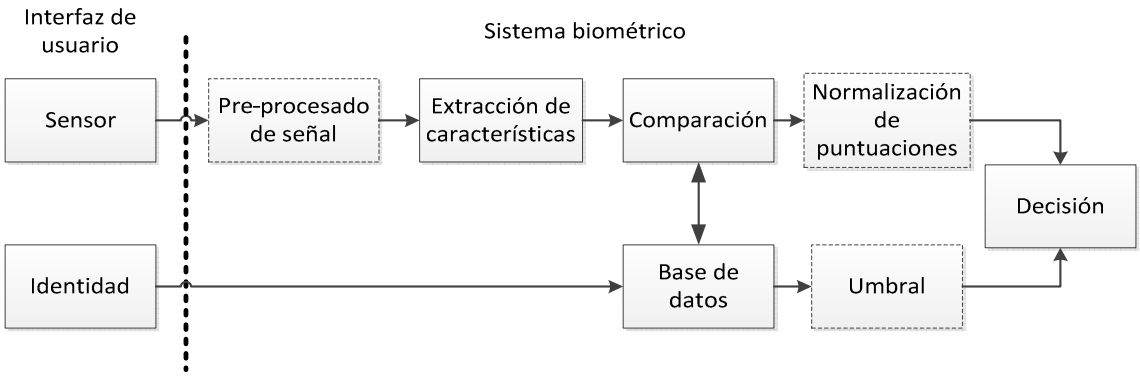


**Figura 1.** Esquema de funcionamiento de un sistema de reconocimiento biométrico.

En la interfaz de usuario se encuentra un **sensor** cuya función será de recoger el rasgo biométrico preestablecido en el sistema biométrico. En caso de un sistema biométrico de verificación, existirá un módulo a través del cual el usuario indicará la **identidad** reclamada (mediante PIN o tarjeta de identificación).

A continuación, es posible que sea necesaria una etapa de **pre-procesado de la señal,** para compensar una posible degradación de ésta o facilitar la posterior **extracción de características**. Dicho módulo de extracción de características será el encargado de extraer de la señal capturada aquellos rasgos más distintivos de la identidad para el rasgo biométrico utilizado.

Una vez extraídas las características se procede a la etapa de **comparación**. En este módulo las características extraídas son comparadas con los patrones de referencia (modelos o plantillas) previamente registrados en la **base de datos.** En caso de un ***sistema de verificación***, se comparan la características extraídas con el patrón de referencia perteneciente a la identidad reclamada y, en caso de un ***sistema de identificación***, se realiza dicha comparación con todos los patrones de referencia almacenados.

Para cada comparación o enfrentamiento (*trial*) que el sistema realiza, se obtiene un valor de similitud entre las características extraídas y el patrón de referencia. Este valor de similitud es llamado comúnmente puntuación o *score*. Las puntuaciones obtenidas pueden ser sometidas a una **normalización**, lo que permite la transformación de éstas a un rango de valores donde sea más fácil identificar si una puntuación pertenece a un usuario genuino o a un impostor.

Finalmente, en función del valor de la puntuación y un valor de **umbral**, el sistema tomará una **decisión:** en caso de un ***sistema de verificación***, el sistema decidirá si las características extraídas se corresponden o no con las de la identidad reclamada; en caso de un ***sistema de identificación***, el sistema decidirá si las características extraídas se corresponden o no con alguna de las identidades registradas en la base de datos.

### 2.2.3 Modos de operación

Como ya se ha introducido en el apartado anterior, dependiendo de la finalidad de la aplicación un sistema biométrico puede operar en dos modos: modo verificación o modo identificación. Previo a estos dos modos, el sistema requiere un modo adicional en el que los patrones de referencia de las identidades a reconocer son almacenados en el sistema biométrico: el modo registro.

- **Modo registro:** En este modo se registran en la base de datos todos los patrones de referencia junto con la información del usuario a reconocer**.** Los usuarios, mediante el sensor, presentan su rasgo biométrico al sistema, que lo pre-procesa en caso de ser necesario y extrae las características identificativas.

Dependiendo de la técnica a emplear en el reconocimiento en que se basa el sistema, estas características pueden constituir en sí mismas el patrón de referencia de la identidad a reconocer o puede ser necesario un proceso de "entrenamiento" a partir de dichas características para generar un modelo de la identidad a reconocer.

Es por ello que a este modo también se le conoce como *fase de entrenamiento* del sistema biométrico. Finalmente, estos patrones de referencia o modelos se almacenan en una base de datos con la información de usuario asociada (nombre, PIN, etc.).

Terminada la fase de registro de todos los usuarios que debe contemplar el sistema biométrico, éste puede ponerse a funcionar en uno de estos dos modos:

- **Modo verificación:** En este modo el sistema valida la identidad de una persona mediante la comparación de los rasgos biométricos de la identidad reclamada, los cuales están registrados en la base de datos como patrón de referencia o modelo, y los rasgos biométricos (capturados por el sensor) de la persona que reclama la identidad.

  En este modo el sistema hace una comparación 1:1, ya que sólo se tiene una identidad reclamada como modelo o patrón de referencia, y las características de un único usuario.

- **Modo identificación:** En este modo el sistema biométrico trata de asignar una identidad, entre las registradas en la base de datos, a un rasgo biométrico identidad desconocida.

  Para ello, realiza una comparación entre las características extraídas del rasgo biométrico y todos los patrones de referencia o modelos almacenados en la base de datos, tratándose por tanto de una comparación 1:N, siendo N el número de identidades registradas en la base de datos.

  Existen dos tipos de identificación: identificación en **conjunto cerrado**, en la que el sistema biométrico debe asignar alguna de las N identidades registradas a la identidad desconocida que representa el rasgo biométrico capturado e identificación en **conjunto abierto**, donde existe la posibilidad de que el rasgo no se corresponda con ninguna de las identidades almacenadas.

La Figura 2 muestra esquemáticamente el funcionamiento de los tres posibles modos de funcionamiento de un sistema de reconocimiento biométrico basado en características extraídas de la voz. La fase de **pre-procesado** que se muestra en la figura dependerá si se requiere compensación de la señal o facilitar al sistema la extracción de características.

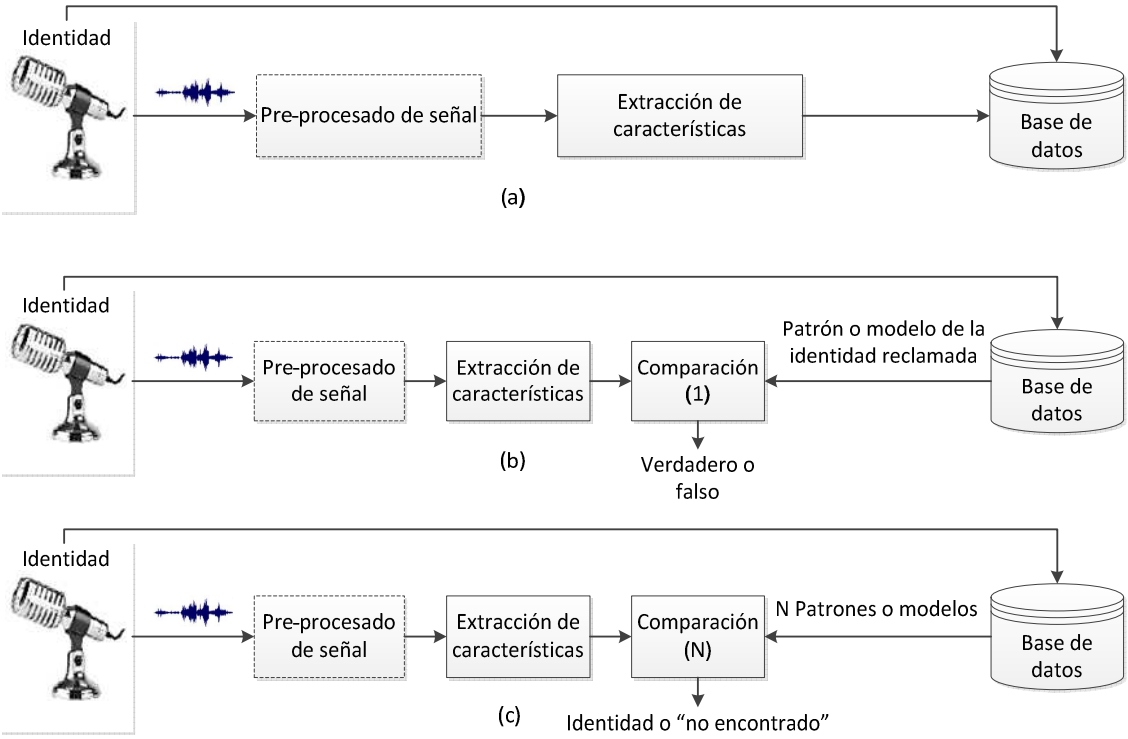


**Figura 2.** Modos de funcionamiento de un sistema de reconocimiento biométrico (a) Modo de registro. (b) Modo de verificación. (c) Modo de identificación.

## 2.3 Información del locutor en la señal de voz

La comunicación oral es la forma más habitual de transmitir información entre las personas. Dicha transmisión de información se logra mediante el uso de mecanismos complejos, los cuales nos permiten construir un mensaje lingüísticamente correcto y articular los distintos órganos que se utilizan en la producción de la señal que transporta el mensaje.

Adicionalmente, la señal generada debe tener una clara información de los fonemas existentes y estar en concordancia con una serie de normas gramaticales para poder ser comprendida por los receptores, que además tienen que conocer la lengua en la que se ha emitido el mensaje.

Una vez cumplidos estos requisitos, es una equivocación pensar que las distintas señales generadas por una sola persona presentan uniformidad, pues existen infinitas realizaciones acústicas posibles para la trasmisión de un mismo mensaje en la misma lengua. Esto da lugar a uno de los problemas más graves que se ha de afrontar en las tecnologías del habla: la variabilidad. Esta variabilidad depende de factores como el estado de ánimo, la edad, el estado de salud o la existencia de patologías fonatorias y/o fisiológicas.

En este sentido, la identidad de los locutores está directamente relacionada con características fisiológicas (longitud y forma de tracto vocal, configuración de los

órganos articulatorios, etc.), con factores sociolingüísticos (nivel de educación, contexto lingüístico y diferencias dialectales) y de comportamiento (hábitos lingüísticos, características en la entonación, etc.).

En función de estas características, cada locutor introduce particularidades en la señal de voz que nos permite diferenciarlo de los demás, pues conllevan información sobre su identidad. En estas peculiaridades de las características extraídas de la señal de voz reside la base del reconocimiento de la identidad del locutor. Por tanto, nuestro trabajo se debe centrar en la extracción eficiente de características individualizadoras, dependiendo del tipo o nivel de información en el que estemos interesados.

### 2.3.1 Niveles de información

Existen multitud de estudios en los que se muestran los mecanismos mediante los cuales las personas son capaces de reconocer la identidad de los distintos locutores. Todos ellos parecen apuntar a que la clave está en la combinación de los distintos niveles de información, así como en el peso o la importancia que se le otorga a cada uno de ellos.

Los sistemas de reconocimiento automático de locutor tratarán de asemejarse al comportamiento humano, combinando las distintas fuentes de información de la mejor manera posible [Reynolds, 2003].

Las particularidades de la voz pueden dividirse en cuatro grandes grupos según el nivel de información en el que se den: nivel lingüístico, nivel fonético, nivel prosódico y nivel acústico.

- **Nivel lingüístico**

  En este nivel se encuentran las características idiolectales [Doddington, 2001]. Estas características describen la forma en que el locutor hace uso del sistema lingüístico y se verán influenciadas por diferentes aspectos relacionados a la educación, al origen y a las condiciones sociológicas del locutor.

  En función de estas características se pueden tener sistemas que modelen locutores por la frecuencia de uso de las palabras o secuencias temporales de palabras.

- **Nivel fonético**

  Está compuesto por las características fonotácticas [Carr, 1999], es decir fonemas y sus respectivas secuencias temporales. El uso de estas características conforma un patrón único para cada locutor. Por tanto, los sistemas fonotácticos intentan modelar el uso que los locutores hacen de estas unidades.

- **Nivel prosódico**

  La prosodia se define como la combinación de energía, duración y tono de los fonemas, y es la principal forma de dotar a la voz de sentido y naturalidad. La prosodia consta de elementos comunes para todos los hablantes, permitiendo distinguir el tipo de mensaje: afirmativo, interrogativo o imperativo.

  Sin embargo, cada locutor emplea dichos elementos prosódicos de manera distinta. Dos de los elementos prosódicos más representativos del hablante son el tono y la energía.

- **Nivel acústico**

  Es el nivel más bajo de la clasificación. Está compuesto por las características espectrales a corto plazo de la señal de voz y su evolución a lo largo del tiempo. Estas características están directamente relacionadas con las acciones articulatorias de cada individuo, la forma en que se produce cada sonido y la configuración fisiológica del aparato fonatorio en el mecanismo de producción de voz.

  La información espectral extrae las particularidades del tracto vocal de cada locutor así como su dinámica de articulación. Esta información puede dividirse en dos grupos: información estática e información dinámica.

  La información estática es la extraída del análisis de cada trama procesada de manera individual. Por el contrario, la información dinámica se extrae del análisis de las tramas de forma conjunta, por lo que es posible recoger el cambio entre las posiciones de articulación.

### 2.3.2 Variabilidad de parámetros determinantes de la identidad

Como se comentó al inicio de la sección, la variabilidad es una de las mayores dificultades cuando se ha de trabajar con la señal de voz. Muchos de los factores que influyen en la variabilidad son controlables voluntariamente por el locutor, aunque también hay muchos otros que no lo son.

Si realizamos un análisis más concreto de los factores de variabilidad que están presentes en las características determinantes de la identidad de locutor, éstos se pueden clasificar en dos grandes grupos: factores que se relacionan con *características intrínsecas* del locutor y factores que están relacionados con *características extrínsecas* al locutor.

- **Factores intrínsecos:** Entre estos destacan factores como la edad del locutor, el estado emocional, el estado físico, la velocidad de articulación, etc. Aunque algunos de estos factores son controlables por el locutor, la mayoría no se

puede controlar voluntariamente, motivo por el cual las características de la señal de voz no permanecen fijas.

- **Factores extrínsecos:** Están relacionados con características externas al locutor, tales como el entorno acústico y los dispositivos de adquisición y transmisión. Cada uno de estos factores introduce una serie de características propias como el ancho de banda, un margen dinámico, reverberación, etc.

Ambos factores pertenecen a la variabilidad intra-locutor o inter-sesión (variabilidad para un mismo locutor entre distintas sesiones de captura), siendo deseable que esta variabilidad sea lo menor posible. Mediante técnicas como *Factor Analysis* [Kenny, 2006], se pretende modelar la variabilidad y compensarla. Por el contrario, se intentará también maximizar la variabilidad existente entre distintos locutores (variabilidad inter-locutor) de forma que sea más fácil distinguirlos.

## 2.4 Extracción de características de locutor

La extracción de características de la señal de voz, también conocida como etapa de parametrización, es el paso previo a cualquier sistema de reconocimiento automático. Estas características, en el caso de la señal de voz, idealmente deben cumplir las siguientes condiciones [Rose, 2002; Wolf, 1972]:

- Presentar poca variabilidad para un mismo locutor (intra-locutor) y gran variabilidad entre locutores distintos (inter-locutor).
- Ser robustas frente al ruido y la distorsión.
- Ocurrir de forma frecuente y natural en el habla.
- Ser fáciles de medir a partir de la señal de voz.
- Ser difíciles de imitar.
- Ser independiente de la salud de locutor o variaciones a largo plazo en la voz.

El tipo de característica a emplear en el reconocimiento dependerá del nivel de información al que la identidad del locutor quiera reconocerse. Es decir, existen características que son más aptas para trabajar a nivel prosódico que a nivel acústico, por ejemplo.

Los sistemas de reconocimiento de alto nivel (como los basados en fonemas) necesitan reconocer la secuencia de fonemas de la locución, por lo que deben apoyarse en sistemas de reconocimiento de habla (*Automatic Speech Recognition, ASR*), los cuales hacen uso de las características acústicas y espectro-temporales.

Una posible clasificación de las características es la que atiende su interpretación física [Kinnunen and Li, 2010]. Se han incluido en esta clasificación (entre paréntesis) los niveles de información que tratan de capturar cada tipo de características:

- **Características espectrales a corto plazo (nivel acústico)**

  Se obtienen sobre segmentos de voz de entre 20 y 30 milisegundos de duración. La información espectral a corto plazo está acústicamente relacionada con el timbre, así como de las propiedades de resonancia del tracto vocal. Es decir, estas características modelan como se modifica o como varía la voz durante el recorrido por el tracto vocal.

- **Características de la fuente de voz (nivel acústico)**

  Estas características representan la forma en que se origina la voz en el tracto vocal, es decir, modelan lo que se conoce como flujo glotal.

- **Características prosódicas espectro-temporales (nivel acústico y prosódico)**

  Recogen información de la entonación y el timbre del locutor sobre segmentos de voz de duración entre decenas y centenas de milisegundos.

- **Características de alto nivel (nivel lingüístico y fonético)**

  Capturan particularidades a nivel de conversación de los locutores, como por ejemplo el uso característico de fonemas o palabras determinadas, o secuencias de éstos.

### 2.4.1 Coeficientes MFCC (Mel-Frecuency Cepstral Coefficients)

Los coeficientes *Mel Frequency Cepstral Coefficients* (MFCC) [Davis and Mermelstein, 1980] fueron introducidos originalmente para el reconocimiento de habla y posteriormente adaptados para el reconocimiento de locutor. El proceso de extracción de estos coeficientes se compone de una secuencia de etapas, descritas continuación.

En primer lugar, se divide el flujo de la señal de audio en tramas de duración igual a 20 o 30 milisegundos, que son procesadas de manera individual. Dado que la señal de voz cambia continuamente, esta división en tramas permite que dentro del intervalo temporal limitado por una trama la señal de voz pueda considerarse estacionaria. Además, la división en tramas se realiza usualmente con un solapamiento del 50 % (10-15 ms) con el objetivo de no perder la información de transición existente entre ellas (véase Figura 3).

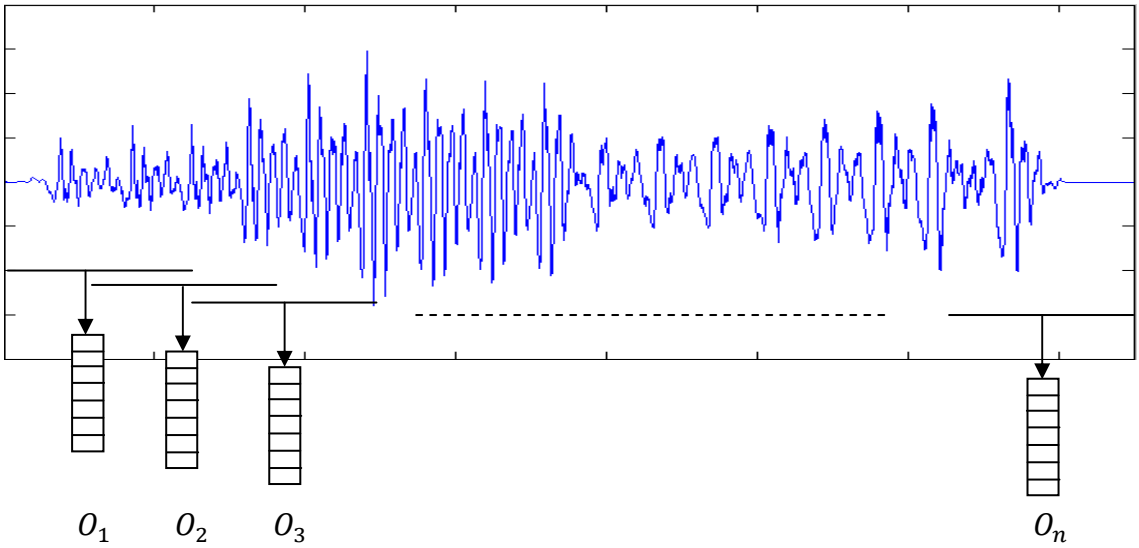


**Figura 3.** División de la señal de audio en tramas para la extracción de características.

A continuación, se realiza el "enventanado" de la señal de voz, normalmente de tipo *Hamming* (véase Figura 4), necesario para un correcto análisis espectral posterior a través de la transformada discreta de Fourier (DFT) [Oppenheim *et al.*, 1999] o su implementación rápida, la FFT (*Fast Fourier Transform*), método normalmente empleado por su mayor eficacia.

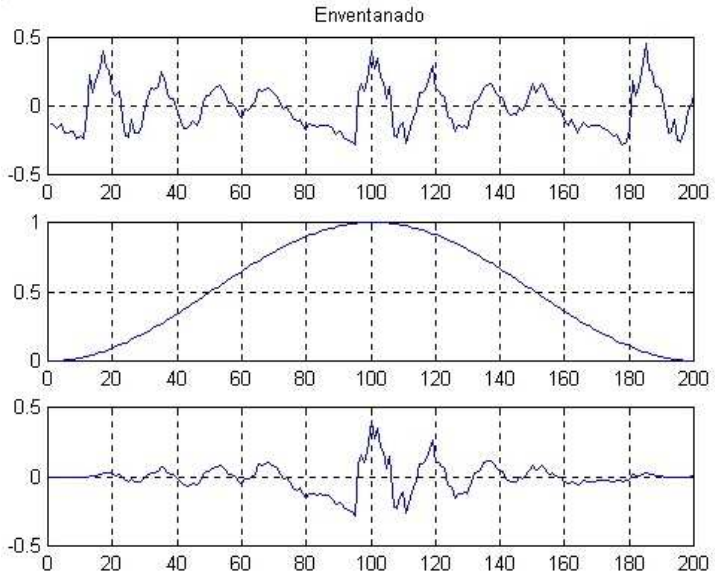


**Figura 4.** Enventanado de una señal con una ventana de tipo Hamming [López et al., 2003].

Como se ha comentado, posteriormente al enventanado se procede al cálculo de la FFT (*Fast Fourier Transform*) de la señal obtenida. Normalmente, sólo se guarda la amplitud del espectro obtenido. La información de dicha envolvente se recoge mediante un banco de filtros perceptual en escala Mel. El objetivo de este filtrado es aproximar la resolución espectral a la respuesta del oído humano mediante la siguiente transformación:

$$Mel(f) = 2595 \cdot \log_{10}\left(1 + \frac{f}{700}\right) \qquad (2.1)$$

A la salida de los filtros, que integran la energía existente de la señal de audio dentro de su ancho de banda, se le aplica el logaritmo natural y luego la transformada discreta del coseno (*Discrete Cosine Transform*, DCT) con el objetivo de comprimir la información en pocos coeficientes. De las salidas de los filtros, denotadas mediante Y(m), m = 1,…, M , los coeficientes MFCC se obtienen a partir de la siguiente transformación:

$$C_n = \sum_{m=1}^{M} [lnY(m)]\cos\left[\frac{\pi.n}{M}(m-\frac{1}{2})\right] \qquad (2.2)$$

donde n es el índice del coeficiente *cepstral*. El vector de características final se forma con 12 o 20 primeros coeficientes $C_n$. La Figura 5 representa esquemáticamente el proceso de obtención de los coeficientes MFCC.

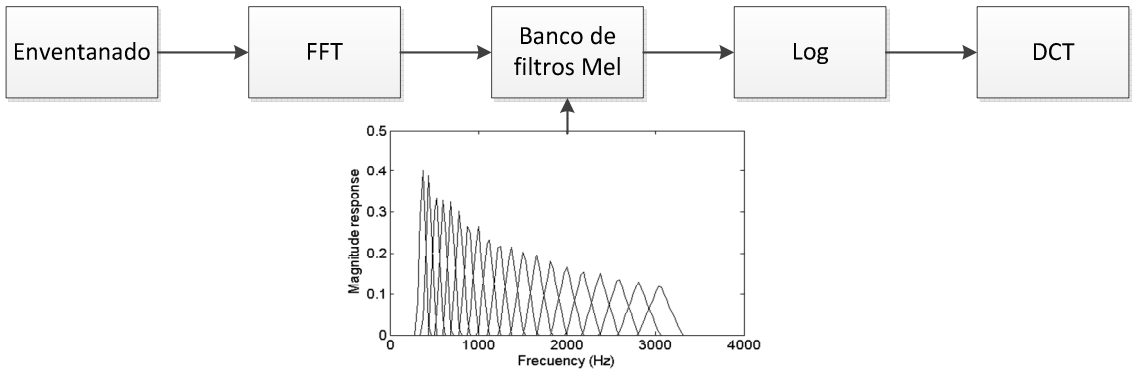


**Figura 5.** Proceso de obtención de los coeficientes MFCC.

Como se ha explicado anteriormente, lo que le interesa al sistema de reconocimiento de locutor es que los vectores de características extraídos a partir de la señal de audio sean identificativos del locutor objetivo. Por este motivo, se hace uso de una etapa de detección de actividad de voz (*Voice Activity Detection*, VAD).

La detección de actividad de voz es una técnica que, mediante algún tipo de algorítmica, extrae de la señal de audio aquellos segmentos que contienen voz útil del locutor objetivo para el sistema de reconocimiento de locutor. Los MFCC se extraerán de los segmentos que el detector de actividad de voz haya determinado que son voz útil. Existen diferentes formas de implementar el detector de actividad de voz, dependiendo de la información que éstos utilicen de la señal de audio para cumplir su cometido.

Actualmente, la mayoría de los detectores de actividad de voz están basados en características temporales o frecuenciales extraídas a partir de la señal de audio. Por ejemplo, en el dominio temporal se utilizan características como la energía promedio de cada trama [Kinnunen and Li, 2010], la tasa de cruces por cero en cada trama (*Zero-Crossing Rate*, ZCR), etc. También se han propuesto detectores de actividad de voz basados en características *cepstrales* [Haigh and Mason, 1993], en medidas de periodicidad [Tucker, 1992], en modelado estadístico de las características extraídas [Sohn et al., 1999] o basados en un detector de fonemas, como en [Schwarz, 2009].

En la Figura 6 se representa un ejemplo de una señal de audio y los segmentos que contienen voz útil del locutor (región sombreada).

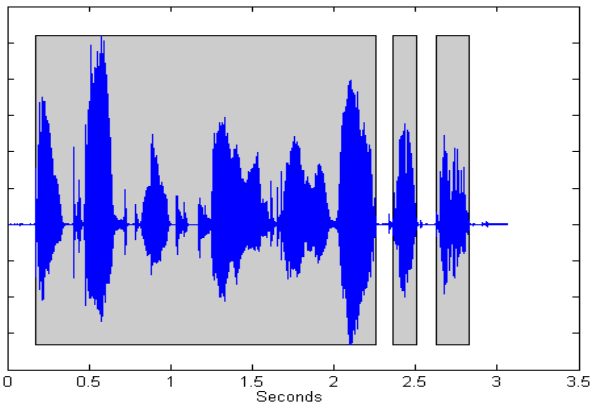


**Figura 6.** Ejemplo de una señal de audio y los segmentos que contienen voz útil (región sombreada).

Antes de finalizar la sección cabe destacar que una de las ventaja de los coeficientes MFCC reside en su naturaleza ortogonal, debida a la DCT aplicada durante el proceso. Esto permite trabajar con matrices de covarianza diagonales en los sistemas basados en GMMs [Reynolds and Rose, 1995], detallados en el apartado 2.6.2, debido a la independencia entre las distintas dimensiones. Por otra parte, la influencia del canal en el dominio *cepstral* se convierte en una componente aditiva, facilitando técnicas como la normalización con respecto a la media *cepstral* (*Cepstral Mean Normalization*, CMN) o el filtrado RASTA (*RelAtiveSpecTrAl*) que permiten reducir la influencia del canal.

### 2.4.2 Normalización de características

Dado que una de las fuentes de variabilidad que afectan a la señal de voz es el canal, existen técnicas de normalización que intentan compensar el impacto que éste pueda tener en la señal de voz. A continuación se presentan distintas técnicas que tratan de compensar la influencia del canal.

- **Normalización por media *cepstral* (*Cepstral Mean Normalization*, CMN)**

  El efecto de canal sobre la voz se modela en el dominio espectral mediante el producto de la señal S($z$), y la función de transferencia del canal, G($z$) [Atal, 1974] [Furui, 1981], esto es:

$$T(z) = S(z) \cdot G(z) \tag{2. 3}$$

  En el dominio *cepstral*, el efecto del canal se convierte en aditivo, al utilizarse el logaritmo de las componentes espectrales (véase la expresión (2.2)):

$$\mathrm{DFT}^{-1}(\log|T(z)|) = \mathrm{DFT}^{-1}(\log|S(z)|) + \mathrm{DFT}^{-1}(\log|G(z)|) \tag{2. 4}$$

Asumiendo que el canal es invariante y que la media de los coeficientes *cepstrales* es nula, el canal puede ser estimado como la media temporal de la señal filtrada T(*z*), por lo que sustrayendo la media temporal de los coeficientes *cepstrales*, el efecto aditivo del canal será minimizada en cierta medida.

$$\mathrm{y[n]} = t[n] - \frac{1}{N}\sum_{n=1}^{N} t[n] \qquad (2.5)$$

donde t[n] representa el vector de características en el instante n de la señal de voz, afectada por el canal g[n], e y[n] representa el vector de características normalizado.

- **Filtrado RASTA (RelAtiveSpecTrAl)**

  El filtrado RASTA [Hermanski and Morgan, 1994] [Malayath et al., 2000] realiza un filtrado paso banda sobre la trayectoria temporal de cada característica (dimensión del vector) en el dominio *cepstral* para suprimir frecuencias de modulación que estén fuera del rango típico de la señal de voz.

  Su funcionamiento se basa en que cualquier constante o componente que varíe muy rápido o muy lento, no se considera habla. Por ejemplo, la función de transferencia de un típico filtro RASTA es la siguiente:

$$\mathrm{H(z)} = 0.1z^{4} \cdot \frac{2+z^{-1}-z^{-3}-2z^{-4}}{1-0.94z^{-1}} \qquad (2.6)$$

- **Feature warping**

  Su objetivo es modificar la distribución de los vectores de características a corto plazo para ajustarse a una distribución *Gaussiana* de media nula y varianza unidad, ya que la distorsión de canal modifica la distribución real de los coeficientes *cepstrales* en cortos periodos de tiempo. Cada dimensión del vector de características se trata independientemente, procesando por tanto todas las dimensiones en paralelo mediante ventanas deslizantes de, típicamente, 3 segundos de duración [Pelecanos and Sridharan, 2001].

### 2.4.3 Coeficientes Δ-Cepstrales

En el apartado 2.4.1 se hizo un enfoque sobre los coeficientes MFCC, también llamados parámetros estáticos. Estos parámetros estáticos modelan las características de la señal de voz a nivel segmental o local. Para incorporar características supra-segmentales a los vectores de características debemos analizar la información transicional que aparece en éstos.

Mediante la primera y la segunda derivadas temporales de los coeficientes *cepstrales* se obtienen coeficientes que tratan de representar la información de coarticulación entre fonemas. A estos coeficientes se les conoce como coeficiente de velocidad, o Δ,

y coeficiente de aceleración, o ΔΔ [Furui, 1981] [Huang et al., 2001] [Soong and Rosenberg, 1988], relacionados con la primera derivada y la segunda derivada, respectivamente. Las expresiones a partir de las cuales se pueden obtener estos coeficientes son las siguientes:

$$C_m^{(0)}[n] = \frac{\sum_{k=-K}^{K} h_k . y_m[n+k]}{\sum_{k=-K}^{K} h_k} \quad (2.7)$$

$$C_m^{(1)}[n] = \frac{\sum_{k=-K}^{K} k . h_k . y_m[n+k]}{\sum_{k=-K}^{K} k^2 h_k} \quad (2.8)$$

Donde $h_k$ es una ventana temporal, normalmente rectangular y simétrica, de longitud 2K+1, y $m$ indica la dimensión del vector característica sobre el que se aplica la derivada. Los coeficientes Δ y $\Delta^2$ se calculan sobre el vector de características estáticas previamente normalizadas, y posteriormente se añaden al mismo vector. Por ejemplo, la Figura 7 muestra el proceso para el caso de 7 coeficientes MFCC (estáticos) con sus primeras derivadas (Δ) para K=1 en el instante $t_n$. El vector final de características estará formado por catorce coeficientes, $\vec{x} = \{C_0, C_1, \dots, C_6, \Delta C_0, \Delta C_0, \Delta C_1, \dots, \Delta C_6\}$.

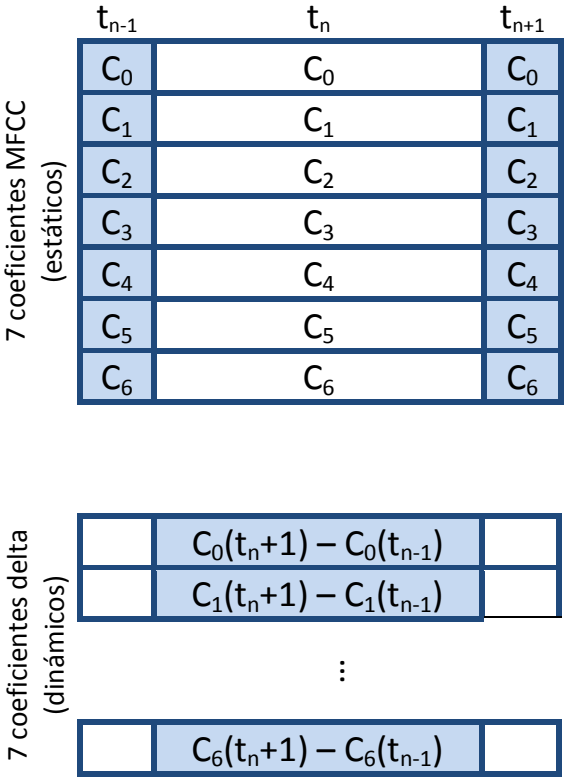


**Figura 7.** Inserción de los coeficientes derivados (información dinámica) a continuación de los coeficientes *cepstrales* (información estática).

## 2.5 Rendimiento de los sistemas de reconocimiento de locutor

En este apartado se hará una descripción de los métodos y herramientas normalmente utilizados para la evaluación del sistema de verificación de locutor y, además, técnicas para mejorar su rendimiento.

### 2.5.1 Relación de Verosimilitud (Likelihood Ratio, LR)

En la literatura relacionada con el campo de la ciencia forense, la relación de verosimilitud o Likelihood Ratio (LR) se define como la relación de las proposiciones o hipótesis del *fiscal* y de la *defensa*. Estas dos hipótesis se definen como sigue:

- $H_p$ (hipótesis del fiscal): el segmento de audio recuperado en la escena del crimen proviene del sospechoso.

- $H_d$ (hipótesis de la defensa): el segmento de audio recuperado en la escena del crimen no proviene del sospechoso.

Por tanto, el ratio o relación se define:

$$LR = \frac{P(E|H_p, I)}{P(E|H_d, I)} \tag{2. 9}$$

donde **E** es la evidencia disponible, la cual está formada por una muestra de origen desconocido y una muestra controlada cuyo origen sí se conoce, e **I** es la información relevante para el caso. Mediante el teorema de Bayes se puede calcular la relación de probabilidades a posteriori teniendo en cuenta el valor de los LR y la información a priori de la siguiente manera:

$$\frac{P(H_p|E, I)}{P(H_d|E, I)} = \frac{P(E|H_p, I)}{P(E|H_d, I)} \cdot \frac{P(H_p|I)}{P(H_d|I)} = LR \frac{P(H_p|I)}{P(H_d|I)} \tag{2. 10}$$

Aplicando la relación de verosimilitud en el reconocimiento de locutor, dado un segmento de habla, ***Y***, y un locutor hipotético, ***S***, el objetivo en los sistemas de verificación de locutor es determinar si el segmento de habla o locución ***Y*** fue dicho por el locutor ***S***. Por tanto, nos encontramos con dos hipótesis:

- $H_0$: ***Y*** pertenece al hipotético locutor ***S***.
- $H_1$: ***Y*** no pertenece al hipotético locutor ***S***.

La toma de decisión entre las dos hipótesis anteriores se basa en el test de la relación de verosimilitud. Este test será idealmente óptimo si se conocen las funciones de verosimilitud que contempla cada hipótesis.

$$\frac{p(Y|H_0)}{p(Y|H_1)} \begin{cases} \geq \theta \ \ se\ acepta\ \ H_0 \\ < \theta \ \ se\ acepta\ \ H_1 \end{cases} \quad (2.\ 11)$$

donde $p(Y|H_i)$, i =0, 1 es la función de densidad de probabilidad de la hipótesis $H_i$ evaluada para el segmento de habla $Y$. En la Figura 8 se muestra el esquema de funcionamiento de un sistema de verificación de locutor basado en la razón de verosimilitud.

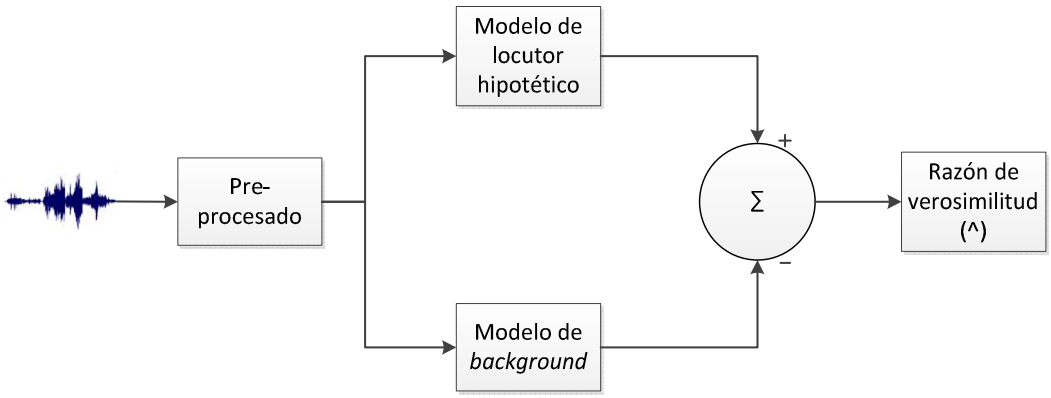


**Figura 8.** Sistema de verificación de locutor basado en relación de verosimilitud.

Partiendo de la señal de voz, en la etapa de pre-procesamiento se extraen las características X = {$x_1$,$x_2$, …, $x_T$}. Posteriormente, estas características son usadas para calcular las funciones de verosimilitud de las hipótesis $H_0$ y $H_1$ y se calcula la razón de verosimilitud. Adicionalmente, se debe tener en cuenta que las hipótesis se representan mediante modelos estadísticos. Por tanto, se designa el modelo $\lambda_{hyp}$ para que represente la hipótesis $H_0$ y el modelo $\overline{\lambda_{\mathrm{hyp}}}$ para que represente la hipótesis alternativa, $H_1$. La nueva razón de verosimilitud, aplicado posteriormente el logaritmo, es:

$$\Lambda = \log\left(p(X|\lambda_{\mathrm{hyp}})\right) - \log\left(p(X|\overline{\lambda_{\mathrm{hyp}}})\right) \quad (2.\ 12)$$

Dentro del contexto de los sistemas de reconocimiento de locutor basados en GMM, el modelo $\lambda_{hyp}$ caracteriza al locutor hipotético ***S*** en el espacio de características de x, mientras que $\overline{\lambda_{\mathrm{hyp}}}$ caracteriza al locutor hipotético dentro del *Universal Background Model (UBM)*, detallado en la sección 2.6.2.1. Es decir, se compara la probabilidad de que las características extraídas provengan del modelo del locutor ($\lambda_{hyp} = \lambda_{target}$) entre la probabilidad que provengan del modelo UBM ($\overline{\lambda_{\mathrm{hyp}}} = \lambda_{UBM}$).

## 2.5.2 Evaluación del rendimiento

El diseño y la implementación de un sistema de reconocimiento de locutor conlleva también una etapa de evaluación. El objetivo de la evaluación es comprobar las capacidades y la bondad del sistema desarrollado. Para ello, mediante un conjunto de pruebas de reconocimiento y con ayuda de herramientas, se evalúa las diferentes técnicas empleadas para el reconocimiento. Estas pruebas deben realizarse en

condiciones lo más parecidas posibles a aquellas en donde el sistema trabaja en un entorno más real, lo que permitirá evaluar el rendimiento de forma más objetiva.

Los sistemas de verificación funcionan normalmente en dos pasos:

En primer lugar, se calcula un valor de similitud (también llamado puntuación o *score*), entre las características de un rasgo biométrico capturado por el sistema y el patrón de referencia de la identidad reclamada. Idealmente, cuanto mayor sea la ***puntuación*** o ***score***, mayor será el apoyo a la hipótesis de que la identidad de ambos coincida. En segundo lugar, mediante el proceso de calibración (2.5.2.1) se obtiene una **relación de verosimilitudes** (2.5.1) que puede ser comparada con un umbral $\theta$. Si el valor de la relación de verosimilitud es mayor que el umbral, el sistema aceptará el rasgo biométrico relacionado a la identidad reclamada. En caso contrario, el sistema lo rechazará. De esta manera pueden darse dos tipos de errores en las decisiones tomadas por el sistema:

- **Error de falso rechazo**: se produce cuando el sistema rechaza el rasgo biométrico de un usuario genuino.

- **Error de falsa aceptación**: se produce cuando el sistema acepta el rasgo biométrico de un usuario impostor.

En base a estos dos tipos de errores se obtienen dos tasas de errores: la tasa de falsa aceptación (*False Acceptance Ratio,* FAR) y la tasa de falso rechazo (*False Rejection Ratio*, FRR). Las tasas de error se definen como el ratio entre el número de errores producidos y el número total de intentos de acceso al sistema o los errores cometidos durante la comparación entre patrones de referencia y características de test.

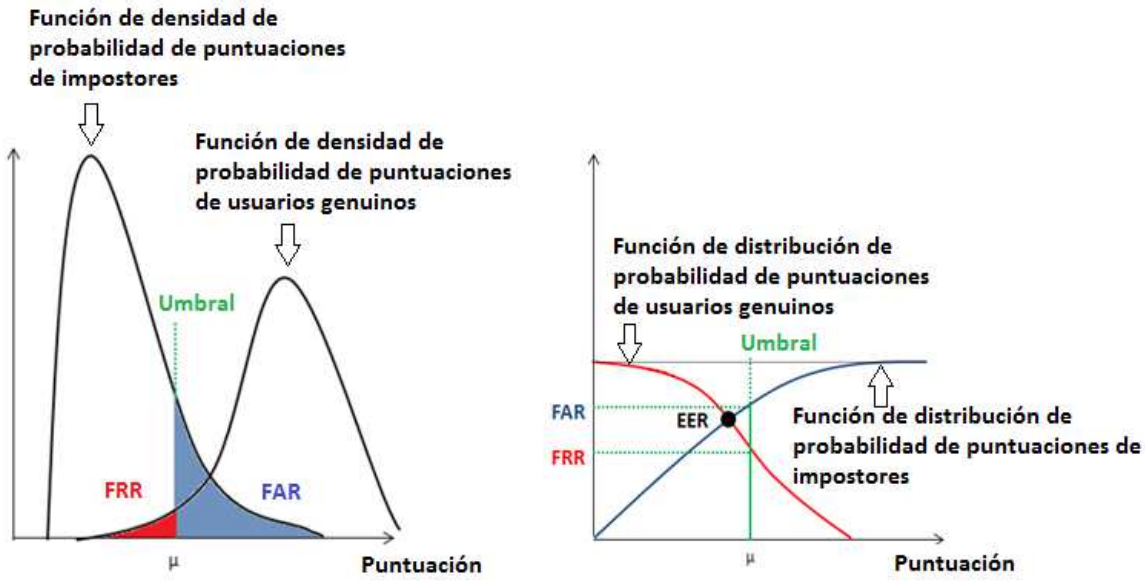


**Figura 9.** Funciones de densidad y distribuciones de probabilidad de usuarios e impostores.

En la Figura 9 se muestran dos formas de representar las tasas de falsa aceptación y de falso rechazo. La gráfica de la izquierda representa la función de densidad de probabilidad de las puntuaciones obtenidas por usuarios genuinos y usuarios impostores. Para un valor de umbral (eje de abscisas), la tasa de falso rechazo (FRR, False Rejection Ratio) o probabilidad de que un usuario genuino sea rechazado es igual al área bajo la curva de densidad de probabilidad de puntuaciones de usuarios genuinos (a la izquierda del umbral). Por el contrario, la tasa de falsa aceptación (FAR, False Acceptance Ratio) o probabilidad de que un usuario impostor sea aceptado se corresponde con el área bajo la curva de densidad de probabilidad de puntuaciones de usuarios impostores (a la derecha del umbral).

En la gráfica de la derecha se representa la función de distribución de probabilidad de puntuaciones de usuarios genuinos e impostores. En esta curva el valor de la tasa de falso rechazo (FRR) y el valor de la tasa de falsa aceptación (FAR) es igual al valor del eje de ordenas correspondiente al valor del umbral en el eje de abscisas. El punto de intersección de las dos curvas, donde el error de falso rechazo y el de falsa aceptación son iguales, se conoce como *Equal Error Rate* (EER).

Es importante precisar que existe una relación directa entre las tasas de error y el valor del umbral. Para un umbral muy bajo, un mayor número de impostores podrían ser aceptados como válidos pero disminuiría el número de usuarios válidos rechazados, mientras que para un valor de umbral muy alto, muchos usuarios válidos serían rechazados pero el número de aceptación de usuarios impostores disminuiría. Por tanto, el valor que se fije en el umbral dependerá de las especificaciones del punto de trabajo deseado para el sistema.

#### 2.5.2.1 Calibración

En los sistemas de verificación de locutor, las puntuaciones (scores) que se obtienen miden la similitud entre las entidades dada una evidencia ***E***. Por ejemplo, mientras el valor de la puntuación sea más alto, más se apoya la hipótesis de que una locución ***Y*** pertenezca a un hipotético locutor ***S***. Sin embargo, no se interpreta como una razón de verosimilitud (Likelihood Ratio, LR). Es decir, una puntuación o *score* no tiene una interpretación probabilística, la cual es requerida en un ámbito forense o para evaluar sistemas independientemente de la función de coste.

Por tanto, el objetivo es calcular el valor de LR para saber el grado de apoyo a una determinada hipótesis. Este proceso de transformación se conoce como calibración. Existen diferentes métodos para la calibración de puntuaciones, pero el más extendido es la transformación lineal de scores [Brümmer and Preez, 2006] mediante regresión logística (*FoCaltooikt http://sites.google.com/site/nikobrummer/focal*).

#### 2.5.2.2 Curvas Tippett

Una forma de valorar los LR obtenidos, en análisis bayesiano de evidencias forenses, es mediante las curvas Tippett. En esta representación, las distribuciones de los valores

de LR de las hipótesis ($H_p$ y $H_d$) son graficadas juntas. De esta forma, se pueden observar las distribuciones y las tasas de fallo simultáneamente. Esta tasa de fallos se define como la proporción de valores de LR que apoyan a la hipótesis incorrecta (LR > 1 si $H_d$ es verdadero y LR < 1 si $H_p$ es verdadero).

#### 2.5.2.3 Función de coste $C_{llr}$

Esta función de coste de log-LR (Logarithmic Likelihood Ratio Cost, $C_{llr}$) es una estimación del rendimiento del sistema sobre un conjunto de valores de LR. Al ser una función de coste que se aplica sobre valores de log-LR (*scores* con sentido forense), permite evaluar tanto la capacidad de discriminación del sistema de reconocimiento como su calibración. Además, al ser una función de coste, cuanto mayor sea el valor $C_{llr}$ peor será el rendimiento del sistema. Esta función se detalla en profundidad en [Leeuwen and Brümmer, 2007] y su expresión queda definida de la siguiente manera:

$$C_{llr} = \frac{1}{N_{H_p}} \sum_{i=1}^{N_{H_p}} \log_2\left(1 + \frac{1}{LR_i}\right) + \frac{1}{N_{H_d}} \sum_{j=1}^{N_{H_d}} \log_2(1 + LR_j) \qquad (2.\ 13)$$

donde $N_{H_p}$ y $N_{H_d}$ son, respectivamente, los números de LR en el conjunto de evaluación cuando $H_p$ o $H_d$ es verdadero. Por otra parte, la función de coste $C_{llr}$ se puede descomponer en otras dos: la pérdida de discriminación, $C_{llr}^{min}$, y la pérdida de calibración, $C_{llr}^{cal}$:

$$C_{llr} = C_{llr}^{min} + C_{llr}^{cal} \qquad (2.\ 14)$$

En la ecuación 2.13, $C_{llr}^{min}$ es la pérdida debida a la limitación del poder discriminativo del conjunto experimental sobre el que se trabaja. Cuanto menor sea, mayor poder discriminativo tendrá el sistema sobre el conjunto experimental. Por tanto, $C_{llr}^{min}$ mide el nivel de discriminación; además, representa el valor mínimo de $C_{llr}$ que puede alcanzar el sistema sin alterar el poder discriminativo del conjunto experimental. El otro sumando, $C_{llr}^{cal}$, sirve como medida de bondad de la calibración al ser la diferencia entre el coste de Log-LR y la pérdida de discriminación. Siempre es un valor positivo y alcanza valores cercanos al cero para sistemas muy bien calibrados.

#### 2.5.2.4 Curvas DET (Detection Error Tradeoff)

Otra forma de representar el rendimiento de los sistemas es mediante la curvas DET. Estas curvas permiten medir la discriminación de un conjunto de puntuaciones y valores de LR. Sobre el eje de ordenadas se representa la probabilidad de falso rechazo y sobre el eje de abscisas la probabilidad de falsa aceptación.

Mediante esta curva resulta más sencillo apreciar el balance entre los dos tipos de errores (FA Y FR). El valor de la tasa de error igual (EER) coincide con la intersección de la curva DET y la diagonal de los ejes de la gráfica (Figura 10).

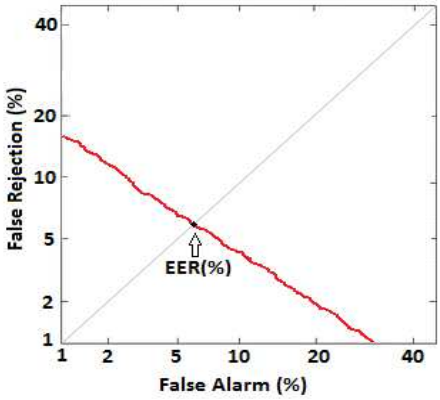

**Figura 10.** Ejemplo de curva DET.

#### 2.5.2.5 Función de detección de coste (DCF, Detection Cost Function)

La función de detección de coste se define de la siguiente manera:

$$C_{DET} = C_{FR} \cdot P_{FR} \cdot P_{Tar} + C_{FA} \cdot P_{FA} \cdot (1 - P_{Tar}) \quad (2.15)$$

Los parámetros que definen la función $C_{DET}$ son los costes relativos de errores de detección, $C_{FR}$ y $C_{FA}$, y la probabilidad a priori ($P_{Tar}$) de que un intento de acceso corresponda a un usuario genuino. Los valores $P_{FR}$ y $P_{FA}$ se obtienen a partir de las funciones de densidad de probabilidad de usuarios e impostores (véase Figura 9).

### 2.5.3 Normalización de puntuaciones o *scores*

Tal como se explicó en el apartado 2.2.2, en un sistema de reconocimiento biométrico puede existir una etapa de normalización de *scores* tras la comparación del patrón de referencia con las características extraídas del rasgo biométrico. Dado que, normalmente, en muchos sistemas el umbral de decisión es común a todos los usuarios, una puntuación determinada puede resultar en el rechazo correcto de un impostor pero también suponer la falsa aceptación de otro impostor. Este hecho se conoce como "*desalinamiento*" en las puntuaciones (véase Figura 11).

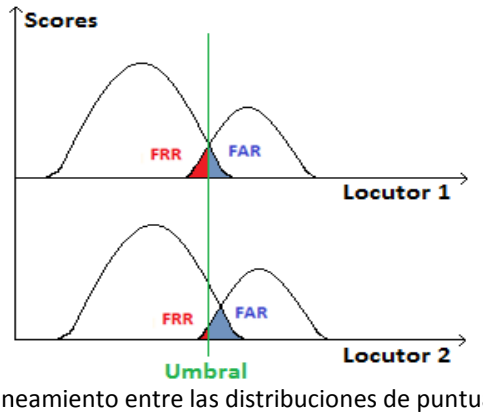

**Figura 11.** Ejemplo de desalineamiento entre las distribuciones de puntuaciones o scores para dos locutores diferentes.

El objetivo de la normalización es proyectar las distribuciones de las puntuaciones, de usuarios e impostores, sobre una función de densidad de probabilidad de media igual a cero y varianza unidad, de forma que las puntuaciones queden alineadas entre diferentes usuarios [Auckenthaler et al., 2000].

Las puntuaciones de impostor son muy usadas para la normalización a partir de la distribución de este tipo de puntuaciones y se conoce como normalización centrada en el impostor [Bimbot et al., 2004] [Fierrez-Aguilar et al., 2005]. La razón de usar las puntuaciones de impostor es que se dispone de mayor número de comparaciones de este tipo en comparación con las puntuaciones disponibles basadas en usuarios genuinos.

#### 2.5.3.1 Z-Norm (Zero Normalization)

La normalización Z-Norm [Li and Porter, 1988] se aplica sobre la puntuación o *score* obtenido tras la comparación entre un rasgo biométrico de test y un patrón de referencia. Para ello, una cohorte de impostores de test se enfrenta con cada patrón o modelo de referencia. A partir de dicha cohorte, se obtiene la distribución de puntuaciones, calculando su media ($\mu_{Znorm}$) y su varianza ($\sigma_{Znorm}$).

Para normalizar las distribuciones de puntuaciones o *scores* se resta la media $\mu_{Znorm}$ a cada uno y se divide cada puntuación entre la raíz cuadrada de la varianza $\sigma_{Znorm}$. Esta distribución de puntuaciones es dependiente del patrón de referencia, de modo que, al aplicar a todos los patrones de referencia la normalización, se alinean sus respectivas distribuciones de impostor para cualquier comparación realizada por el sistema.

$$\bar{s}_{Z_{norm}} = \frac{s - \mu_{Z_{norm}}}{\sigma_{Z_{norm}}} \tag{2. 16}$$

#### 2.5.3.2 T-Norm (Test Normalization)

Se basa en una idea similar a la de Z-Norm, con la diferencia de que los valores de $\mu_{Tnorm}$ y varianza $\sigma_{Tnorm}$ se obtienen de la distribución de puntuaciones de impostor entre una cohorte de patrones o modelos de referencia y un rasgo biométrico de test [Auckenthaler et al., 2000].

Esta transformación es, por tanto, dependiente del rasgo de test y alinea las distribuciones de impostor de todas las puntuaciones de test. Su expresión matemática es la siguiente:

$$\bar{s}_{T_{norm}} = \frac{s - \mu_{T_{norm}}}{\sigma_{T_{norm}}} \tag{2. 17}$$

#### 2.5.3.3 ZT-Norm (Zero and Test Normalization)

En esta normalización, Z-Norm y T-Norm se emplean en cadena o de forma conjunta como sigue:

$$\bar{s}_{ZT_{norm}} = \frac{\frac{s - \mu_{z_{norm}}}{\sigma_{Z_{norm}}} - \mu_{T_{norm}}}{\sigma_{T_{norm}}} \quad (2.\ 18)$$

De la misma manera que en Z-Norm y en T-Norm, a los conjunto de rasgos biométricos de test, en el caso de Z-Norm, y de patrones de referencia, en el caso de T-Norm, se les denomina *cohortes.* Un concepto a reforzar es que en Z-Norm y T-Norm se hace uso de una única cohorte de impostores, de test (Z-Norm) y de modelos de referencia (T-Norm), siendo necesarias dos cohortes de impostores para ZT-norm.

#### 2.5.3.4 S-Norm (Symmetric Score Normalization)

Dada una *cohorte* de impostores, se obtienen las puntuaciones correspondientes a las locuciones de test frente a la *cohorte* (S_test) por un lado y, por otro, se obtienen las puntuaciones correspondientes a los modelos de referencia frente a la cohorte (S_train).

Las puntuaciones de la evaluación (S) se normalizan en media y varianza respecto a las anteriores (S_snorm_test y S_snorm_train) y la puntuación final es la suma:

$$\bar{s}_{norm} = \frac{s - \mu_{test}}{\sigma_{test}} + \frac{s - \mu_{train}}{\sigma_{train}} \quad (2.\ 19)$$

### 2.5.4 Fusión de sistemas

La fusión de sistemas de reconocimiento consiste en la combinación de los resultados de dos o más sistemas con el objetivo de conseguir un sistema más robusto y con mejores prestaciones que los sistemas individuales trabajando por separado [Brümmer et al., 2007] [López-Moreno et al., 2008]. De esta forma, se pueden aprovechar informaciones de rasgos biométricos diferentes o distintos niveles de información procedentes de un mismo rasgo [Reynolds et al., 2003]. La combinación de resultados puede realizarse desde dos perspectivas:

- ***Fusión basada en reglas fijas***

  Combina directamente las puntuaciones obtenidas por los sistemas individuales mediante un operador simple, como la suma, el producto, el máximo o el mínimo. Como requisito es necesario que las puntuaciones se encuentren en un rango de valores homogéneo, dado que puede que el rango de los valores de las puntuaciones obtenidas por los dos sistemas a ser fusionados puede ser muy diferente.

Con este propósito, los tipos de normalización de puntuaciones comúnmente empleados son las normalizaciones *min-max* y *z-score*. La normalización de scores *min-max* transforma el rango de puntuaciones al intervalo [0,1] sin modificar la distribución original de las puntuaciones, mientras que la normalización *z-score* transforma la distribución de las puntuaciones en una distribución con media cero y varianza unidad.

- ***Fusión basada en reglas entrenadas***

  Hace uso de las decisiones (aceptación o rechazo) de los sistemas individuales como patrones de entrada a un nuevo sistema de reconocimiento, tratando la fusión como un problema de clasificación de patrones. Para ello, existen técnicas tales como las redes neuronales, SVMs [Fierrez-Aguilar et al., 2003] o regresión logística [Brümmer and Preez, 2006].

## 2.6 Técnicas de reconocimiento de locutor independiente de texto

En los sistemas independientes de texto no existen limitaciones en cuanto al contenido de las frases empleadas para el modelado de locutores (fase de enteramiento) y las utilizadas como test, por tanto, el contenido léxico de la fase entrenamiento y la fase de test pueden diferir en su totalidad. Esto hace del reconocimiento independiente de texto una tarea muy desafiante.

Durante las dos últimas décadas, estos sistemas han dominado el reconocimiento de locutor, especialmente los basados en características espectrales a corto plazo (o sistemas acústicos). A continuación, se presentan las técnicas más empleadas por los sistemas acústicos.

### 2.6.1 Cuantificación vectorial (Vector Quantization VQ)

El modelo de cuantificación vectorial (VQ), también conocido como modelo centroide, tiene sus orígenes en la compresión de datos [Gersho and Gray, 1991] y fue introducida para el reconocimiento de locutor en la década de los 80 [Burton, 1987] [Soong et al., 1987]. Es uno de los métodos más simples para el reconocimiento de locutor independiente de texto y también se utiliza con propósitos de acelerar el proceso computacional de éstos sistemas [Louradour and Daoudi, 2005] [Kinnunen et al., 2006] [Roch, 2006].

Los sistemas basados en cuantificación vectorial hacen uso de las ventajas teóricas formuladas en la Teoría de la distorsión y del régimen binario de Shannon, que presenta como una de sus conclusiones fundamentales que siempre es posible obtener un mejor rendimiento codificando mediante vectores, que mediante escalares.

La tarea de cuantificación vectorial consiste en cuantificar una señal de entrada que puede tomar valores infinitos mediante la asignación de un vector representativo de entre un conjunto finito posible. Por ejemplo, representando un conjunto de vectores de características próximos entre sí por su vector promedio.

Así, un espacio de características se puede dividir en un número de regiones determinado, típicamente mediante algoritmos de agrupamiento (*clustering*) como K-*means* [Linde et al., 1980]. Cada una de las regiones tendrá su vector representativo o centroide (*codeword*), siendo el conjunto de todos los vectores representativos denominado como libro de códigos o *codebook*.

En el caso de reconocimiento de locutor, la identidad de locutor se puede representar mediante su correspondiente *codebook* (modelo de plantilla del locutor en la técnica de VQ). El número de regiones se establece como potencia de 2, para facilitar la representación de los centroides en notación binaria

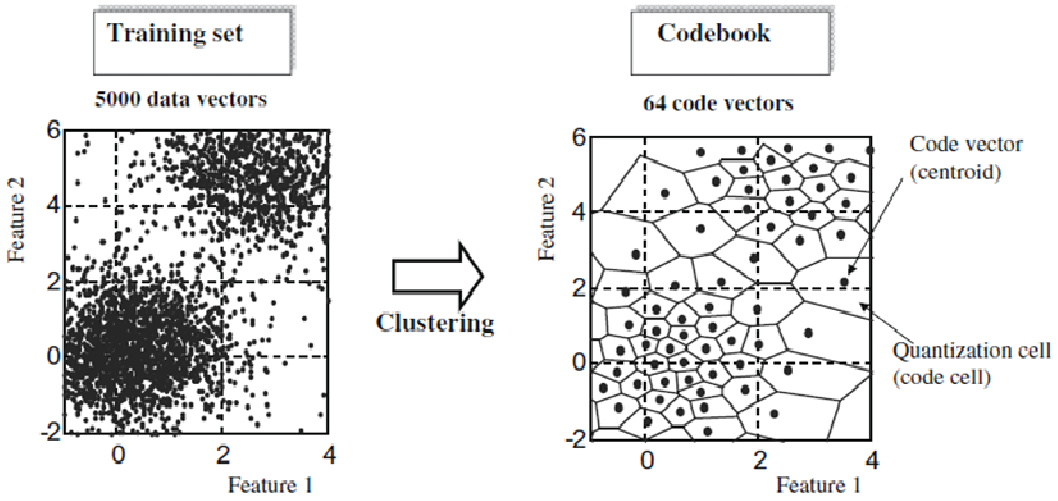


**Figura 12.** Construcción de un *codebook* mediante cuantificación vectorial usando el algoritmo *K-means* [Kinnunen and Li, 2010].

De esta manera, un *codebook* de *b* bits tendrá N = $2^b$ centroides. Por tanto, sólo se almacenan los centroides del *codebook* para cada modelo de locutor, lo que supone una reducción de la información espectral. Los detalles de cómo se realiza la etapa de reconocimiento de locutor se presentan a continuación.

Sean los *codebook* R = {$r_1$, $r_2$, $r_3$,… $r_N$}, se procede a la comparación con una locución de test representada por los vectores de características O = {$o_1$, $o_2$, $o_3$,…, $o_T$}. Dicha comparación se lleva a cabo mediante la distorsión de cuantificación promedio, que se define como:

$$D_Q(O,R) = \frac{1}{T}\sum_{t=1}^{T} min\, d(o_t, r_n); \quad 1 \le n \le N \qquad (2.\ 20)$$

donde $d(o_t, r_n)$ es la medida de la distancia entre el centroide $r_n$ y el vector de características $o_t$. Una métrica de distancia muy utilizada es la distancia euclídea.

Cuanto menor sea la distorsión de cuantificación promedio, mayor será la probabilidad de que el conjunto de vectores de *O* de la locución de test pertenezcan al locutor representado por *R*.

### 2.6.2 Sistemas basados en GMMs (Gaussian Mixture Models, GMMs)

La técnica de modelado mediante GMMs [Reynolds and Rose, 1995] [Reynolds et al., 2000] ha sido, durante muchos años, la metodología de referencia para los sistemas de reconocimiento de locutor independiente de texto.

Los GMMs pueden considerarse como una extensión del modelo VQ, en los que existe cierto solapamiento entre las regiones que divide el espacio de características. Así un vector de características no se asigna a un único centroide, sino que tiene una probabilidad no nula de pertenecer a cualquiera de las regiones.

Dado que, como se ha explicado en la sección 2.5.1, interesa implementar un sistema basado en la razón de verosimilitud, se requiere una función de verosimilitud *p(X|λ)*, siendo en este caso un modelo de mezclas de Gaussianas o GMM.

Un GMM, denotado por λ, está compuesto por un conjunto finito de mezclas de Gaussianas multivariadas que se mezclan en el espacio de características. Esta mezcla de Gaussianas se representa mediante su función de densidad de probabilidad:

$$p(x|\lambda) = \sum_{k=1}^{K} w_k N(x|\mu_k, \sum_k) \qquad (2.\ 20)$$

donde K es el número de Gaussianas del modelo, $w_k$ es la probabilidad a priori (peso de la mezcla) de la *k-ésima* Gaussiana y

$$\mathrm{N}(x|\mu_k, \sum_k) = \frac{1}{(2\pi)^{-D/2}|\sum_k|^{-1/2}} \exp\{-\frac{1}{2}(x-\mu_k)^T \sum (x-\mu_k)_k^{-1}\} \qquad (2.\ 21)$$

es la función de densidad Gaussiana D-variada (D dimensiones), una combinación de densidades Gaussianas uni-modales con vector de media $\mu_k$ y matriz de covarianza $\sum_k$. Las probabilidades a priori están restringidas a sumar la unidad, es decir, $\sum_{k=1}^{K} w_k = 1$ con $w_k \geq 0$.

La dimensión del vector media es Dx1 y, por razones de coste computacional, las matrices de covarianza (dimensión DxD) de los GMM son usualmente diagonales (dimensión de la diagonal Dx1). La naturaleza ortogonal de los coeficientes *cepstrales* MFCC hace que la alta independencia entre las diferentes dimensiones permita usar este tipo de matrices.

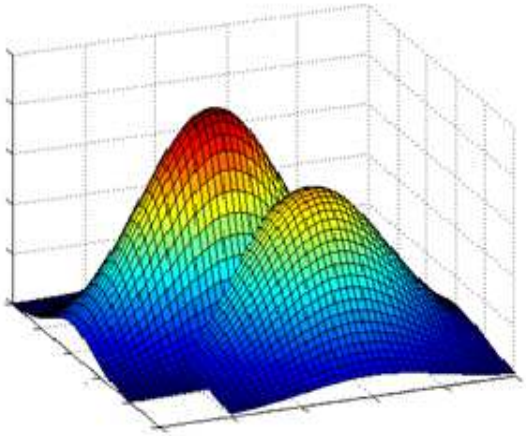

**Figura 13.** Función de densidad de probabilidad de un GMM de 2 Gaussianas en un espacio bidimensional.

El entrenamiento de un GMM consiste en estimar los parámetros de un modelo $\lambda = \{w_k, \mu_k, \sum_k\}_{k=1}^{K}$ a partir de datos de entrenamiento X = {$x_1$, …, $x_T$} y, así, intentar ajustar la distribución del modelo a la de los vectores de características de entrenamiento. Para ajustar el la distribución del GMM a los vectores de características se hace uso del método de máxima verosimilitud (Maximum Likelihood, ML), que pretende estimar los parámetros que maximicen la verosimilitud del GMM dado los datos de entrenamiento.

Para una secuencia de datos X = {$x_1$, …, $x_T$} y asumiendo independencia entre los vectores características:

$$\mathrm{p}(X|\lambda) = \prod_{t=1}^{T} p(x_t|\lambda) \qquad (2.\ 22)$$

La expresión 2.23 no es una función lineal de los parámetros $\lambda$ y por tanto no se puede aplicar directamente el algoritmo de máxima probabilidad (ML). Sin embargo, se puede realizar una aproximación mediante el algoritmo de *Expectation Maximization* (EM), un algoritmo que va cambiando iterativamente los parámetros del GMM. Así, el algoritmo comienza con un modelo inicial $\lambda$ y estima un nuevo modelo $\bar{\lambda} = \{\overline{w_k}, \overline{\mu_k}, \overline{\sum_k}\}_{k=1}^{K}$, de modo que $p\left(X\middle|\bar{\lambda}\right) \geq p(X|\lambda)$.

El nuevo modelo se convierte en el modelo inicial en la siguiente iteración y el proceso sigue hasta que el valor de la probabilidad converge o se alcance un valor máximo de iteraciones. Los parámetros del nuevo modelo serán:

$$\overline{w_k} = \frac{1}{T}\sum_{t=1}^{T} P_r(k|x_t, \lambda) \quad \text{(peso)}, \qquad (2.\ 23)$$

$$\overline{\mu_k} = \frac{\sum_{t=1}^{T} P_r(k|x_t,\lambda)x_t}{\sum_{t=1}^{T} P_r(k|x_t,\lambda)} \quad \text{(media)}, \qquad (2.\ 24)$$

$$\overline{\sigma^2} = \frac{\sum_{t=1}^{T} P_r(k|x_t,\lambda) {x_t}^2}{\sum_{t=1}^{T} P_r(k|x_t,\lambda)} - \overline{{\mu_k}^2} \quad \text{varianza,} \qquad (2.\ 25)$$

$$P_r(k|x_t,\lambda) = \frac{\overline{w_k} N(x_t|\mu_k,\sum_k)}{\sum_{i=1}^{K} w_i N(x_t|\mu_i,\sum_i)} \quad \text{(probabilidad a posteriori),} \qquad (2.\ 26)$$

Por otra parte, puede usarse el método *K*-means para estimar el modelo inicial $\lambda$, de forma que se necesiten menos iteraciones del algoritmo EM. De esta forma, los centroides calculados determinarían los vectores de medias del GMM, las matrices de covarianza de cada gaussiana estarían determinadas por la covarianza de los vectores del conjunto X asignados a cada centroide y los pesos por el porcentaje de vectores del conjunto X asignados a cada centroide. Una vez obtenido el modelo final $\lambda$, mediante el mismo procedimiento, se calcula la probabilidad del conjunto de vectores de una locución frente al modelo.

#### 2.6.2.1 GMM – UBM

La técnica GMM-UBM ó GMM-MAP [Reynolds et al., 2000] hace frente a dos problemas de la técnica GMM clásica:

- Solventa la escasez de datos que existe en muchas ocasiones a la hora de entrenar un modelo de locutor.

- Proporciona un mecanismo que permite ponderar la puntuación de una locución de test en función de lo representativas de la identidad que sean las características extraídas de dicha locución de test.

En los sistemas de reconocimiento basados en GMM-UBM, se entrena primero un modelo universal (Universal Background Model, UBM mediante el algoritmo EM. Este modelo representa la distribución independiente de locutor de todos los vectores de características, es decir, modela las características comunes a todos los locutores. El entrenamiento se realiza a partir de una gran cantidad de audio procedente de un gran número de locutores y de diversas condiciones acústicas. Cuando se registra un nuevo locutor en sistema, los parámetros del UBM se adaptan a la distribución de características del locutor, de forma que el modelo universal o UBM adaptado es el modelo del locutor.

#### 2.6.2.2 Adaptación MAP

En la adaptación de un modelo de locutor, los pámetros del UBM que se pueden adaptar son los pesos, los vectores de medias y las matrices de covarianza. En [Reynolds et al., 2000] se demuestra que sólo adaptando los vectores de medias se consiguen resultados que realizando una adaptación de todos los parámetros que definen el UBM.

Dado el conjunto de vectores de locutor a registrar X = {$x_1$, …, $x_T$} y el modelo UBM $\lambda = \{w_k, \mu_k, \sum_k\}_{k=1}^{K}$, los nuevos vectores de medias adaptados ($\mu_k'$) por el método *maximum a posteriori* (MAP) se obtienen como sumas ponderadas de los datos de entrenamiento de locutor, X, y las medias,$\mu_k$,del modelo UBM:

$$\mu_k' = \alpha_k \frac{1}{n_k} f_k + (1 - \alpha_k)\mu_k \tag{2. 27}$$

donde

$$\alpha_k = \frac{n_k}{n_k + \tau} \tag{2. 28}$$

$$n_k = \sum_t P(k|x_t) \tag{2. 30}$$

$$f_k = \sum_t P(k|x_t)x_t \tag{2. 31}$$

$$P(k|x_t) = \frac{w_k N(x_t|\mu_k, \sum_k)}{\sum_{m=1}^{K} w_m N(x_t|\mu_m, \sum_m)} \tag{2. 32}$$

siendo $n_k$ y $f_k$ los estadísticos de orden cero y primer orden, respectivamente, y $P(k|x_t)$ es la probabilidad a posteriori de ocupación de la Gaussiana. De acuerdo con la ecuación 2.29, se observa que los parámetros $\tau$ (factor de relevancia) y $\alpha_k$(coeficiente de adaptación) controlan la influencia de los datos de entrenamiento sobre el modelo de locutor adaptado respecto al UBM dentro del proceso de adaptación. El modelo adaptado mediante esta técnica se conoce como GMM-MAP y, en el caso de sólo adaptar los vectores de medias, las matrices de covarianza y el vector de pesos son los mismos que los del modelo UBM.

La Figura 14 muetra esquemáticamente el resultado de la adaptación de los vectores de media del modelo UBM a los datos de entrenamiento de locutor.

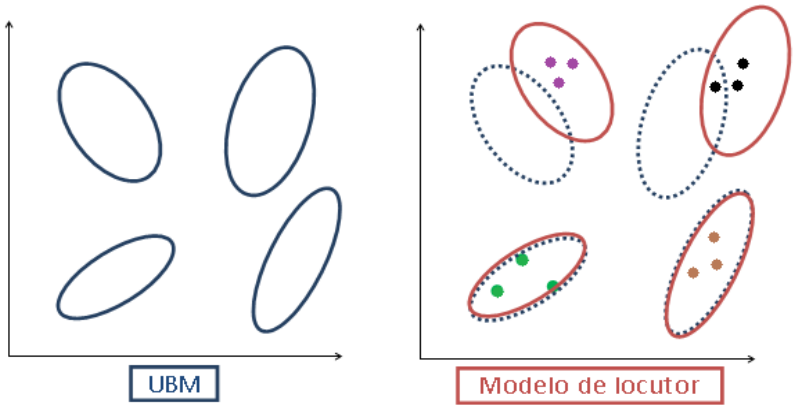


**Figura 14.** Proceso de adaptación MAP de medias del UBM a los datos del locutor.

En la etapa de reconocimiento, se compara la probabilidad de que los datos de test provengan del modelo del locutor adaptado, $\lambda_{target}$, entre la probabilidad de que los datos de test provengan del modelo UBM,$\lambda_{UBM}$. La puntuación o valor de verosimilitud se obtiene mediante la relación de las log probabilidades promedio:

$$LLR_{avg}(X, \lambda_{target}, \lambda_{UBM}) = \frac{1}{T}\sum_{t=1}^{T}\{\log\left(p(x_t|\lambda_{target})\right) - \log\left(p(x_t|\lambda_{UBM})\right)\} \qquad (2.\ 33)$$

De acuerdo con la ecuación 2.33, cuanto mayor sea el valor de la verosimilitud mayor será la probabilidad de que la identidad de la locución se corresponda con la del modelo. Adicionalmente, el empleo de un UBM en común para todos los locutores hace que las puntuaciones de los distintos locutores se encuentren en márgenes comparables, resultando una primera normalización de puntuaciones.

### 2.6.2.3 Supervectores

Un supervector representa de forma compacta la información de locutor presente en un GMM [Campbell et al., 2006a]. Un supervector consiste en la concatenación de los vectores de medias, de dimensión *1xd*, de las *K* Gaussianas de un GMM, obteniéndose así un vector de dimensión *1xKd* (Figura 15).

Esta forma de representación de una locución mediante un único punto en el espacio de supervectores permite eliminar del supervector la variabilidad no deseada. Este proceso se llama *compensación explícita de la variabilidad inter-sesión* y se puede realizar a través de varias técnicas [Burget et al., 2007] [Kenny et al., 2008] [Vogt and Sridharan, 2008].

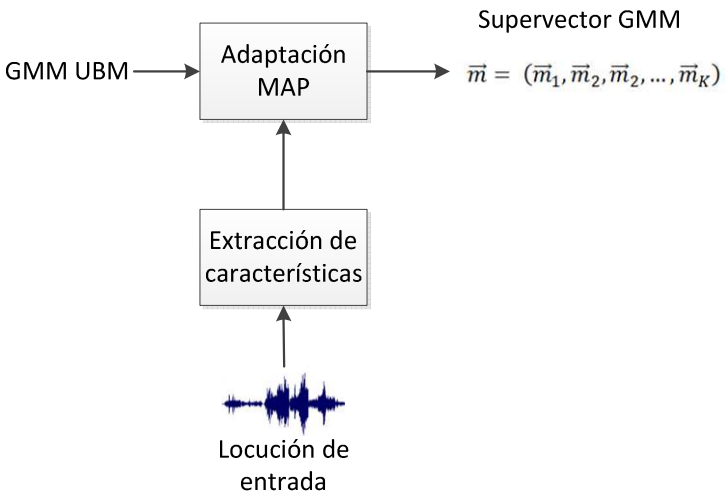


**Figura 15.** Concepto de supervector.

La ventaja de estas técnicas es que no necesitan conjuntos de entrenamiento que contemplan cada tipo de canal o entorno para cada locutor, sino que se entrena un modelo de variabilidad inter-sesión independiente de locutor que luego puede

aplicarse al supervector de cualquier locutor. Las técnicas de *Factor Analysis* (FA) aplican esta compensación sobre sistemas basados en GMMs.

## 2.6.3 Técnicas de Factor Analysis (FA)

En los últimos años, los sistemas de verificación de locutor independiente de texto se han centrado en el uso de técnicas basadas en *Factor Analysis (FA)* por su habilidad para tratar con la variabilidad de inter-sesión. A continuación, se presenta una descripción de las técnicas enmarcadas dentro de este ámbito.

### 2.6.3.1 Joint Factor Analysis (JFA)

La técnica Joint Factor Analysis (JFA) se basa en el modelado conjunto tanto de la variabilidad intra-locutor como de la variabilidad debida al canal [Kenny, 2006]. El modelo de locutor utilizado es el supervector (2.6.2.3), es decir, la concatenación de las medias del GMM de locutor, previa adaptación MAP únicamente de las medias a partir un único modelo UBM. Por tanto, los modelos adaptados comparten el vector de pesos y las matrices de covarianzas.

Para un locutor determinado, los supervectores obtenidos de distintas locuciones de entrenamiento pueden diferir debido a la variabilidad de canal de transmisión. Por tanto, se requiere una compensación de esa variabilidad de forma que datos procedentes de un canal distinto al del entrenamiento puedan ser comparados correctamente con el del modelo de locutor. Para ello, es necesario modelar la variabilidad de forma explícita.

El modelado JFA asume que hay una variabilidad no deseada dentro de un subespacio de baja dimensionalidad que modifica el supervector de locutor *s* para una locución *h:*

$$s = m_s + Ux_h \tag{2. 34}$$

donde **U** representa el subespacio de variabilidad de sesión y las componentes de $x_h$ son los factores de canal o *channel factors* y dependen de la locución ***h.*** Éstos se estiman a partir de los datos de entrenamiento del locutor y determinan la importancia de cada dirección de variabilidad en **U**. Las columnas de la matriz **U** se denominan *eigenchannels* y son estimadas a partir de un conjunto de datos entrenamiento con gran variabilidad de canal.

Respecto al supervector de locutor $\boldsymbol{m_s}$, se puede descomponer de la siguiente manera:

$$m_s = m + Vy_s + Dz_s \tag{2. 35}$$

donde:

***m***: es el supervector de medias del UBM.

***V***: contiene la varianza de locutor y es una matriz rectangular cuyas columnas se denominan *eigenvoices*.

$\boldsymbol{y_s}$: son los pesos que representan al locutor ***s*** en el subespacio de variabilidad de locutor expandido por ***V***. A cada componente del vector $\boldsymbol{y_s}$ se le denomina *speaker factor.*

***D***: representa el desplazamiento del supervector como resultado de la adaptación MAP. Lo conforma una matriz diagonal de (*Kd* x *Kd*), siendo $\boldsymbol{z_s}$ un vector columna de *Kd* x 1.

De acuerdo con la expresión 2.35, si $\boldsymbol{y_s} = 0$, entonces $m_s = m + Dz_s$, lo que describe el proceso de adaptación MAP. Por tanto, la técnica JFA puede verse como una expansión de la técnica de adaptación MAP con la inclusión del modelado *eigenvoice* $Vy_s$.

Dicho producto restringe la adaptación de medias en el entrenamiento del modelo del locutor a las direcciones dadas por $y_s$ y dentro del subespacio determinado por $V$. En el proceso de identificación, se obtienen los denominados estadísticos de orden cero (vector de dimensión *1*x*K*) y primer orden (vector de dimensión *1*x*Kd*) de la locución test frente al UBM, dados por las ecuaciones 2.29 y 2.30 respectivamente.

#### 2.6.3.2 i-vectors

Los sistemas basados en *i-vectors* [Dehak et al, 2011] se han convertido en el estado del arte en los sistemas de verificación de locutor debido a la capacidad de reducción de la gran dimensión de los datos de entrada reteniendo la información más relevante.

$$s = m + Tw \tag{2. 36}$$

De acuerdo con la expresión 2.36, se hace uso de un conjunto de factores de variabilidad total (*Total Variability*, TV) ***w*** para representar, mediante la matriz de variabilidad total T, la locución de un determinado locutor (en sistemas GMM, un supervector ***s)***. A éstos factores de variabilidad total se les conoce normalmente con el nombre de *i-vectors*.

## 2.7 Técnicas de *scoring* en sistemas de reconocimiento de locutor basados en i-vectors

Una vez obtenido los *i-vectors* entre el locutor y el modelo o plantilla de referencia a enfrentar en la tarea de reconocimiento, existen diferentes formas de obtener una puntuación o score. En este trabajo se presentan tres: el score similitud coseno (Cosine Similarity Score, CSS), el score similitud coseno entre ambos *i-vectors* con inter-variabilidad compensada mediante Linear Discriminant Analysis (LDA) [Dehak et al, 2011] y Probabilistic Linear Discriminant Analysis (PLDA) [Prince and Elder, 2007].

### 2.7.1 Score similitud coseno (CSS)

La obtención de una puntuación o score se puede realizar de manera sencilla pero eficaz a través el cálculo de la similitud coseno. De ésta manera, dados dos i-vectors generados mediante la proyección de dos supervectores en el espacio de variabilidad total (TV), la similitud coseno se define como:

$$score(iv_1, iv_2) = \frac{(iv_1)^t \cdot iv_2}{\|iv_1\| \cdot \|iv_2\|} \quad (2.\ 37)$$

Es importante destacar que el score similitud coseno considera el ángulo entre los dos *i-vectors* y no sus magnitudes. Si existe información no relacionada con el locutor (como el canal o la sesión) que esté afectando a la magnitud de los *i-vectors*, no tenerla en cuenta en la etapa de obtención de puntuaciones puede mejorar la robustez del sistema.

#### 2.7.1.1 Linear Discriminant Analysis (LDA)

La puntuación o *score* se calcula de la misma manera que en la sección 2.7.1, es decir, mediante el cálculo de la similitud coseno. La diferencia radica en que, previo cálculo de dicha similitud, los dos *i-vectors* involucrados se proyectan a un espacio de menor dimensionalidad a través de LDA.

LDA se basa, conceptualmente, en la búsqueda de una nueva base ortogonal en el espacio de características (*i-vectors*) de menor (o igual) dimensión que mejor recoja la discriminación entre las diferentes clases. En nuestro caso, estas clases son los locutores.

Por tanto, LDA busca minimizar la variabilidad intra-locutor y maximizar la variabilidad inter-locutor simultáneamente, motivo por el cual LDA puede verse como una técnica de compensación de variabilidad. Para trasladar los *i-vectors* al nuevo espacio, hace uso de una matriz de proyección A formada por los mejores *eigenvectors* (los que mayores *eigenvalues* tengan) de la ecuación:

$$M_b v = \lambda \cdot M_w v \quad (2.\ 38)$$

donde λ es la matriz diagonal de *eigenvalues* y las matrices $M_b$ y $M_w$ se corresponden con matrices de covarianza inter-locutor e intra-locutor, respectivamente. Se calculan como sigue:

$$M_b = \sum_{i=1}^{S} (w_i - \overline{w})(w_i - \overline{w})^t \quad (2.\ 39)$$

$$M_w = \sum_{s=1}^{S} \frac{1}{n_s} \sum_{i=1}^{n_s} (w_i^s - \overline{w})(w_i^s - \overline{w})^t \quad (2.\ 40)$$

donde $\overline{w}$ es la media de los i-vectors de cada locutor, S es el número total de locutores y $n_s$ es el número de locutores de cada locutor s. Dado que se hace uso de la intra-variabilidad y la inter-variabilidad de locutor, LDA es una técnica supervisada en el sentido de que necesita etiquetas de locutor.

## 2.7.2 Probabilistic Linear Discriminant Analysis (PLDA)

Probabilistic Linear Discriminant Analysis (PLDA) [Simon J.D. Prince, 2007] puede verse como una operación análoga a JFA, solo que ésta aplica a supervectores y PLDA lo hace sobre *i-vectors*. Por tanto, no es explícitamente una técnica de *scoring*, sino que es una tecnología de compensación de variabilidad que nos permite obtener un score entre dos modelos de locutor. Si se tienen N locuciones de M locutores diferentes, siendo la locución *n* del locutor *m* $\mathrm{x_{n,m}}$, el modelo PLDA se define como:

$$x_{n,m} = \mu + Fh_n + Gw_{n,m} + \xi_{n,m} \quad (2.\ 41)$$

donde:

***μ***: es la media de los datos de entrenamiento.

***F***: matriz cuyas columnas contienen una base del subespacio de inter-locutor.

$\boldsymbol{h_n}$: representa la posición del i-vector $\mathrm{x_{n,m}}$ en el subespacio definido por F. Tiene especial importancia por contener la identidad del individuo n y se conoce como *latent identity variable*.

***G:*** matriz cuyas columnas contienen una base del subespacio de intra-locutor.

$\boldsymbol{w_{n,m}}$: representa la posición del i-vector $\mathrm{x_{n,m}}$ en el subespacio definido por G.

$\boldsymbol{\xi_{n,m}}$: representa el resto de la variabilidad y se conoce como término residual.

En la expresión 2.41 se pueden observar principalmente dos términos:

- $\boldsymbol{\mu + \mathrm{Fh_n}}$: solo depende de la identidad de locutor (no tiene dependencia de *m*), por lo que tiene relación con la variabilidad inter-locutor.

- $\mathbf{Gw_{n,m}} + \boldsymbol{\xi}_{\mathbf{n,m}}$: de manera similar, este segundo término está relacionado con la variabilidad intra-locutor.Existen dos fases en las que participa este modelo: la fase de entrenamiento o *training phase* y la fase de reconocimiento.

En la fase de entrenamiento, se estiman los parámetros que definen el modelo PLDA (matrices F y G, principalmente) a partir de un conjunto de *i-vectors* de entrenamiento. Esto se lleva a cabo mediante un proceso iterativo del algoritmo de Expectación Maximización (EM). En la fase de reconocimiento, se busca determinar si dos *i-vectors* pertenecen o no al mismo locutor. Para ello se realiza como un ratio o relación de probabilidades:

$$\frac{P(x_1, x_2|H_1)}{P(x_1, x_2|H_0)} = \frac{\int g(\bar{x}|\theta)d\theta}{\int g(x_1|\theta)d\theta \int g(x_2|\theta)d\theta} \quad (2.\ 42)$$

donde $\mathrm{x}_1, \mathrm{x}_2$ son los dos i-vectors involucrados en el reconocimiento, $\mathrm{H}_1$ es la hipótesis de que ámbos pertenezcan al mismo locutor y $\mathrm{H}_0$ es la hipótesis contraria. De la misma manera que LDA, PLDA también requiere de etiquetas de locutor.

# 3. Agrupamiento jerárquico de locutores

## 3.1 Introducción

La tarea de agrupamiento o *clustering* consiste en la creación de grupos a partir de un conjunto de observaciones de tal manera que aquellas que acaben perteneciendo a un mismo grupo (*cluster*) sean más similares entre sí que a aquellas que pertenecen a otros grupos.

En los sistemas de reconocimiento de locutor, la tarea de *clustering* es esencial cuando no se dispone de datos con etiquetas de identidad, situación muy realista, para el entrenamiento de sistemas como Linear Discriminant Analysis (LDA) o Probabilistic Linear Discriminant Analysis (PLDA), descritos en la sección 2.7 de este documento. Esta tarea, en el ámbito del reconocimiento de locutor, se conoce como *speaker clustering*, ya que las etiquetas a obtener aportan información sobre la identidad de los locutores.

En este capítulo se presenta la solución adoptada en este trabajo para resolver este problema, mediante el uso del algoritmo de agrupamiento *Agglomerative Hierarchical Clustering* (AHC). El tipo de observación sobre la que se aplica este algoritmo es el *i-vector* (véase sección 2.6.3.1).

## 3.2 Agglomerative Hierarchical Clustering (AHC)

Este algoritmo es una de las aproximaciones de *clustering* predominantes utilizadas en los sistemas de diarización de locutor (*speaker diarization*) [Tranter and Reynolds, 2006], técnica que divide un archivo de audio en varios segmentos de acuerdo a la identidad de locutor mediante *clustering* sobre MFCC's, típicamente.

Normalmente la diarización de locutor se utiliza para responder a la pregunta "¿Quién habla cuando?". La respuesta a la pregunta anterior es una tarea muy importante cuando se desconoce el número de locutores presentes en una locución (por ejemplo, en programas de televisión o radio) y en tareas de verificación para conocer si un locutor objetivo se encuentra entre los locutores presentes en una locución. Por este motivo se ha decidido utilizar en este estudio al ser un campo directamente relacionado con el reconocimiento automático de locutor.

Durante su inicialización, cada observación se establece como un grupo o *cluster* único y, de manera iterativa, se van uniendo dichos *clusters* de dos en dos mediante una métrica de distancia definida.

### 3.2.1 Métricas de distancia

En esta sección se presentan algunas de las medidas de distancia comúnmente utilizadas:

- **Distancia coseno**

$$d_{coseno}(x_i, x_j) = 1 - \frac{x_i x_j'}{\sqrt{(x_i x_i')(x_j x_j')}} \quad (3.1)$$

- **Distancia euclídea**

$$d_{euclídea}(x_i, x_j) = \sqrt{(x_i - x_j)(x_i - x_j)'} \quad (3.2)$$

- **Distancia mahalanobis**

$$d_m(x_i, x_j) = \sqrt{(x_i - x_j)C^{-1}(x_i - x_j)'} \quad (3.3)$$

  donde C es la matriz de covarianza.

Cuando en la unión entre dos *clusters* participa al menos uno que ya contiene más de una observación resulta necesario definir la distancia entre dichos *clusters,* pues las métricas de distancia anteriores solo están definidas entre dos observaciones. Esta distancia suele denominarse método de *linkage*.

### 3.2.1 Métodos de *linkage*

Entre los métodos de *linkage* existentes, los siguientes se encuentran entre los más utilizados:

- **Unweighted Pair Group Method With Averaging (UPGMA)**

  La distancia entre *clusters* se calcula como la distancia promedio entre todos los pares de observaciones que forman los dos *clusters* a ser unidos, esto es:

$$d_{UPGMA}(r, s) = \frac{1}{n_r n_s} \sum_{i=1}^{n_r} \sum_{j=1}^{n_s} d(x_{ri}, x_{sj}) \quad (3.4)$$

  donde r y s son los *clusters* a ser unidos y están formados por $n_r$ y $n_s$ observaciones, respectivamente.

- **Weighted Pair Group Method With Averaging (UPGMA)**

  En WPGMA, la distancia entre *clusters* se calcula de la siguiente manera recursiva: si un *cluster r* ha sido creado combinando otros dos *clusters*, *p* y *q*, la distancia entre el *cluster r* y otro *cluster s* se define como el promedio de la distancia entre *p* y *s* y la distancia entre *q* y *s*. La ecuación 3.5 describe este método:

$$d_{WPGMA}(r,s) = \frac{d(p,s) + d(q,s)}{2} \tag{3.5}$$

- **Shortest Distance (SD)**

  También conocido como distancia al vecino más cercano, este método utiliza la menor distancia entre las observaciones que forman los dos *clusters*:

$$d_S(r,s) = min\left(dist\left(x_{ri}x_{sj}\right)\right), i \in (1,\dots,n_r)\ y\ j \in (1,\dots,n_s) \tag{3.6}$$

- **Furthest Distance (FD)**

  Método de *linkage* contrario al anterior, pues calcula la distancia entre dos *clusters* como la máxima de las distancias entre las observaciones que los forman. También se conoce como distancia al vecino más lejano.

$$d_F(r,s) = max\left(dist\left(x_{ri}x_{sj}\right)\right), i \in (1,\dots,n_r)\ y\ j \in (1,\dots,n_s) \tag{3.7}$$

- **Centroid linkage**

  Este método utiliza la distancia euclídea entre los centroides de los dos *clusters*:

$$d_C(r,s) = \|\bar{x}_r - \bar{x}_s\|_2 \tag{3.8}$$

  donde $\bar{x}_k = \frac{1}{n_k}\sum_{i=1}^{n_k} x_{ki}$.

- **Ward's linkage**

  El método de *linkage* de Ward utiliza la suma incremental de cuadrados, es decir, el incremento de la suma de cuadrados total intra-*cluster* como resultado de la unión de dos *clusters*.

  La suma intra-*cluster* de los cuadrados se define como la suma de los cuadrados de las distancias entre todos los *i-vectors* de un *cluster* y el centroide del *cluster*.

Matemáticamente, se define como sigue:

$$d_W(r,s) = \sqrt{\frac{2n_r n_s}{(n_{r+}n_s)}} \cdot \|\bar{x}_r - \bar{x}_s\|_2 \quad (3.\ 9)$$

donde $\bar{x}_k$ es el centroide del *cluster* k, $n_k$ es el número de *i-vectors* en el *cluster* k y $\|\ \|_2$ es la distancia euclídea.

### 3.2.1 Criterios de parada

En algoritmo AHC finaliza cuando un criterio de parada es alcanzado, por ejemplo, el criterio BIC [Han and Narayanan, 2007], la máxima distancia entre los *clusters* a ser unidos, el máximo número de *clusters* a crear, etc. En los experimentos de este documento se han utilizado dos criterios de parada: la máxima distancia entre los *clusters* a ser unidos y el máximo número de *clusters* que el algoritmo debe generar.

## 3.3 Medidas de rendimiento de *clustering*

Existen diversas maneras de evaluar el rendimiento de un algoritmo de *clustering*, etapa conocida como validación del algoritmo de *clustering*. En esta sección se presentan algunas de las técnicas más comunes de validación de soluciones de *clustering*.

### 3.3.1 Índice Calinski-Harabasz

Este índice [Calinski and Harabasz, 1974], a veces llamado *Variance Ratio Criterion (VRC)*, mide la relación entre la varianza global inter-*cluster* y la varianza global intra-*cluster*. Se define como sigue:

$$VRC_k = \frac{SS_B}{SS_W} \cdot \frac{(N-k)}{(k-1)} \quad (3.\ 10)$$

donde $SS_B$ es la varianza global inter-*cluster*, $SS_W$ es la varianza global intra-*cluster*, k es el número de *clusters* y N el número de observaciones. Las varianzas inter-*cluster* e intra-*cluster* se definen de la siguiente manera:

$$SS_B = \sum_{i=1}^{k} n_i \|m_i - m\|^2 \quad (3.\ 11)$$

$$SS_W = \sum_{i=1}^{k} \sum_{x \in c_i} \|x - m_i\|^2 \quad (3.\ 12)$$

siendo *x* una observación, *k* el número de *clusters*, $m_i$ el centroide del *cluster* $c_i$ y *m* la media global de las observaciones. Cuanto mayor sea este índice, mejor será la solución de *clustering*. Es un criterio que, por cómo está definido, sólo tiene sentido en soluciones de *clustering* obtenidas mediante la métrica de distancia euclídea.

### 3.3.2 Índice Davies-Bouldin

El criterio Davies-Bouldin [Davies and Bouldin, 1979] está basado en un ratio de las distancias inter-*cluster* e intra-*cluster*. Se define como sigue:

$$DB = \frac{1}{k}\sum_{i=1}^{k} max_{j\neq i}\{D_{ij}\} \tag{3. 13}$$

donde $D_{ij}$ es el ratio de distancias intra-inter *cluster* para los *clusters* $i_{th}$ y $j_{th}$.En términos matemáticos:

$$D_{ij} = \frac{\bar{d}_i + \bar{d}_j}{d_{ij}} \tag{3.14}$$

siendo $\bar{d}_i$ la distancia promedio entre los puntos del *cluster* $i_{th}$ y su centroide, $\bar{d}_j$ es la distancia promedio entre los puntos del *cluster* $i_{th}$ y el centroide del *cluster* $j_{th}$ y $d_{ij}$*es la distancia euclídea entre los centroides de los clusters* $i_{th}$ y $j_{th}$.

Cuanto menor sea el índice Davies-Bouldin, mejor será la solución del *clustering*. De nuevo es un criterio que, por cómo está definido, sólo tiene sentido en soluciones de *clustering* obtenidas mediante la métrica de distancia euclídea.

### 3.3.3 Criterio Gap

Una aproximación gráfica común a la validación de soluciones de *clustering* consiste en representar una medida de error para varias soluciones, localizando un "codo" en esta representación. La región en la que aparece el codo es aquella en la que se produce la mayor disminución de la medida de error y se selecciona como la mejor solución de clustering. El criterio Gap formaliza esta aproximación estimando la localización del codo a partir de la solución que mayor valor de Gap tenga. Matemáticamente, el criterio Gap se define como sigue:

$$Gap_n(k) = E_n^*\{\log(W_k)\} - \log(W_k) \tag{3. 15}$$

donde n es el tamaño de la muestra, k es el número de *clusters* y $W_k$ es una medida de dispersión de la distancia intra-*cluster*:

$$W_k = \sum_{r=1}^{k} \frac{1}{2n_r} D_r \tag{3. 16}$$

siendo $n_r$ el número de puntos del *cluster* r y $D_r$ la suma de las distancias dos a dos entre todos los puntos del *cluster* r. Intuitivamente, el criterio Gap compara $\log(W_k)$ con su valor esperado bajo una distribución nula de referencia, siendo la mejor solución aquella que tenga mayor valor de Gap.

El valor esperado $E_n^*\{\log(W_k)\}$ se determina mediante el muestreo de Monte Carlo en una distribución de referencia y $\log(W_k)$ se obtiene a partir de los datos. Este criterio puede aplicarse a soluciones de *clustering* obtenidas con cualquier métrica de distancia. Sin embargo, es computacionalmente muy costoso debido a que para cada solución de *clustering* debe ser aplicado también a los datos de referencia. Los detalles de cómo se modela la distribución de referencia se encuentran en [Tibshirani et al, 2001].

### 3.3.4 Criterio *silhouette*

El criterio *silhouette* [Rouseeuw, 1987] es un ratio que mide, para cada punto u observación (*i-vector*), cómo de similar es ese punto a los otros que pertenecen a su mismo *cluster* respecto a su siguiente *cluster* más próximo. El valor de *silhouette* se define como:

$$S(i) = \frac{b(i) - a(i)}{\max(a(i), b(i))} \tag{3. 17}$$

donde $a(i)$ es la distancia promedio desde el punto $i$ al resto de puntos de su mismo *cluster* y $b(i)$ es la mínima distancia promedio desde el punto $i$ al resto de puntos de otros *clusters*.

Cuando el valor de *silhouette* de una observación es negativo significa que probablemente dicha observación esté mal asignada a su *cluster*. Normalmente, se utiliza el criterio *silhouette* promedio de todas las observaciones para seleccionar la mejor solución de *clustering*. Sin embargo, en este proyecto se utilizará con el objetivo de eliminar *i-vectors*, de manera individual, que hayan podido quedar mal asignados a su *cluster*.

### 3.3.5 Impurezas de cluster y de clase

En [Leeuwen, 2010], su autor propone dos medidas de validación de *clustering*: la impureza de clase (locutor) y la impureza de *cluster*. La impureza de clase mide, para una solución de *clustering*, si las observaciones (*i-vectors*) de cada locutor han quedado asignados a un único *cluster* o si están repartidos entre varios *clusters*.

De forma complementaria, la impureza de *cluster* mide si los *i-vectors* que han quedado asignados a cada *cluster* pertenecen a un único locutor o a varios. A continuación se detallan formalmente ambas impurezas.

#### 3.3.5.1 Impureza de cluster

Sea una solución de *clustering* que ha encontrado los *clusters* $C_i, i = 1, \ldots, C$, cada uno con uno o más *i-vectors* *j* y $R(j)$ el locutor de referencia para el *i-vector j*. Si $f_k(R(C_i))$ es la frecuencia de ocurrencia del locutor de referencia *k* en el *cluster* $C_i$ y se ordenan los *k* locutores de referencia en orden decreciente de frecuencia en $C_i$:

$$f_{ik} = f_k(R(C_i)) \tag{3. 18}$$

Si el número de *i-vectors* en el *cluster* $i$ es $n_i = \sum_k f_{ik}$ y el número total de *i-vectors* es $N = \sum_i n_i$ , entonces la impureza de *cluster* $I^c$ se define como:

$$I^c = 1 - \frac{1}{N}\sum_i f_{i1} \tag{3. 19}$$

#### 3.3.5.2 Impureza de clase

Se parte de un conjunto de *i-vectors* $S_k$ del locutor de referencia *k*. Si se determina, para cada *i-vector* $j \in S_k$ el *cluster* $C(j)$ al que ha sido asignado y se calculan las frecuencias de asignación del locutor *k* a dicho *cluster* en orden decreciente:

$$g_{ki} = g_i(C(S_k)) \tag{3. 20}$$

Se puede entonces definir la impureza de clase $I^s$ como:

$$I^s = 1 - \frac{1}{M}\sum_k g_{k1} \tag{3. 21}$$

donde $m_k = \sum_i g_{ki}$ es el número de *i-vectors* del locutor *k* y $M = \sum_k m_k$. El punto de trabajo que interesará es aquel en el que ambas impurezas se cruzan, es decir, el punto donde ambas impurezas se minimizan de manera conjunta.

# 4. Descripción del sistema

## 4.1 Introducción

En este capítulo se presenta una descripción del sistema de reconocimiento de locutor implementado, detallando las técnicas utilizadas desde la extracción de características hasta la obtención de los *i-vectors,* así como de la solución de *clustering* adoptada. En la última sección se detallan las medidas de rendimiento utilizadas en los experimentos.

## 4.2 Extracción de características de locutor

Para la extracción de las características de locutor, se lleva a cabo un procesado del audio que consiste en varias etapas. La Figura 18 representa esquemáticamente este procesado.

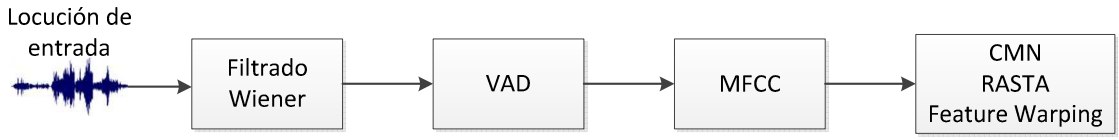


**Figura 16.** Procesado realizado para la extracción de los vectores de características de locutor.

En primer lugar, el audio es pre-filtrado mediante un filtro de *Wiener* [ICSI] para reducir el ruido. Después, se realiza una etapa de detección de actividad de voz (VAD) para eliminar aquellos segmentos de audio en los que determina que no hay voz. Para ello, se ha combinado un VAD simple basado en energía temporal y el VAD que proporciona la herramienta *Sound eXchange* [Sound eXchange software], basado en una medida de la potencia *cepstral* de la señal.

De los segmentos etiquetados como voz, se realiza la extracción de características de locutor. Cada 10 ms. se extraen 40 MFCC (19 MFCC + c0 + Δ) con una ventana de *Hamming* de 20 ms (*secciones 2.4.1 y 2.4.3*). A continuación, se aplican tres técnicas de compensación de canal: *Cepstral Mean Normalization* (CMN), *RASTA* y *Feature Warping* (sección 2.4.2).

## 4.2 Universal Background Model

Para representar el espacio global o universo de características de locutor, se ha entrenado un *Universal Background Model* (UBM, sección 2.6.2.1) independiente de género de 2048 Gaussianas con datos procedentes de evaluaciones del *National*

*Institute of Standars and Technology* de reconocimiento de locutor (*NIST Speaker Recognition Evaluation*) de 2004, 2005, 2006, 2008 y 2010.

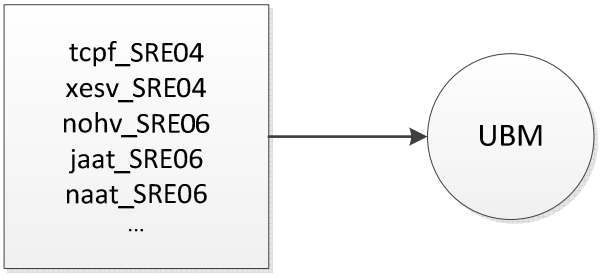


**Figura 17.** Entrenamiento del Universal Background Model (UBM).

## 4.3 Extracción de *i-vectors*

Posteriormente, se ha aplicado *Total Variability* (TV) para la extracción de los *i-vectors* (sección 2.6.3.2), con una matriz de variabilidad total T de 600 dimensiones independiente de género, estimada a partir de los mismos datos utilizados para el entrenamiento del UBM.

Por último, se realiza un procesado de los *i-vectors* que consiste en una etapa de *whitening* (media cero y matriz de covarianza identidad) y normalización de longitud (Lnorm) [Romero and Wilson], para *Gaussianizar* la distribución de los datos y paliar la variabilidad de duración de las locuciones. En la Figura 20 se representa esquemáticamente el proceso de extracción de *i-vectors*.

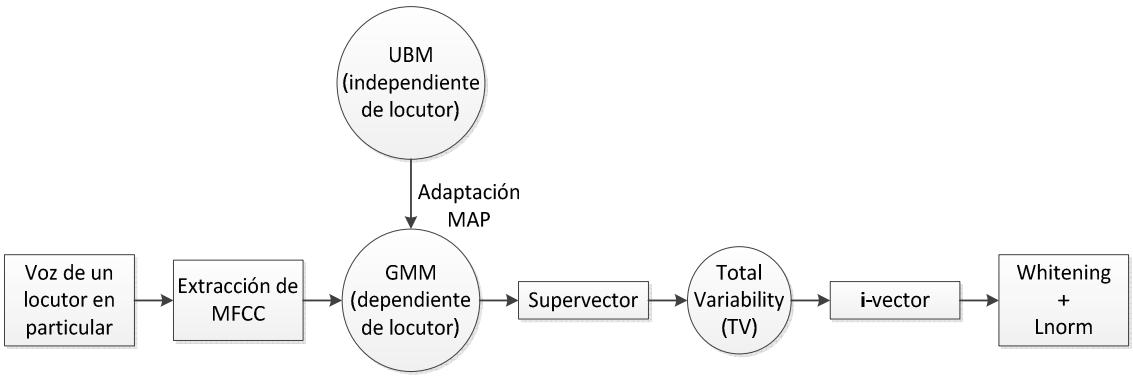


**Figura 18.** Esquema de extracción de *i-vectors* para una locución dada.

## 4.4 Agglomerative Hierarchical Clustering

El objetivo del presente trabajo es buscar una solución a la situación realista de desconocimiento de etiquetas de locutor para el entrenamiento de técnicas de compensación de variabilidad como *Linear Discriminant Analysis* (LDA) o *Probabilistic Linear Discriminant Analysis* (PLDA).

La aproximación realizada se basa, principalmente, en la obtención de agrupaciones de locutores, mediante la técnica de *Agglomerative Hierarchical Clustering* (AHC), detallada en la sección 3.2. Se han utilizado dos criterios de parada: el número máximo de *clusters* y la máxima distancia.

### 4.4.1 Estimación de la máxima distancia

Una forma de proceder para utilizar el criterio de parada de máxima distancia entre *clusters* a ser unidos es la 'fuerza bruta'. Esto es, realizar experimentos en un rango extenso de valores de máxima distancia y seleccionar como valor de parada aquel que mejor rendimiento ofrezca. Sin embargo, es una metodología computacionalmente muy costosa (para cada solución de *clustering* hay que realizar un experimento completo de reconocimiento de locutor) y no siempre la distancia está acotada (por ejemplo, la distancia euclídea no lo está).

Por este motivo, se ha utilizado una representación gráfica para estimar un rango de valores útil en el que realizar experimentos. De cada observación (*i-vector*), se obtiene la distancia a sus K-vecinos más cercanos y, para cada vecino, dichas distancias se ordenan de manera descendiente y se representan gráficamente (véase Figura 19). En la gráfica, cada curva representa al *k* vecino más próximo, creciendo k según nos alejamos del origen.

La representación permite hacerse una idea de la distribución de las distancias entre los *i-vectors*. El rango de valores de interés es aquel en el que se concentran o aquel al que convergen las máximas distancias entre los *i-vectors* y sus K-vecinos más cercanos.

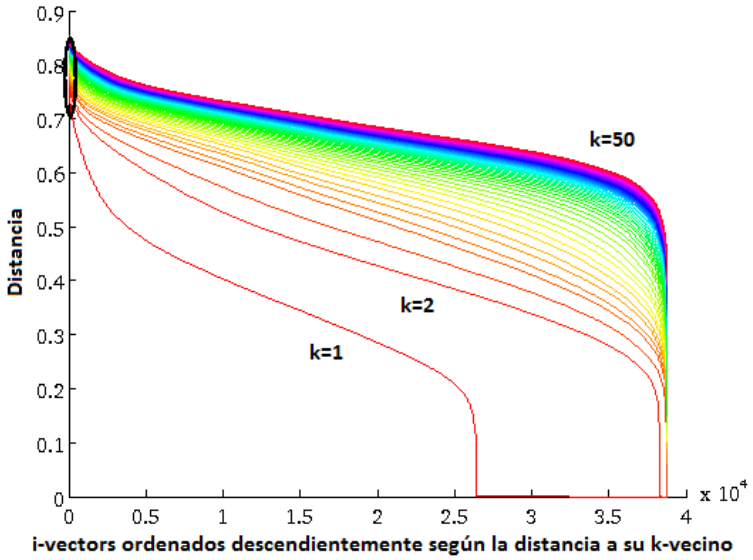


**Figura 19**. Ejemplo de representación gráfica de las distancia de los *i-vectors* a sus K=50 vecinos más próximos ordenadas descendientemente.

### 4.4.1 Utilización del coeficiente silhouette

Si bien en la sección 3.3.4 se presenta el criterio *silhouette* como medida de validación de una solución de *clustering*, en el presente proyecto se ha utilizado con un significado diferente. Como se ha mencionado, el coeficiente *silhouette* aporta información de cómo de similar es cada *i-vector* a los otros que pertenecen a su mismo *cluster* respecto a aquellos del siguiente *cluster* más próximo. Es decir, para cada i-vector, nos da una idea de si ha quedado o no bien asignado a su *cluster*.

En este sentido, se ha utilizado para eliminar *i-vectors* que probablemente no hayan quedado bien asignados a su *cluster*, por ser conceptualmente *i-vectors* que puedan degradar la calidad de los sistemas de reconocimiento de locutor.

## 4.5 Técnicas de *scoring*

En este estudio, se hace uso de dos tecnologías comúnmente utilizadas en el área del reconocimiento de locutor: el *score* coseno y *Probabilistic Linear Discriminant Analysis* (PLDA). Además, también se utiliza una variante del *score* coseno aplicando una etapa previa de *Linear Discriminant Analysis* (LDA), como técnica de compensación de variabilidad, a los *i-vectors*.

Como variante de PLDA se realiza una etapa final de normalización de *scores simétrica* (S-norm), mediante una cohorte de 400 locutores (200 *male* y 200 *female*) representados por *i-vectors* extraídos a partir de locuciones de 300s (150s de habla neta). Dado que LDA y PLDA son tecnologías supervisadas, necesitan etiquetas de locutor. Es por este motivo por el que se hace uso de la etapa previa de *clustering*.

## 4.6 Medidas de rendimiento

En esta sección se detallan las medidas de rendimiento utilizadas en la parte experimental, dividiéndose en dos bloques: medidas de rendimiento del *clustering* y medidas de rendimiento de reconocimiento de locutor.

### 4.6.1 Medidas de rendimiento de *clustering*

Las métricas utilizadas para comprobar el funcionamiento del *clustering* son la impureza de clase y la impureza de locutor, descritas en la sección 3.3.5, por su sencillez y por su significado.

Actualmente, los sistemas de reconocimiento de locutor utilizan las etiquetas de referencia de locutor para el entrenamiento de técnicas como LDA o PLDA. Como esta es la aproximación habitual, se ha decidido evaluar el rendimiento del *clustering* respecto a las etiquetas de referencia, pues el objetivo es que las etiquetas obtenidas

por el *clustering* alcancen resultados similares a los que se podrían obtener sin el conocimiento de etiquetas de referencia.

### 4.6.2 Medidas de rendimiento de reconocimiento de locutor

En este proyecto se han utilizado tres métricas de rendimiento diferentes:

- **Equal Error Rate (EER):** punto en el que los errores de falsa aceptación (FAR) y falso rechazo (FRR) del sistema son iguales (sección 2.5.2).

- **minDCF**: valor mínimo de la función de coste de decisión (DCF, sección 2.5.2.5), tal y como se define en el plan de evaluación del *NIST i-vector Machine Learning Challenge* de 2014 [NIST i-vector Challenge, 2014].

- **minCllr:** valor mínimo del log-likelihood ratio Cost, tal y como se define en [Brummer and Preez, 2006] y [Leeuwen and Brümmer, 2007] (sección 2.5.2.3).

# 5. Experimentos y resultados

## 5.1 Introducción

En el estado del arte del reconocimiento de locutor, los sistemas de reconocimiento de locutor hacen uso del conocimiento de etiquetas de identidad de locutor para la aplicación de técnicas de compensación de variabilidad como LDA o PLDA. Esto significa que las locuciones de las bases de datos de audio utilizadas han de tener etiquetas de locutor.

Dado que la cantidad de archivos de audio en las bases de datos típicamente utilizadas (NIST SRE) es del orden de decenas de miles o superior, el etiquetado manual de cada archivo es una tarea exigente en cuanto a recursos necesarios para llevarla a cabo. Si se quiere desarrollar una nueva aplicación en la que se necesite recopilar una gran cantidad de audio puede que la tarea de etiquetado manual haga inviable la utilización de las técnicas mencionadas de compensación de variabilidad, perdiendo así la mejora en el rendimiento del sistema de reconocimiento de locutor que generalmente conllevan.

En este capítulo se presenta el entorno experimental, detallando la base de datos utilizada, así como los experimentos realizados y sus respectivos resultados para observar el impacto de la falta de etiquetas de locutor (base de datos con locuciones anónimas) en los sistemas de reconocimiento de locutor.

## 5.2 Entorno experimental

### 5.2.1 Protocolos de evaluación

Un protocolo de evaluación se define como el conjunto de condiciones que se imponen a los sistemas de reconocimiento de locutor a implementar. Estas condiciones determinan la base de datos a utilizar para entrenar los modelos de locutor a reconocer y testear, así como la duración del audio usado en la evaluación.

Además, determina el número de enfrentamientos realizados por cada identidad a reconocer y la proporción de locutores por género. De esta forma se mide de forma objetiva el rendimiento del sistema a evaluar. El desarrollo de este proyecto ha seguido el protocolo de evaluación del *National Institute of Standards and Technology (*NIST Speaker Recognition Evaluation, NIST SRE) [NIST SRE].

#### 5.2.1.1 Evaluación NIST

El objetivo de las evaluaciones NIST es impulsar el desarrollo tecnológico, medir el estado del arte y encontrar técnicas novedosas que hagan frente a los nuevos desafíos que se presentan en las tareas de reconocimiento de locutor independiente de texto. Para ello, establece unas condiciones competitivas protocolares que permiten determinar el rendimiento de los sistemas participantes y así comparar las distintas técnicas y configuraciones empleadas.

Las evaluaciones NIST han sido organizadas anualmente desde 1996 hasta 2006 y, a partir de ese año, las evaluaciones NIST han pasado a ser bianuales. A lo largo de los años, las condiciones de la evaluación han evolucionado en muchos sentidos: de tener datos sólo de canal telefónico a incorporar también audio de tipo microfónico, de haber estilo de habla conversacional a incluir habla de tipo entrevista, de tener locutores de un único idioma a ampliarlos a varios idiomas, etc.

Además, las evaluaciones han pasado a ser de carácter abierto, permitiendo así la participación de grupos de investigación, empresas o entidades, con la obligación de presentar los sistemas desarrollado. De esta forma se ha conseguido una mayor competitividad y un gran impulso a la investigación en el desarrollo de sistemas de reconocimiento de locutor.

El gran impacto de estas evaluaciones en la comunidad de reconocimiento de locutor e idioma ha provocado que sus conjuntos de datos y protocolos de evaluación se hayan convertido en un estándar al momento de publicar resultados en publicaciones de este ámbito.

El protocolo de evaluación define la medida del rendimiento (función de coste) y los datos sobre los que realizar las decisiones de evaluación: datos de entrenamiento (segmentos de *train*) para modelar la identidad a reconocer y datos de test (locuciones de prueba o de *test*) para cotejar con los modelos de locutor generados. Por ello, existen diversas condiciones en función de la cantidad y tipo de datos que se dispone para el entrenamiento y *test*:

- **Datos de *train***: desde 10 segundos (10s) de habla en 1 conversación hasta 8 conversaciones (8c) de 300 segundos cada una (150 segundos de habla neta) por locutor.

- **Datos de *test***: desde 10 segundos (10s) de habla hasta 300 segundos (150 segundos de habla neta) por locutor, siempre una única conversación.

La combinación entre una determinada cantidad de habla de entrenamiento y una cantidad determinada de habla de test se denomina *condición* de prueba. Además, en la mayoría de las condiciones se proporcionan los ficheros de audio de ambos locutores presentes en la conversación por separado (condiciones *2-channels* o *4-wire*). Sin embargo, existen otras condiciones en las que las conversaciones para

entrenamiento o test pueden estar mezcladas en un mismo canal (condiciones *summed-channel*).

Respecto a las medidas de rendimiento del sistema de reconocimiento de locutor, en las evaluaciones NIST se emplea la función de detección de coste DCF, definido en el apartado 2.5.2.5. En cada evaluación NIST se proporcionan los costes de falsa aceptación ($C_{FA}$) y falso rechazo ($C_{FR}$), y se establece la probabilidad a priori de que una locución de test dada pertenezca al locutor contra el que se enfrenta, $P_{Tar}$.

### 5.2.2 Base de datos para reconocimiento de locutor

El rendimiento de los sistemas se ve afectado en gran medida por la base de datos utilizada, ya que a partir de ésta se entrenan los *Universal Background Models* (UBMs), las matrices de compensación de las técnicas de *Factor Analysis* (FA), las cohortes para las normalizaciones de puntuaciones como Z-norm y S-norm, etc.

Por tanto, es muy deseable que la base de datos tenga una gran población de locutores y recoja la mayor cantidad de factores de variabilidad posible de la señal de voz: multi-sesión (grabaciones obtenidas en distintos momentos de tiempo), múltiples lenguajes, múltiples condiciones ambientales (con y sin ruido de fondo), múltiples canales de grabación (telefónico, microfónico), etc.
En el desarrollo de nuestro sistema de reconocimiento de locutor y de los experimentos, se ha utilizado un subconjunto de datos de la evaluación NIST SRE 2012 [NIST evaluation plan, 2012] con la mitad de locutores de *train* y locuciones de test de 300 segundos (voz telefónica). En concreto, los conjuntos de datos utilizados son:

- Conjunto de ***development***: Utilizado para el entrenamiento de LDA y PLDA, con 38766 locuciones de 10, 30 y 150 segundos pertenecientes a 1775 locutores, siempre que esos locutores tengan 8 o más locuciones.

  Para el entrenamiento del UBM se añaden a este conjunto los locutores de *train* y aquellos que se han descartado para el entrenamiento de LDA y PLDA por no tener al menos 8 locuciones. Del conjunto de *development* se supondrá, en la sección experimental, que no se conocen las etiquetas de locutor.

- Conjunto de ***train***: Utilizado para el entrenamiento de los modelos de locutor. Contiene 14693 locuciones pertenecientes a 959 locutores. Cada modelo de locutor se forma como el *i-vector* promedio de todos los que pertenecen a dicho locutor.

- Conjunto de **test**: Conjunto de datos utilizado para evaluar el rendimiento de los sistemas de reconocimiento de locutor. Consta de 16218 locuciones y se realizan 370846 enfrentamientos.

## 5.3 Experimentos realizados

### 5.3.1 Sistema de referencia (baseline)

El primer experimento realizado busca la obtención de un sistema de referencia (*baseline*) en el que las etiquetas de locutor de los datos de *development* son conocidas (Tabla 2). El objetivo es evaluar el sistema de reconocimiento en esta situación considerada "ideal" aunque haya sido, históricamente, la situación común en el ámbito del reconocimiento de locutor.

| Sistema | Baseline | | |
|---|---|---|---|
| | EER (%) | minDCF | minCllr |
| CSS | 7,13 | 0,510 | 0,233 |
| LDA | **5,61** | **0,451** | **0,187** |
| PLDA (Lnorm) | 6,73 | 0,491 | 0,249 |
| PLDA (Lnorm+Snorm) | 6,32 | 0,477 | 0,222 |

**Tabla 2.** Resultados obtenidos haciendo uso de las etiquetas de locutor (baseline).

El sistema CSS (*score* coseno sin hacer uso de etiquetas de locutor) ofrece el peor rendimiento, aunque es el sistema más sencillo. Además, si bien PLDA es un sistema, a priori, más potente y debería por tanto tener mejor rendimiento, el sistema que mejores resultados nos ofrece es LDA. Pensamos que esto puede deberse a que los *i-vectors* de *development* no estén lo suficientemente separados (por las locuciones utilizadas, por el proceso de extracción de *i-vectors*, etc.) y LDA sea capaz de explotar esa situación mejor que PLDA.

Una vez obtenido el sistema de referencia con las etiquetas de las locuciones de *development* conocidas, el objetivo es intentar alcanzar rendimientos similares a los así obtenidos pero sin hacer uso de dichas etiquetas de locutor.

### 5.3.2 AHC conociendo el número de locutores

En este apartado, se ha fijado el número exacto de locutores conocido (N=1775) presente en el conjunto de datos de *development* sin hacer uso de las etiquetas de locutor. Es decir, el criterio de parada utilizado ha sido el máximo número de *clusters* a generar.

El experimento se ha realizado para dos medidas de distancia (distancia coseno y distancia euclídea entre *i-vectors* normalizados a longitud unidad) y para cuatro métodos de *linkage* (UPGMA, WPGMA, FD y SD), detallados en la secciones 3.2.1 y 3.2.2 de este documento, respectivamente. Los resultados de este experimento se presentan en las Tablas 3 y 4. Destacar que, de aquí en adelante, no se presentarán en las tablas los resultados obtenidos mediante la técnica CSS por no variar su rendimiento (no hace uso de etiquetas de locutor).

| Sistema | | | Class Imp. | Cluster Imp. | EER(%) | minDCF | minCllr |
|---|---|---|---|---|---|---|---|
| Scoring | Distancia | Linkage | | | | | |
| LDA | coseno | UGPMA | 0,25 | 0,47 | **6,93** | **0,472** | **0,220** |
| | | WPGMA | 0,3 | 0,43 | 6,95 | 0,504 | 0,226 |
| | | SD | 0,07 | 0,92 | 14,36 | 0,659 | 0,449 |
| | | FD | 0,42 | 0,45 | 6,91 | 0,511 | 0,224 |
| PLDA (Lnorm) | coseno | UGPMA | 0,25 | 0,47 | 10,94 | 0,561 | 0,291 |
| | | WPGMA | 0,3 | 0,43 | 14,36 | 0,562 | 0,321 |
| | | SD | 0,07 | 0,92 | 12,35 | 0,571 | 0,332 |
| | | FD | 0,42 | 0,45 | 13,75 | 0,577 | 0,309 |
| PLDA (Lnorm + Snorm) | coseno | UGPMA | 0,25 | 0,47 | 7,73 | 0,528 | 0,259 |
| | | WPGMA | 0,3 | 0,43 | 8,73 | 0,536 | 0,283 |
| | | SD | 0,07 | 0,92 | 9,79 | 0,559 | 0,296 |
| | | FD | 0,42 | 0,45 | 7,79 | 0,553 | 0,267 |

**Tabla 3.** Resultados obtenidos al fijar el número máximo de *clusters* (N=1775) con la distancia coseno.

| Sistema | | | Class Imp. | Cluster Imp. | EER(%) | minDCF | minCllr |
|---|---|---|---|---|---|---|---|
| Scoring | Distancia | Linkage | | | | | |
| LDA | euclídea* | UGPMA | 0,24 | 0,47 | 7,13 | **0,480** | 0,231 |
| | | WPGMA | 0,3 | 0,43 | **6,26** | 0,497 | **0,217** |
| | | SD | 0,07 | 0,92 | 14,71 | 0,659 | 0,448 |
| | | FD | 0,41 | 0,45 | 6,90 | 0,513 | 0,226 |
| PLDA (Lnorm) | euclídea* | UGPMA | 0,24 | 0,47 | 11,55 | 0,547 | 0,311 |
| | | WPGMA | 0,3 | 0,43 | 12,75 | 0,551 | 0,319 |
| | | SD | 0,07 | 0,92 | 12,55 | 0,600 | 0,360 |
| | | FD | 0,41 | 0,45 | 13,71 | 0,575 | 0,307 |
| PLDA (Lnorm + Snorm) | euclídea* | UGPMA | 0,24 | 0,47 | 8,13 | 0,522 | 0,272 |
| | | WPGMA | 0,3 | 0,43 | 7,53 | 0,534 | 0,278 |
| | | SD | 0,07 | 0,92 | 9,54 | 0,568 | 0,319 |
| | | FD | 0,41 | 0,45 | 7,80 | 0,551 | 0,264 |

* La distancia euclídea se calcula sobre i-vectors normalizados a longitud unidad.

**Tabla 4.** Resultados obtenidos al fijar el número máximo de *clusters* (N=1775) con la distancia euclídea.

La tendencia de los sistemas sigue al de referencia (*baseline*, Tabla 2), ofreciendo el mejor rendimiento LDA, aunque lejos de los obtenidos al hacer uso de las etiquetas de locutor. Haciendo uso de las etiquetas de locutor a posteriori para comprobar el rendimiento del algoritmo AHC, encontramos que los sistemas que encuentran estructuras de locutores más parecidas a las reales utilizan los métodos de *linkage* UPGMA y WPGMA, tanto para la distancia coseno como para la distancia euclídea sobre *i-vectors* normalizados a longitud unidad.

### *Gaussianización de i-vectors* previa al *clustering*

Como se describe en la sección 4.3, es común realizar una *Gaussianización* de la distribución de los *i-vectors* a través de una etapa de *whitening* y normalización de longitud en técnicas como LDA. Por este motivo, se decidió realizar la misma transformación como paso previo al *clustering* para ver si facilitaba su tarea.

Los resultados obtenidos con esta modificación y fijando el número de locutores (N=1775) se presentan en las Tablas 5 y 6, para distancias coseno y euclídea y métodos

de *linkage* UPGMA y WPGMA, dado que son los que mejor rendimiento ofrecen en el experimento anterior (Tablas 3 y 4).

| Sistema | | | Class Imp. | Cluster Imp. | EER(%) | minDCF | minCllr |
|---|---|---|---|---|---|---|---|
| Scoring | Distancia | Linkage | | | | | |
| LDA | coseno | UGPMA | 0,19 | 0,33 | 6,32 | **0,465** | **0,201** |
| | | WPGMA | 0,23 | 0,33 | **5,92** | 0,471 | 0,203 |
| PLDA (Lnorm) | coseno | UGPMA | 0,19 | 0,33 | 12,81 | 0,545 | 0,289 |
| | | WPGMA | 0,23 | 0,33 | 12,55 | 0,548 | 0,277 |
| PLDA (Lnorm + Snorm) | coseno | UGPMA | 0,19 | 0,33 | 7,49 | 0,516 | 0,249 |
| | | WPGMA | 0,23 | 0,33 | 6,52 | 0,500 | 0,223 |

**Tabla 5.** Resultados obtenidos al fijar el número máximo de *clusters* (N=1775) para distancia coseno sobre *i-vectors* previamente Gaussianizados.

| Sistema | | | Class Imp. | Cluster Imp. | EER(%) | minDCF | minCllr |
|---|---|---|---|---|---|---|---|
| Scoring | Distancia | Linkage | | | | | |
| LDA | euclídea* | UGPMA | 0,19 | 0,33 | **6,12** | **0,475** | **0,203** |
| | | WPGMA | 0,23 | 0,33 | 6,79 | 0,476 | 0,211 |
| PLDA (Lnorm) | euclídea* | UGPMA | 0,19 | 0,33 | 12,62 | 0,538 | 0,275 |
| | | WPGMA | 0,23 | 0,33 | 13,35 | 0,529 | 0,298 |
| PLDA (Lnorm + Snorm) | euclídea* | UGPMA | 0,19 | 0,33 | 7,13 | 0,514 | 0,236 |
| | | WPGMA | 0,23 | 0,33 | 7,64 | 0,518 | 0,263 |

* La distancia euclídea se calcula sobre i-vectors normalizados a longitud unidad.

**Tabla 6.** Resultados obtenidos al fijar el número máximo de *clusters* (N=1775) para distancia euclídea sobre *i-vectors* previamente Gaussianizados.

Al realizar una *Gaussianización* de la distribución de los *i-vectors*, el rendimiento del *clustering* mejora considerablemente (un 24% de mejora relativa promedio en impureza de clase y un 26% de mejora relativa promedio en impureza de *cluster*) en todos los casos. Con ello, mejora también el rendimiento de los sistemas de reconocimiento de locutor respecto al experimento anterior (Tabla 3) en casi todos los casos, tanto en EER(%) como en minDCF y minCllr.

### 5.3.3 AHC en base a la máxima distancia

Este experimento consiste en el entrenamiento de los sistemas de reconocimiento de locutor sin hacer uso ni del conocimiento del número de locutores existentes ni de las etiquetas de locutor. Para ello, se utiliza como criterio de parada la máxima distancia entre los *clusters* a ser unidos, estimada como se menciona en la sección 4.4.1 de este documento.

De nuevo se han utilizado las distancias coseno y euclídea y los métodos de *linkage* UPGMA y WPGMA. En la Tablas 7 y 8 se presentan los resultados de los sistemas de reconocimiento de locutor cuando el *clustering* está trabajando en el punto de cruce de impurezas.

| Sistema | | | Class Imp. | Cluster Imp. | EER(%) | minDCF | minCllr |
|---|---|---|---|---|---|---|---|
| Scoring | Distancia | Linkage | | | | | |
| LDA | coseno | UGPMA (0,76){3396} | 0,29 | 0,29 | **7,09** | 0,494 | **0,226** |
| | | WPGMA (0,81) {2496} | 0,33 | 0,33 | 7,13 | **0,481** | 0,236 |
| PLDA (Lnorm) | coseno | UGPMA (0,76){3396} | 0,29 | 0,29 | 10,41 | 0,538 | 0,319 |
| | | WPGMA (0,81) {2496} | 0,33 | 0,33 | 9,94 | 0,555 | 0,320 |
| PLDA (Lnorm + Snorm) | coseno | UGPMA (0,76){3396} | 0,29 | 0,29 | 8,85 | 0,537 | 0,284 |
| | | WPGMA (0,81) {2496} | 0,33 | 0,33 | 8,73 | 0,537 | 0,286 |

**Tabla 7.** Resultados obtenidos al utilizar la máxima distancia coseno (entre paréntesis) como criterio de parada. Entre llaves se detalla el número de *clusters* creados.

| Sistema | | | Class Imp. | Cluster Imp. | EER(%) | minDCF | minCllr |
|---|---|---|---|---|---|---|---|
| Scoring | Distancia | Linkage | | | | | |
| LDA | euclídea* | UGPMA (1,23){3479} | 0,29 | 0,29 | 6,93 | **0,485** | **0,216** |
| | | WPGMA (1,27) {2547} | 0,32 | 0,33 | **6,43** | 0,492 | 0,222 |
| PLDA (Lnorm) | euclídea* | UGPMA (1,23){3479} | 0,29 | 0,29 | 9,56 | 0,532 | 0,312 |
| | | WPGMA (1,27) {2547} | 0,32 | 0,33 | 10,94 | 0,532 | 0,326 |
| PLDA (Lnorm + Snorm) | euclídea* | UGPMA (1,23){3479} | 0,29 | 0,29 | 8,73 | 0,523 | 0,270 |
| | | WPGMA (1,27) {2547} | 0,32 | 0,33 | 8,93 | 0,534 | 0,297 |

* La distancia euclídea se calcula sobre i-vectors normalizados a longitud unidad.

**Tabla 8.** Resultados obtenidos al utilizar la máxima distancia euclídea (entre paréntesis) como criterio de parada. Entre llaves se detalla el número de *clusters* creados.

De nuevo, el rendimiento de los sistemas de reconocimiento de locutor continua con la misma tendencia que el sistema de referencia, siendo mejor LDA. Sin embargo, el rendimiento del mejor sistema sigue lejano al sistema de referencia, con una degradación relativa promedio (EER, minDCF y minCllr) cercana al 13%.

Para ilustrar el proceso de obtención de los rangos útiles de distancia máxima, obtenidos mediante grafos de K-vecinos (sección 4.4.1) y su relación con el punto de cruce de impurezas, se representan a continuación los utilizados para la realización de este experimento así como las impurezas obtenidas a posteriori. Para la medida de distancia coseno:

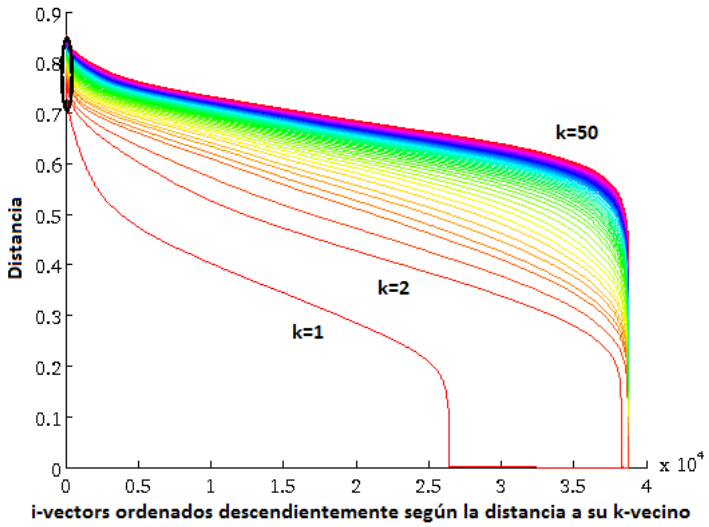


**Figura 20**. Representación gráfica de las distancia coseno de los *i-vectors* a sus K=50 vecinos más próximos ordenadas descendientemente.

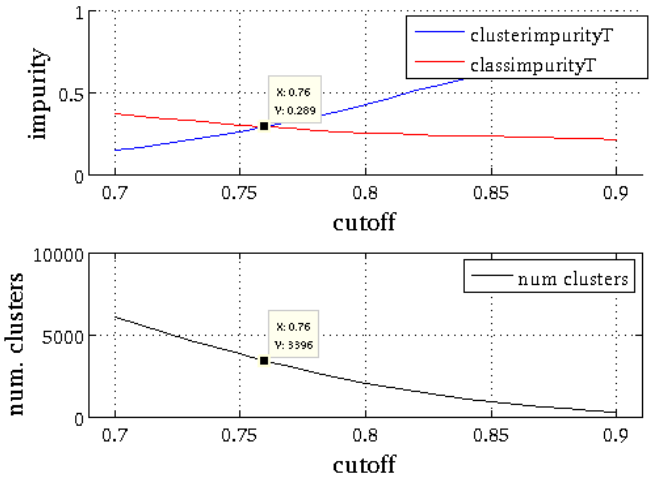


**Figura 21.** Representación gráfica de las impurezas obtenidas del *clustering* con distancia coseno, criterio de parada distancia máxima entre 0.7 y 0.9 y método de *linkage* UPGMA.

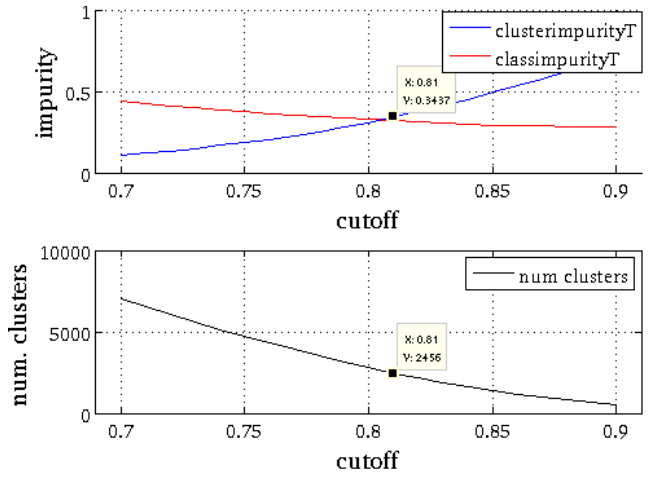


**Figura 22.** Representación gráfica de las impurezas obtenidas del *clustering* con distancia coseno, criterio de parada distancia máxima entre 0.7 y 0.9 y método de *linkage* WPGMA.

Repitiendo el mismo proceso para la distancia euclídea:

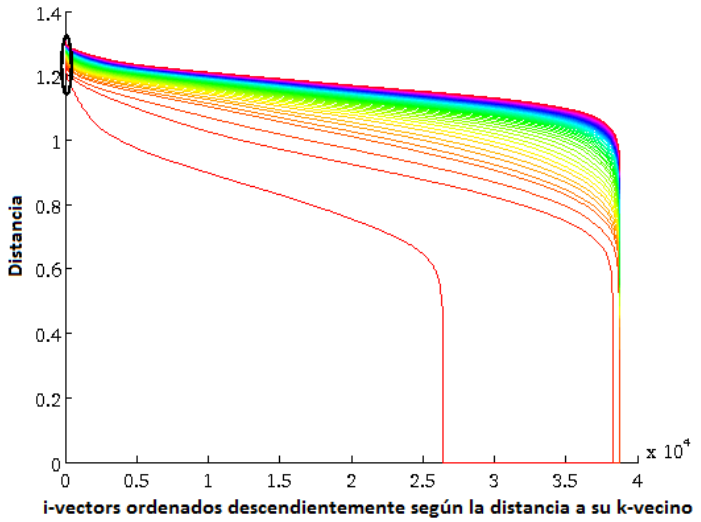


**Figura 23.** Representación gráfica de las distancia euclídea de los *i-vectors* (normalizados a longitud unidad)a sus K=50 vecinos más próximos ordenadas descendientemente.

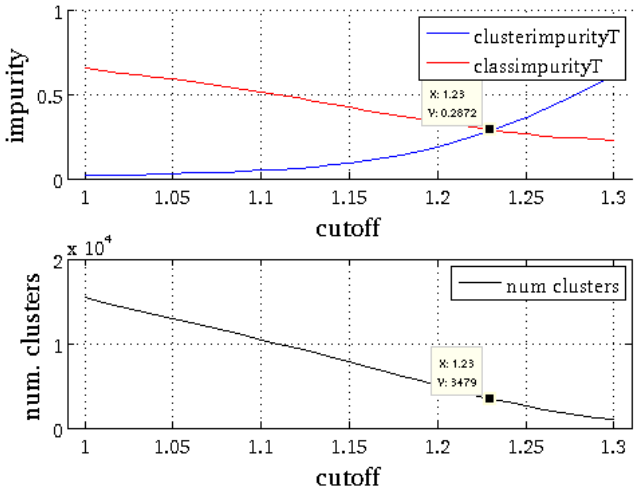


**Figura 24**. Representación gráfica de las impurezas obtenidas del *clustering* con distancia euclídea sobre *i-vectors* normalizados a longitud unidad, criterio de parada distancia máxima entre 1.0 y 1.3 y método de *linkage* UPGMA.

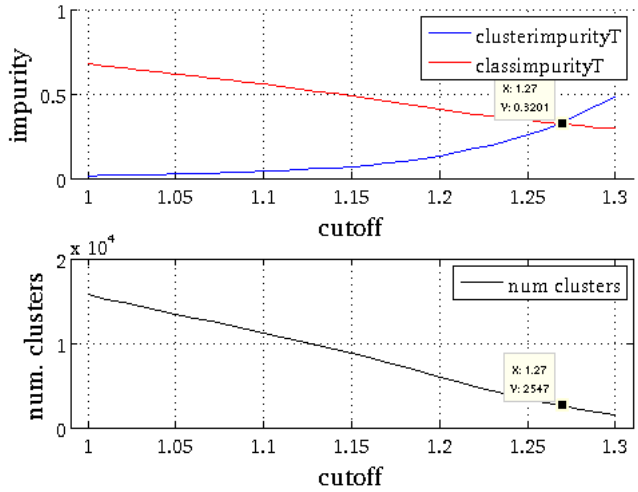


**Figura 25.** Representación gráfica de las impurezas obtenidas del *clustering* con distancia euclídea sobre *i-vectors* normalizados a longitud unidad, criterio de parada distancia máxima entre 1.0 y 1.3 y método de *linkage* WPGMA.

### *Gaussianización de i-vectors* previa al *clustering*

De nuevo, se lleva a cabo una *Gaussianización* de los *i-vectors* previa al *clustering*. Los resultados de este experimento se presentan en las Tablas 9 y 10.

| Sistema | | | Class Imp. | Cluster Imp. | EER(%) | minDCF | minCllr |
|---|---|---|---|---|---|---|---|
| Scoring | Distancia | Linkage | | | | | |
| LDA | coseno | UGPMA (0,91){2669} | 0,21 | 0,21 | 5,89 | 0,467 | 0,198 |
| | | WPGMA (0,93) {2338} | 0,24 | 0,24 | **5,72** | **0,465** | **0,189** |
| PLDA (Lnorm) | coseno | UGPMA (0,91){2669} | 0,21 | 0,21 | 12,37 | 0,535 | 0,295 |
| | | WPGMA (0,93) {2338} | 0,24 | 0,24 | 9,34 | 0,500 | 0,272 |
| PLDA (Lnorm + Snorm) | coseno | UGPMA (0,91){2669} | 0,21 | 0,21 | 6,71 | 0,522 | 0,249 |
| | | WPGMA (0,93) {2338} | 0,24 | 0,24 | 7,22 | 0,500 | 0,232 |

**Tabla 9.** Resultados obtenidos al utilizar la máxima distancia coseno (entre paréntesis) como criterio de parada con etapa previa de *Gaussianización* de la distribución de *i-vectors*. Entre llaves se detalla el número de *clusters* creados.

| Sistema | | | Class Imp. | Cluster Imp. | EER(%) | minDCF | minCllr |
|---|---|---|---|---|---|---|---|
| Scoring | Distancia | Linkage | | | | | |
| LDA | euclídea* | UGPMA (1,35){2563} | 0,21 | 0,22 | 6,13 | 0,479 | 0,201 |
| | | WPGMA (1,36) {2410} | 0,24 | 0,24 | **5,92** | **0,466** | **0,195** |
| PLDA (Lnorm) | euclídea* | UGPMA (1,35){2563} | 0,21 | 0,22 | 12,34 | 0,522 | 0,296 |
| | | WPGMA (1,36) {2410} | 0,24 | 0,24 | 11,54 | 0,511 | 0,293 |
| PLDA (Lnorm + Snorm) | euclídea* | UGPMA (1,35){2563} | 0,21 | 0,22 | 6,93 | 0,496 | 0,242 |
| | | WPGMA (1,36) {2410} | 0,24 | 0,24 | 7,08 | 0,503 | 0,250 |

* La distancia euclídea se calcula sobre i-vectors normalizados a longitud unidad.

**Tabla 10.** Resultados obtenidos al utilizar la máxima distancia euclídea (entre paréntesis) como criterio de parada con etapa previa de *Gaussianización* de la distribución de *i-vectors*. Entre llaves se detalla el número de *clusters* creados.

### 5.3.4 Inclusión del criterio silhouette

Por último, se hace uso del criterio *silhouette* (véase sección 3.3.4) para eliminar aquellos *i-vectors* que, al finalizar el *clustering*, probablemente no hayan quedado asignados correctamente a su *cluster*.

Haciendo uso de este criterio, se eliminan cantidades próximas a 100, 500, 1500 y 3000 *i-vectors* a partir de valores de *silhouette* negativos y próximos a 0, consiguiendo así mejorar el rendimiento de los sistemas de reconocimiento de locutor en la mayoría de los casos.

En las Tablas 11 y 12 se presentan los resultados al eliminar *i-vectors* basándonos en el coeficiente *silhouette* correspondientes al *clustering* con distancia coseno y métodos de *linkage* UPGMA y WPGMA, respectivamente.

| Sistema | | | | EER(%) | minDCF | minCllr |
|---|---|---|---|---|---|---|
| Scoring | Distancia | Linkage | Silhouette | | | |
| LDA | coseno | UGPMA (0,91){2669} | - | 5,89 | 0,467 | 0,198 |
| | | | S < 0 (-3006 i-vectors) | 5,93 | 0,462 | 0,199 |
| | | | S < -0.02 (-1589 i-vectors) | **5,72** | **0,454** | **0,194** |
| | | | S < -0.05 (-518 i-vectors) | 5,72 | 0,465 | 0,194 |
| | | | S < -0.09 (-98 i-vectors) | 5,91 | 0,467 | 0,196 |
| PLDA (Lnorm) | coseno | UGPMA (0,91){2669} | - | 12,37 | 0,535 | 0,295 |
| | | | S < 0 (-3006 i-vectors) | 9,94 | 0,505 | 0,262 |
| | | | S < -0.02 (-1589 i-vectors) | 10,94 | 0,504 | 0,268 |
| | | | S < -0.05 (-518 i-vectors) | 11,54 | 0,511 | 0,279 |
| | | | S < -0.09 (-98 i-vectors) | 12,34 | 0,530 | 0,293 |
| PLDA (Lnorm + Snorm) | coseno | UGPMA (0,91){2669} | - | 6,71 | 0,522 | 0,249 |
| | | | S < 0 (-3006 i-vectors) | 6,72 | 0,473 | 0,228 |
| | | | S < -0.02 (-1589 i-vectors) | 6,52 | 0,492 | 0,220 |
| | | | S < -0.05 (-518 i-vectors) | 6,65 | 0,494 | 0,237 |
| | | | S < -0.09 (-98 i-vectors) | 6,72 | 0,513 | 0,249 |

**Tabla 11.** Resultados obtenidos incluyendo el criterio *silhouette* (S) para eliminar *i-vectors* que probablemente hayan quedado mal asignados entre su *cluster* para distancia máxima coseno y *linkage* UPGMA con etapa previa de *Gaussianización* de la distribución de *i-vectors*.

| Sistema | | | | EER(%) | minDCF | minCllr |
|---|---|---|---|---|---|---|
| Scoring | Distancia | Linkage | Silhouette | | | |
| LDA | coseno | WGPMA (0,93){2338} | - | 5,72 | 0,465 | 0,189 |
| | | | S < 0 (-3465 i-vectors) | 6,52 | 0,469 | 0,198 |
| | | | S < -0.026 (-1531 i-vectors) | 6,01 | 0,465 | 0,193 |
| | | | S < -0.055 (-509 i-vectors) | 5,72 | 0,459 | 0,192 |
| | | | S < -0.095 (-107 i-vectors) | **5,72** | **0,459** | **0,189** |
| PLDA (Lnorm) | coseno | WGPMA (0,93){2338} | - | 9,34 | 0,500 | 0,272 |
| | | | S < 0 (-3465 i-vectors) | 8,13 | 0,506 | 0,253 |
| | | | S < -0.026 (-1531 i-vectors) | 7,33 | 0,508 | 0,257 |
| | | | S < -0.055 (-509 i-vectors) | 7,93 | 0,506 | 0,260 |
| | | | S < -0.095 (-107 i-vectors) | 9,34 | 0,499 | 0,272 |
| PLDA (Lnorm + Snorm) | coseno | WGPMA (0,93){2338} | - | 7,47 | 0,499 | 0,232 |
| | | | S < 0 (-3465 i-vectors) | 7,16 | 0,502 | 0,216 |
| | | | S < -0.026 (-1531 i-vectors) | 6,33 | 0,502 | 0,221 |
| | | | S < -0.055 (-509 i-vectors) | 7,08 | 0,504 | 0,225 |
| | | | S < -0.095 (-107 i-vectors) | 7,21 | 0,496 | 0,230 |

**Tabla 12.** Resultados obtenidos incluyendo el criterio *silhouette* (S) para eliminar *i-vectors* que probablemente hayan quedado mal asignados entre su *cluster* para distancia máxima coseno y *linkage* WPGMA con etapa previa de *Gaussianización* de la distribución de *i-vectors*.

La inclusión del criterio *silhouette* para eliminar *i-vectors* mejora los sistemas de reconocimiento de locutor en muchos de los casos, logrando resultados similares a los obtenidos por el sistema que si hacía uso de las etiquetas de locutor (*baseline*). Para evaluar con mayor facilidad los resultados obtenidos, en la Tabla 13 se presenta la comparación de los más relevantes.

| Sistema | Baseline | | | Agglomerative Hiearchical Clustering (AHC) | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | | | | Distancia | Linkage | Silhouette | Distancia | Linkage | Silhouette |
| | | | | coseno | UPGMA | S < -0.02 | coseno | WPGMA | S < -0.095 |
| | EER (%) | minDCF | minCllr | EER (%) | minDCF | minCllr | EER (%) | minDCF | minCllr |
| CSS | 7,13 | 0,510 | 0,233 | 7,13 | 0,510 | 0,233 | 7,13 | 0,510 | 0,233 |
| LDA | **5,61** | **0,451** | **0,187** | **5,72** | **0,454** | **0,194** | **5,72** | **0,459** | **0,189** |

**Tabla 13.** Recopilación de los resultados obtenidos más relevantes.

Si se hace uso de etiquetas de locutor (*baseline*) para aprovechar técnicas de compensación de variabilidad, el mejor resultado se obtiene con LDA (EER(%)=5.61, minDCF=0.451, minCllr=0.187). En caso de no hacer uso de etiquetas de locutor ni utilizar técnicas de *clustering* que nos permitan aprovechar tecnologías como LDA o PLDA, el sistema sería CSS (EER(%)=7.13, minDCF=0.510, minCllr=0.233), lejos del rendimiento de LDA.

Sin embargo, si se utilizan técnicas de *clustering* (AHC) para encontrar agrupaciones de locuciones que permitan entrenar técnicas de compensación de variabilidad, se puede llegar a soluciones con rendimientos muy similares a los obtenidos cuando si se hace uso de etiquetas de locutor. En nuestros experimentos, respecto al mejor sistema de referencia (LDA), se obtienen rendimientos con degradaciones promedio (EER, minDCF y minCllr) del 2,06% para distancia coseno y método de *linkage* UPGMA y del 1,53% para distancia coseno y método de *linkage* WPGMA, en sus mejores versiones.

Por tanto, es posible llegar a soluciones con rendimientos semejantes a los que se obtienen al hacer uso de etiquetas de locutor cuando no se dispone de éstas.

# 6. Conclusiones y trabajo futuro

En este proyecto se trata uno de los puntos débiles de las tecnologías de compensación de variabilidad (como LDA y PLDA) que se utilizan comúnmente en los sistemas de reconocimiento de locutor: la necesidad de etiquetas de identidad de locutor. Cuando las bases de datos utilizadas en el sistema tienen el orden de decenas de miles o superior, el etiquetado manual de éstas es una tarea muy costosa que puede llegar a hacer un proyecto inviable.

Como solución a este problema se presenta la extracción automática de etiquetas de locutor mediante la técnica de *clustering* AHC aplicada sobre *i-vectors*. De los experimentos realizados se pueden extraer las siguientes conclusiones:

- Sin hacer uso de etiquetas de identidad de locutor, el único sistema posible (CSS) tiene los siguientes rendimientos: EER(%)=7.13, minDCF=0.510, y minCllr=0.233, muy lejos de los obtenidos por el mejor sistema que si hace uso de etiquetas de locutor (LDA, EER(%)=5.61, minDCF=0.451, minCllr=0.187). Por tanto, las técnicas de compensación de variabilidad aportan una mejora de rendimiento al sistema de reconocimiento de locutor que hay que intentar aprovechar.

- Mediante AHC se pueden obtener etiquetas de identidad de locutor que permitan utilizar técnicas de compensación de variabilidad, como LDA, en caso de no estar disponibles.

- En el mejor sistema que hace uso de AHC para estimar las etiquetas de identidad de locutor, se obtiene una degradación de rendimiento promedio (EER, minDCF y minCllr) del 1,53% respecto al mejor sistema que sí hace uso de las etiquetas de identidad de locutor reales.

Pese a que los resultados obtenidos son satisfactorios y cumplen los objetivos marcados en este trabajo, quedan algunos interrogantes por resolver en el futuro:

- ¿Es la solución de *clustering* (AHC) propuesta, para obtener las etiquetas de identidad de locutor, la que mejores rendimientos puede ofrecer?

- Todo el estudio se enfoca a la obtención de etiquetas de identidad de locutor que permitan estimar los subespacios de variabilidad de técnicas como LDA o PLDA de manera similar a como lo harían las etiquetas reales. Sin embargo, ¿son las etiquetas reales el objetivo?¿Pueden existir otras agrupaciones de locuciones ocultas que permitan estimar dichos subespacios mejor que las etiquetas reales?

# Referencias bibliográficas


**[Atal, 1974]**
B. Atal. "*Effectiveness of linear prediction characteristics of the speech wave for automatic speaker identification and verification*." *J. Acoust. Soc. Amer*. 55 (6), 1304-1312, 1974.

**[Auckenthaler *et al*., 2000]**
R. Auckenthaler, M. Carey and H. Lloyd-Thomas. "*Score normalization for text-independent speaker verification systems.*" *Digital Signal Processing*, V10, pp. 42-54, 2000.

**[Bimbot *et al*., 2004]**
F.Bimbot, J.-F. Bonastre, C. Fredouille, G. Gravier, I. Margin-Chagnolleau, S. Meignier, T. Merlin, J. Ortega-Garcia, D. Petrovska-Delacretaz, and D. A. Reynolds. A tutorial on text-independent speaker verification. *Journal on Applied Signal Processing*, 2004(4):430-451, 2004.

**[Brümmer and Preez, 2006]**
N. Brümmer and J. du Preez. "*Application-Independent Evaluation of Speaker Detection*". Computer Speech & Language, 20(2-3):230–275, 2006.

**[Brümmer *et al*., 2007]**
N. Brümmer, L. Burget, J. Cernocky, O. Glembek, F. Grezl, M. Karafiat, D. A. van Leeuwen, P. Matejka, P. Scwartz, and A. Strasheim. Fusion of Heterogeneous Speaker Recognition Systems in the STBU Submission for the NIST Speaker Recognition Evaluation 2006. IEEE Transactions on Audio, Speech and Signal Processing, 15(7):2072–2084, 2007.

**[Burget *et al*., 2007]**
L. Burget, P. Matejka, O. Glembek and J. Cernocky. "*Analysis of feature extraction and channel compensation in a GMM speaker recognition system.*" *IEEE Trans. Audio, Speech Language Process*. 15 (7), 1979-1986, 2007.

**[Burton, 1987]**
D. Burton. "*Text-independent speaker verificaction using vector quantization source coding.*" *IEEE Trans. Acoustics, Speech, Signal Process.* 35 (2), 133-143, 1987.

**[Calinski and Harabasz, 1974]**
Calinski, T., and J. Harabasz. "*A dendrite method for cluster analysis*." *Communications in Statistics*. Vol. 3, No. 1, 1974, pp. 1–27.

**[Campbell *et al*., 2006a]**
W. Campbell, J. Campbell, D. Reynolds, E. Singer, P. Torres-Carrasquillo. "*Support vector machines for speaker and language recognition.*" Comput. Speech Lang. 20 (2-3), 210-229, 2006.

**[Carr, 1999]**
P. Carr. English Phonetics and Phonology: An Introduction. *Blackwell Publishing*, 1999.

**[Dehak *et al*, 2011]**
Dehak, N., et al., "*Front-End Factor Analysis for Speaker Verification*", IEEE Trans. on Audio, Speech, and Lang. Proc., 19(4), 788-798, May 2011.

**[Davies and Bouldin, 1979]**
Davies, D. L., and D. W. Bouldin. "*A Cluster Separation Measure*." *IEEE Transactions on Pattern Analysis and Machine Intelligence*. Vol. PAMI-1, No. 2, 1979, pp. 224–227.

**[Davis and Mermelstein, 1980]**
S. Davis and P. Mermelstein. "*Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences.*" *IEEE Trans. Acoustics, Speech, Signal Process.* 28 (4), 357-366, 1980.

**[Doddington, and 2001]**
G. R. Doddington, "*Speaker recognition based on idiolectal differences between speakers.," in Proc. of Interspeech,* 2001.

**[Fierrez-Aguilar *et al*., 2003]**
J. Fierrez-Aguilar, J. Ortega-García, D. García-Romero y J. González-Rodríguez. "*A comparative evaluation of fusion strategies for multimodal biometric verification.*" *Proc. 4th IAPR Intl. Conf. on Audio and Video Based Person Authentication AVBPA*, pp. 830-837, Junio 2003.

**[Fierrez-Aguilar *et al*., 2005]**
J. Fierrez-Aguilar, J. Ortega-Garcia, and J. Gonzalez-Rodriguez. "*Target dependent score normalization techniques and their application to signature verification.*" *IEEE Trans. on Systems, Man and Cybernetics, part C*, 35(3):418:425, 2005.

**[Furui, 1981]**
S. Furui. "*Cepstral analysis technique for automatic speaker verification.*" *IEEE Trans. Acoustics, Speech Signal Process.* 29 (2), 254-272, 1981.

**[Gersho and Gray, 1991]**
A. Gersho and R. Gray. "*Vector Quantization and Signal Compression.*" *Kluwer Academic Publishers*, Boston, 1991.

**[Haigh and Mason, 1993]**
J. Haigh and J. Mason, "*Robust voice activity detection using cepstral features,*" *in IEEE Region 10 Conference on Computer, Communication, Control and Power Engineering* (TENCON'93), no. 0, Oct 1993, pp.321-324.

**[Han and Narayanan, 2007]**
Kyu Jeong Han, and Shrikanth S. Narayanan, "*A robust stopping criterion for agglomerative hierarchical clustering in a speaker diarization system," INTERSPEECH, page 1853-1856. ISCA,* (2007).

**[Hermanski and Morgan, 1994]**
H. Hermansky and N. Morgan. RASTA processing of speech. *IEEE Trans. Speech Audio Process.* 2 (4), 578-589.

**[Huang *et al*., 2001]**
X. Huang, A. Acero and H.-W. Hon. "*Spoken Language Processing: a Guide to Theory, Algorithm, and System Development.*" *Prentice-Hall*, New Jersey, 2001.

**[Kenny, 2006]**
P. Kenny. "*Joint factor analysis of speaker and session variability: theory and algorithms*". *Technical Report* CRIM-06/08-14, 2006.

**[Kenny *et al.*, 2008]**
P. Kenny, P. Ouellet, N. Dehak, V. Grupta and P. Dumouchel. "*A study of inter-speaker variability in speaker verification*". *IEEE Trans. Audio, Speech Language Process.* 16 (5), 980-988, 2008.

**[Kinnunen *et al.*, 2006]**
Kinnunen, T., Karpov, E., Fränti, P., 2006. "*Real-time speaker identification and verification.*" IEEE Trans. Audio, Speech Language Process. 14 (1), 277–288.

**[Kinnunen and Li, 2010]**
Kinnunen, T., and Li, H., "*An overview of text-independent speaker recognition: from features to supervectors*", Speech Communication, vol. 52, pp. 12-40, 2010.

**[Leeuwen, 2010]**
D. van Leeuwen, "Speaker linking in large data sets," in *Odyssey: The Speaker and Language Recognition Workshop*, Brno, Czech Republic, June 2010.

**[Leeuwen and Brümmer, 2007]**
D. A. Leeuwen and N. Brümmer. "*An introduction to application-independent evaluation of speaker recognition systems.*" In C. Müller, editor, Speaker Classification I, pages 330–353. Springer-Verlag, Berlin, Heidelberg, 2007.

**[Linde *et al*., 1980]**
Y. Linde, A. Buzo, R. Gray. "*An algorithm for vector quantizer design.*" *IEEE Trans. Comm.* 28 (1), 84-95.

**[Li and Porter, 1988]**
K. P. Li and J. E. Porter. "*Normalizations and selection of speech segments for speaker recognition scoring.*" In *Proc. of ICASSP*, páginas 595-598, New York, NY, USA, 1988.

**[López-Moreno *et al*., 2008]**
I. Lopez-Moreno, D. Ramos, J. Gonzalez-Rodriguez, and D. T. Toledano. "*Anchor Model Fusion for LanguageRecognition.*" In Proceedings of Interspeech 2008, September 2008.

**[López *et al*., 2003]**
E. López, G. Sosa y M. Rocamora: "*Tratamiento de Voz*". Disponible en: http://iie.fing.edu.uy/investigacion/grupos/gmm/audio/seminario/seminariosviejos/2003/charlas/charla1/voz8.htm

**[Louradour and Daoudi, 2005]**
Louradour, J., Daoudi, K., 2005. "*SVM speaker verification using a new sequence kernel.*" In: Proc. 13th European Conf. on Signal Processing (EUSIPCO 2005), Antalya, Turkey, September2005.

**[Malayath *et al*., 2000]**
N. Malayath, H. Hermansky, S. Kajarekar and B. Yegnanarayana. "*Data-driven temporal filters and alternatives to GMM in speaker verification.*" *Digital Signal Process.* 10 (1-3), 55-74, 2000.

**[Maltoni *et al.*, 2003]**
D. Maltoni, D. Maio, A. K. Jain and S. Prabhakar. Handbook of Fingerprint Recognition, *Springer* 2003.

**[NIST evaluation plan, 2012]**
NIST SRE 2012 evaluation plan, available at
http://www.nist.gov/itl/iad/mig/upload/NIST_SRE12_evalplan-v17-r1.pdf

**[NIST i-vector Challenge, 2014]**
NIST i-vector Machine Learning Challenge evaluation plan, available at
http://www.nist.gov/itl/iad/mig/upload/sre-ivectorchallenge_2013-11-18_r0.pdf

**[NIST SRE]**
Página web de las evaluaciones NIST de reconocimiento de locutor: http://www.nist.gov/itl/iad/mig/sre.cfm

**[Oppenheim *et al*., 1999]**
A. Oppenheim, R. Schafer and J. Buck. Discrete-Time Signal Processing, segunda edición. *Prentice-Hall*, 1999.

**[Pelecanos and Sridharan, 2001]**
J. Pelecanos y S. Sridharan. "*Feature warping for robust speaker verification.*" *Proc. Speaker Odyssey: the Speaker Recognition Workshop (Odyssey 2001)*, Creta, Grecia, Junio de 2001, pp. 213-218.

**[Prince and Elder, 2007]**
S.J.D. Prince and J.H. Elder, "Probabilistic Linear Discriminant Analysis for Inferences about Identity, " *Proc. IEEE 11th Int'l Conf. Computer Vision*, 2007.

**[Reynolds and Rose, 1995]**
D. Reynolds and R. Rose. "*Robust text-independent speaker identification using Gaussian mixture speaker models.*" *IEEE Trans. Speech Audio Process.* 3, 72-83, 1995.

**[Reynolds *et al*., 2000]**
D. A. Reynolds, T. F. Queatieri and R. B. Dunn: "*Speaker Verification Using Adapted Gaussian Mixture Models"*. Digital Sign al Processing 10, 19-41 (2000).

**[Reynolds *et al*., 2003]**
Douglas Reynolds, Walter Andrews, Joseph Campell, Jiri Navratil, Barbara Peskin, Andre Adam, Qin Jin, David Klusacek, Joy Abramson, Radu Mihaescu, Jack Godfrey, Doug Jones, Bing Xiang. SuperSID Project: "*Exploiting high-level information for high accuracy speaker recognition*". *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Vol. 4, pp 784-787, Abril 2003.

**[Reynolds, 2003]**
D. Reynolds. "*Channel robust speaker verification via feature mapping*." *Proc. Internat. Conf. on Acoustics, Speech, and Signal Processing (ICASSP 2003)*, Vol. 2, Hong Kong, China, Abril de 2003, pp. 53-56.

**[Roch, 2006]**
Roch, M., 2006. "*Gaussian-selection-based non-optimal search for speaker identification.*" Speech Commu. 48, 85–95.

**[Romero and Wilson, 2006]**
D. Garcia-Romero and C.Y. Espy-Wilson, "*Analysis of i-vector length normalization in speaker recognition systems,* " in Proc. INTERSPEECH, Florence, Italy, Aug. 2011, pp. 249-252.

**[Rose, 2002]**
P. Rose. "*Forensic Speaker Identification.*" *Taylor & Francis*, London, 2002.

**[Rouseeuw, 1987]**
Rouseeuw, P. J. "Silhouettes: a graphical aid to the interpretation and validation of cluster analysis." *Journal of Computational and Applied Mathematics*. Vol. 20, No. 1, 1987, pp. 53–65.

**[Sohn *et al*., 1999]**
J. Sohn, N. S. Kim and W. Sung, "*A statistical model-based voice activity detection.*" *Signal Processing Letters*, IEEE, vol. 6, no. 1, pp. 1-3, Jan. 1999.

**[Soong *et al*., 1987]**
F. K. Soong, A. E. Rosenberg, B. -H. Juang and L. R. Rabiner. "*A vector quantization approach to speaker recognition*." *AT&T Technical J*. 66, 14-26. 1997.

**[Soong and Rosenberg, 1988]**
F. Soong and A. Rosenberg. "*On the use of instantaneous and transitional spectral information in speaker recognition.*" *IEEE Trans. Acoustics, Speech Signal Process*. 36 (6), 871-879, 1988.

**[Sound eXchange software]**
"Sound eXchange software, " available at *http://sox.sourceforge.net/*

**[Schwarz, 2009]**
P. Schwarz, "*Phoneme recognition based on long temporal context*," Ph.D. dissertation, Brno University of Technology, 2009.

**[Tibshirani *et al*, 2001]**
Tibshirani, R., G. Walther, and T. Hastie. "Estimating the number of clusters in a data set via the gap statistic." *Journal of the Royal Statistical Society: Series B*. Vol. 63, Part 2, 2001, pp. 411–423.

**[Tranter and Reynolds, 2006]**
Sue E. Tranter and Douglas A. Reynolds, "*An overview of automatic speaker diarization systems, " IEEE Transactions on Audio, Speech, and Language Processing*, 2006.

**[Tucker, 1992]**
R. Tucker, "*Voice activity detection using a periodicity measure,"Communications, Speech and Vision*, IEE Proceedings I, vol. 139, no. 4, pp. 377-380, Aug. 1992.

**[Vogt and Sridharan, 2008]**
R. Vogt and S. Sridharan. "*Explicit modeling of session variability in text-independent speaker verification.*" *Comput. Speech Lang.* 22 (1), 17-38, 2008.

**[Wolf, 1972]**
J. Wolf. "*Efficient acoustic parameters for speaker recognition.*" *J. Acoust. Soc. Amer.* 51 (6), 2044-20556 (Part 2), 1972.

# A. Presupuesto

**1) Ejecución Material**

- Compra de ordenador personal (Software incluido)..... ..... 1100 €
- Material de oficina ..... 150 €
- Total de ejecución material..... 1250 €

**2) Gastos generales**

- 16 % sobre Ejecución Material..... 200 €

**3) Beneficio Industrial**

- 6 % sobre Ejecución Material..... 75 €

**4) Honorarios Proyecto**

- 1400 horas a 15 € / hora..... 21000 €

**5) Material fungible**

- Gastos de impresión ..... 100 €
- Encuadernación ..... 100 €
- Total de material fungible.....200 €

**6) Subtotal del presupuesto**

- Subtotal Presupuesto..... 22450 €

**7) I.V.A. aplicable**

- 21% Subtotal Presupuesto..... 4714.5 €

**8) Total presupuesto**

- Total Presupuesto ..... 27164,5 €

Madrid, Junio 2014

El Ingeniero Jefe de Proyecto

Fdo.: Iván Gómez Piris
Ingeniero Superior de Telecomunicación

# B. Pliego de condiciones

Este documento contiene las condiciones legales que guiarán la realización, en este proyecto, de una EXTRACCIÓN DE INFORMACIÓN DE SEÑALES DE VOZ PARA EL AGRUPAMIENTO POR LOCUTORES DE LOCUCIONES ANÓNIMAS.

En lo que sigue, se supondrá que el proyecto ha sido encargado por una empresa cliente a una empresa consultora con la finalidad de realizar dicho sistema. Dicha empresa ha debido desarrollar una línea de investigación con objeto de elaborar el proyecto. Esta línea de investigación, junto con el posterior desarrollo de los programas está amparada por las condiciones particulares del siguiente pliego.

Supuesto que la utilización industrial de los métodos recogidos en el presente proyecto ha sido decidida por parte de la empresa cliente o de otras, la obra a realizar se regulará por las siguientes:

**Condiciones generales**

1. La modalidad de contratación será el concurso. La adjudicación se hará, por tanto, a la proposición más favorable sin atender exclusivamente al valor económico, dependiendo de las mayores garantías ofrecidas. La empresa que somete el proyecto a concurso se reserva el derecho a declararlo desierto.

2. El montaje y mecanización completa de los equipos que intervengan será realizado totalmente por la empresa licitadora.

3. En la oferta, se hará constar el precio total por el que se compromete a realizar la obra y el tanto por ciento de baja que supone este precio en relación con un importe límite si este se hubiera fijado.

4. La obra se realizará bajo la dirección técnica de un Ingeniero Superior de Telecomunicación, auxiliado por el número de Ingenieros Técnicos y Programadores que se estime preciso para el desarrollo de la misma.

5. Aparte del Ingeniero Director, el contratista tendrá derecho a contratar al resto del personal, pudiendo ceder esta prerrogativa a favor del Ingeniero Director, quien no estará obligado a aceptarla.

6. El contratista tiene derecho a sacar copias a su costa de los planos, pliego de condiciones y presupuestos. El Ingeniero autor del proyecto autorizará con su firma las copias solicitadas por el contratista después de confrontarlas.

7. Se abonará al contratista la obra que realmente ejecute con sujeción al proyecto que sirvió de base para la contratación, a las modificaciones autorizadas por la superioridad o a las órdenes que con arreglo a sus facultades le hayan comunicado por escrito al Ingeniero Director de obras siempre que dicha obra se haya ajustado a los preceptos de los pliegos de condiciones, con arreglo a los cuales, se harán las modificaciones y la valoración de las diversas

unidades sin que el importe total pueda exceder de los presupuestos aprobados. Por consiguiente, el número de unidades que se consignan en el proyecto o en el presupuesto, no podrá servirle de fundamento para entablar reclamaciones de ninguna clase, salvo en los casos de rescisión.

8. Tanto en las certificaciones de obras como en la liquidación final, se abonarán los trabajos realizados por el contratista a los precios de ejecución material que figuran en el presupuesto para cada unidad de la obra.

9. Si excepcionalmente se hubiera ejecutado algún trabajo que no se ajustase a las condiciones de la contrata pero que sin embargo es admisible a juicio del Ingeniero Director de obras, se dará conocimiento a la Dirección, proponiendo a la vez la rebaja de precios que el Ingeniero estime justa y si la Dirección resolviera aceptar la obra, quedará el contratista obligado a conformarse con la rebaja acordada.

10. Cuando se juzgue necesario emplear materiales o ejecutar obras que no figuren en el presupuesto de la contrata, se evaluará su importe a los precios asignados a otras obras o materiales análogos si los hubiere y cuando no, se discutirán entre el Ingeniero Director y el contratista, sometiéndolos a la aprobación de la Dirección. Los nuevos precios convenidos por uno u otro procedimiento, se sujetarán siempre al establecido en el punto anterior.

11. Cuando el contratista, con autorización del Ingeniero Director de obras, emplee materiales de calidad más elevada o de mayores dimensiones de lo estipulado en el proyecto, o sustituya una clase de fabricación por otra que tenga asignado mayor precio o ejecute con mayores dimensiones cualquier otra parte de las obras, o en general, introduzca en ellas cualquier modificación que sea beneficiosa a juicio del Ingeniero Director de obras, no tendrá derecho sin embargo, sino a lo que le correspondería si hubiera realizado la obra con estricta sujeción a lo proyectado y contratado.

12. Las cantidades calculadas para obras accesorias, aunque figuren por partida alzada en el presupuesto final (general), no serán abonadas sino a los precios de la contrata, según las condiciones de la misma y los proyectos particulares que para ellas se formen, o en su defecto, por lo que resulte de su medición final.

13. El contratista queda obligado a abonar al Ingeniero autor del proyecto y director de obras así como a los Ingenieros Técnicos, el importe de sus respectivos honorarios facultativos por formación del proyecto, dirección técnica y administración en su caso, con arreglo a las tarifas y honorarios vigentes.

14. Concluida la ejecución de la obra, será reconocida por el Ingeniero Director que a tal efecto designe la empresa.

15. La garantía definitiva será del 4% del presupuesto y la provisional del 2%.

16. La forma de pago será por certificaciones mensuales de la obra ejecutada, de acuerdo con los precios del presupuesto, deducida la baja si la hubiera.

17. La fecha de comienzo de las obras será a partir de los 15 días naturales del replanteo oficial de las mismas y la definitiva, al año de haber ejecutado la provisional, procediéndose si no existe reclamación alguna, a la reclamación de la fianza.

18. Si el contratista al efectuar el replanteo, observase algún error en el proyecto, deberá comunicarlo en el plazo de quince días al Ingeniero Director de obras, pues transcurrido ese plazo será responsable de la exactitud del proyecto.

19. El contratista está obligado a designar una persona responsable que se entenderá con el Ingeniero Director de obras, o con el delegado que éste designe, para todo relacionado con ella. Al ser el Ingeniero Director de obras el que interpreta el proyecto, el contratista deberá consultarle cualquier duda que surja en su realización.

20. Durante la realización de la obra, se girarán visitas de inspección por personal facultativo de la empresa cliente, para hacer las comprobaciones que se crean oportunas. Es obligación del contratista, la conservación de la obra ya ejecutada hasta la recepción de la misma, por lo que el deterioro parcial o total de ella, aunque sea por agentes atmosféricos u otras causas, deberá ser reparado o reconstruido por su cuenta.

21. El contratista, deberá realizar la obra en el plazo mencionado a partir de la fecha del contrato, incurriendo en multa, por retraso de la ejecución siempre que éste no sea debido a causas de fuerza mayor. A la terminación de la obra, se hará una recepción provisional previo reconocimiento y examen por la dirección técnica, el depositario de efectos, el interventor y el jefe de servicio o un representante, estampando su conformidad el contratista.

22. Hecha la recepción provisional, se certificará al contratista el resto de la obra, reservándose la administración el importe de los gastos de conservación de la misma hasta su recepción definitiva y la fianza durante el tiempo señalado como plazo de garantía. La recepción definitiva se hará en las mismas condiciones que la provisional, extendiéndose el acta correspondiente. El Director Técnico propondrá a la Junta Económica la devolución de la fianza al contratista de acuerdo con las condiciones económicas legales establecidas.

23. Las tarifas para la determinación de honorarios, reguladas por orden de la Presidencia del Gobierno el 19 de Octubre de 1961, se aplicarán sobre el denominado en la actualidad “Presupuesto de Ejecución de Contrata” y anteriormente llamado ”Presupuesto de Ejecución Material” que hoy designa otro concepto.

**Condiciones particulares**

La empresa consultora, que ha desarrollado el presente proyecto, lo entregará a la empresa cliente bajo las condiciones generales ya formuladas, debiendo añadirse las siguientes condiciones particulares:

1. La propiedad intelectual de los procesos descritos y analizados en el presente trabajo, pertenece por entero a la empresa consultora representada por el Ingeniero Director del Proyecto.

2. La empresa consultora se reserva el derecho a la utilización total o parcial de los resultados de la investigación realizada para desarrollar el siguiente proyecto, bien para su publicación o bien para su uso en trabajos o proyectos posteriores, para la misma empresa cliente o para otra.

3. Cualquier tipo de reproducción aparte de las reseñadas en las condiciones generales, bien sea para uso particular de la empresa cliente, o para cualquier otra aplicación,

contará con autorización expresa y por escrito del Ingeniero Director del Proyecto, que actuará en representación de la empresa consultora

4. En la autorización se ha de hacer constar la aplicación a que se destinan sus reproducciones así como su cantidad.

5. En todas las reproducciones se indicará su procedencia, explicitando el nombre del proyecto, nombre del Ingeniero Director y de la empresa consultora.

6. Si el proyecto pasa la etapa de desarrollo, cualquier modificación que se realice sobre él, deberá ser notificada al Ingeniero Director del Proyecto y a criterio de éste, la empresa consultora decidirá aceptar o no la modificación propuesta.

7. Si la modificación se acepta, la empresa consultora se hará responsable al mismo nivel que el proyecto inicial del que resulta el añadirla.

8. Si la modificación no es aceptada, por el contrario, la empresa consultora declinará toda responsabilidad que se derive de la aplicación o influencia de la misma.

9. Si la empresa cliente decide desarrollar industrialmente uno o varios productos en los que resulte parcial o totalmente aplicable el estudio de este proyecto, deberá comunicarlo a la empresa consultora.

10. La empresa consultora no se responsabiliza de los efectos laterales que se puedan producir en el momento en que se utilice la herramienta objeto del presente proyecto para la realización de otras aplicaciones.

11. La empresa consultora tendrá prioridad respecto a otras en la elaboración de los proyectos auxiliares que fuese necesario desarrollar para dicha aplicación industrial, siempre que no haga explícita renuncia a este hecho. En este caso, deberá autorizar expresamente los proyectos presentados por otros.

12. El Ingeniero Director del presente proyecto, será el responsable de la dirección de la aplicación industrial siempre que la empresa consultora lo estime oportuno. En caso contrario, la persona designada deberá contar con la autorización del mismo, quien delegará en él las responsabilidades que ostente.