TESIS DOCTORAL

Técnicas de minería de datos en el proceso de secuencias temporales

Aplicaciones a la clasificación industrial de sonidos

Autor:

Javier Romero Lemos

Directores:

Amalia Luque Sendra

Alejandro Carrasco Muñoz

Sevilla, 2017

AGRADECIMIENTOS

En primer lugar quisiera expresar mis más sinceros agradecimientos a la Dra. Dña. Amalia Luque Sendra y al Dr. D. Alejandro Carrasco Muñoz por haber realizado la dirección del presente trabajo de tesis. Gracias por la confianza mostrada, el impulso y su dedicación que han hecho posible la realización del mismo.

De igual modo, hago extensivos mis agradecimientos a los miembros del Departamento de Ingeniería del Diseño y del Departamento de Tecnología Electrónica de la Universidad de Sevilla que han participado de alguna forma en la consecución de este trabajo.

Manifiesto también mi agradecimiento al Dr. Rafael Ignacio Márquez Martínez de Orense (Museo Nacional de Ciencias Naturales) y al Dr. Juan Francisco Beltrán Gala (Facultad de Biología de la Universidad de Sevilla) por su colaboración y apoyo.

Deseo también expresar mi reconocimiento a las distintas instituciones que han financiado este trabajo: a la Consejería de Innovación, Ciencia y Empresa, Junta de Andalucía a través del proyecto de excelencia eSAPIENS (TIC-5705); y a la empresa Telefónica, mediante la “Cátedra de Telefónica Inteligencia en la Red”.

Gracias a mis padres, hermanas y familia por su apoyo y fe en mis capacidades. A mi suegra y cuñado por la cobertura ofrecida sin la que poco hubiera podido hacer.

Especialmente quiero agradecer y dedicar este trabajo a mi mujer y mis hijos. A mi mujer por su enorme esfuerzo y apoyo incondicional. A mis hijos, a los que he robado tantas horas de mi tiempo que les correspondían y usé para la realización de esta tesis.

¡Gracias a todos por ayudarme a cumplir un sueño!

RESUMEN

El proceso de secuencias temporales supone un campo de trabajo específico dentro de las técnicas de minería de datos o aprendizaje automático. Entre las tareas de esta disciplina se encuentra la clasificación de secuencias temporales que, por su especificidad, admite el uso de tratamientos diferenciados.

Entre los datos con estructura de secuencia temporal pueden destacarse las señales sonoras. Existen numerosas aplicaciones en las que resulta de utilidad la clasificación automatizada de sonidos. En muchas de ellas se requiere que la solución propuesta tenga unas características que podríamos calificar de industriales: robustez, inmunidad al ruido, normalización, operación en tiempo real, bajo consumo y bajo coste.

En esta tesis se analizan y comparan distintos métodos de clasificación de sonidos. El primer paso para ello supone la segmentación del sonido en fragmentos (ventanas) de muy corta duración. A continuación se caracteriza cada ventana de sonido mediante un conjunto de parámetros. Para ello se propone el uso del estándar ISO MPEG-7 cuya aplicación permite obtener un conjunto normalizado de parámetros.

Se consideran a continuación hasta nueve algoritmos de clasificación que, tomando como patrones distintos sonidos de clases conocidas, realizan una clasificación supervisada de cada una de las ventanas sin tener en cuenta el carácter secuencial de las mismas (clasificación no secuencial).

Para tener en cuenta el carácter secuencial de los sonidos se proponen y comparan distintos métodos (clasificación secuencial). Algunos de ellos se basan en dotar a cada ventana de parámetros que tienen en cuenta las ventanas anteriores y posteriores. Otros sin embargo, como los Modelos Ocultos de Markov, permiten la caracterización nativa de una secuencia de ventanas.

Para pasar de la clasificación de una ventana, o secuencia de ventanas, a la clasificación de un sonido completo se puede proceder mediante un simple conteo de las clases individuales. Sin embargo, en la presente investigación se propone una alternativa: la clasificación de series derivadas. Se define una serie (vectorial) derivada como la secuencia de probabilidades de que cada ventana pertenezca a una determinada clase.

Se propone la caracterización de las series derivadas como si se tratase de sonidos, es decir, mediante la caracterización de cada uno de sus ventanas usando parámetros MPEG-7 y su posterior clasificación supervisada usando alguno de los algoritmos clasificadores propios de la minería de datos.

El conjunto de técnicas anteriores ha sido utilizado en la clasificación de diversos archivos de sonidos de anuros proporcionados por la Fonoteca Zoológica del Museo Nacional de Ciencias Naturales. Una característica común a todos estos sonidos es que se han grabado en el hábitat natural con una significativa presencia de ruidos (viento, agua, lluvia, tráfico, voces,…), lo que ha supuesto un desafío adicional al proceso y clasificación de la señal sonora.

El resultado del análisis realizado permite afirmar que el uso de los parámetros MPEG-7 constituye una buena alternativa para caracterizar sonidos. En la aplicación analizada el mejor clasificador no secuencial ha resultado ser el árbol de decisión. Por otra parte la introducción de un método de ventana deslizante aparece como la mejor opción de clasificación secuencial, aunque con una mejora muy discreta sobre la técnica no secuencial. Adicionalmente, se ha podido evidenciar que la clasificación de las series derivadas supone una mejora muy notable en las prestaciones del clasificador. Por último, se ha comprobado que la solución propuesta presenta las características adecuadas para poder proclamar su carácter industrial.

ÍNDICE DE CONTENIDOS

ÍNDICE DE FIGURAS

ÍNDICE DE TABLAS

CAPÍTULO 1. INTRODUCCIÓN

1.1. Objeto de la tesis

Numerosos proyectos industriales, así como amplios campos de la ciencia y la tecnología, se enfrentan al problema del procesamiento de grandes volúmenes de datos que tienen estructura de secuencias temporales. Su tratamiento se realiza, en muchos casos, utilizando técnicas específicas referidas al problema planteado que son válidas dentro de su campo de aplicación.

Por otra parte, el incremento constante en la capacidad de almacenamiento y proceso de grandes volúmenes de información está posibilitando la difusión de técnicas que, bajo el nombre genérico de minería de datos, permiten abordar numerosos problemas mediante enfoques estadísticos y de inteligencia artificial.

El objetivo de la tesis es el estudio de la aplicabilidad de esas técnicas de minería de datos al caso en que la información a procesar tenga una estructura de secuencias temporales, lo que conlleva importantes condicionantes.

Los métodos propuestos en la tesis son contrastados y validados mediante su aplicación a un problema real: la clasificación de sonidos con carácter industrial y, en concreto, a la identificación automática de especies de anuros. El carácter industrial de la clasificación se reflejará, entre otros parámetros, en: la tolerancia ante ruidos y perturbaciones; la normalización de la representación de los sonidos; la capacidad de proceso en tiempo real; y la integrabilidad en sistemas de bajo consumo y bajo coste.

1.2. Contexto de la aplicación

Desde que en 1950 Roger Revelle alertara sobre las consecuencias de los gases de efecto invernadero, la comunidad científica internacional ha ido buscando indicadores de la influencia humana en el clima. En 1972, Dennis Meadows presentó en su informe sobre los límites del crecimiento (Meadows, Meadows, Randers, & Behrens, 1972) el primer modelo informático predictivo del calentamiento causado por los combustibles

fósiles. Desde entonces se han propuesto multitud de modelos que tratan de explicar la evolución a largo plazo de diferentes indicadores climáticos.

Una de las consecuencias del cambio climático es la incidencia que tiene sobre el desarrollo de funciones fisiológicas básicas de diversas especies (Deutsch et al., 2008; Duarte et al., 2012; Huey et al., 2009; Kearney, Shine, & Porter, 2009; Pörtner & Knust, 2007). Así, por ejemplo, el sonido producido durante el canto de llamada juega un papel central en la selección sexual y en la reproducción de numerosas especies ectotermas (las que regulan su temperatura a partir de la temperatura ambiental), entre las que se incluyen los anuros (ranas y sapos), peces e insectos (Bradbury & Vehrencamp, 1998; Fay & Popper, 2012; Gerhardt & Huber, 2002). Los distintos patrones acústicos se utilizan para la atracción de potenciales parejas, como medio de defensa, para alejar a los oponentes, y para responder a los riesgos de depredación. Estos sonidos son, por tanto, fundamentales para la adaptación de los individuos al medio.

Sin embargo, la producción de sonidos en los animales ectotermos está fuertemente influida por la temperatura ambiente (Bellis, 1957; Gayou, 1984; Gerhardt & Mudry, 1980; Márquez & Bosch, 1995; Pires & Hoy, 1992; Schneider, 1974; Walker, 1957, 1962) pudiendo afectar a diversas características de su sistema de comunicación acústico. De hecho la temperatura ambiente, una vez superado cierto umbral, puede restringir los procesos fisiológicos asociados a la producción de sonidos hasta el punto de llegar a inhibir los comportamientos de llamada. Como consecuencia, la temperatura puede llegar a afectar de forma notable a los patrones de los cantos de llamada modificando el comienzo, la duración y la intensidad de los episodios de llamada y, consecuentemente, la actividad reproductiva.

Por ello el análisis y clasificación de los sonidos producidos por determinadas especies animales se ha mostrado un potente indicador de los cambios de temperatura y, por tanto, de la existencia del cambio climático. Especialmente interesante son los resultados obtenidos a partir de la consideración de los sonidos producidos por anuros (Llusia, Márquez, Beltrán, Benítez, & do Amaral, 2013).

Este análisis requiere en primer lugar la grabación de los distintos sonidos en su entorno natural, pudiendo usarse para ello dispositivos como los descritos en (M. E. Cambron & Bowker, 2006). El procesado de los sonidos grabados puede realizarse en tiempo real de forma local (Aide et al., 2013) o en un centro remoto requiriendo, en este caso, un adecuado sistema de transmisión, normalmente redes de sensores inalámbricas (*wireless sensor networks* WSN), lo que generalmente requiere la aplicación de técnicas adecuadas de compresión de la información (Diaz, Nakamura, Yehia, Salles, & Loureiro, 2012).

Cuando el objetivo es la determinación de indicadores de cambio climático, el requisito de proceso en tiempo real en campo no suele ser necesario, por lo que los sonidos pueden ser analizados a partir de los disponibles en bases de datos como los disponibles en la Fonoteca del Museo Nacional de Ciencias Naturales (Fonozoo, 2015), la *Macaulay Library of Natural Sounds* (Macaulaylibrary, 2015), la *British Library Sound Archive* (British Library, 2015), y la *Animal Sound Archive in Berlin* (Animalsoundarchive, 2015). En (Bardeli, 2009; Weninger & Schuller, 2011) se presentan distintas técnicas de procesado y clasificación de sonidos de animales a partir de esta última fonoteca.

El procesado y clasificación de sonidos de animales es un tema recurrente en la literatura por ejemplo en (Potamitis, 2015). En (Huang, Yang, Yang, & Chen, 2009) se presenta un sistema de identificación específica de anuros con el objetivo de proporcionar consulta abierta *on-line*.

1.3. Antecedentes tecnológicos

El proceso de identificación de sonidos en general, y de sonidos de anuros en particular, consta de dos fases. En la primera de ellas se realiza una extracción de un conjunto más o menos amplio de parámetros. La segunda etapa, a continuación, realiza la clasificación del sonido en base a los parámetros anteriores.

La mayoría de los algoritmos utilizados basan la clasificación en parámetros espectrales o temporales tales como, por ejemplo, el centroide espectral, el ancho de banda o la tasa de cruces por cero (Benesty, 2008; Fulop, 2011; L. Rabiner & Juang, 1993). Dependiendo de las aplicaciones, de los tipos de sonido y, en muchos casos, de la elección de los autores, estos algoritmos carecen de homogeneidad, tanto en el tipo de parámetro utilizado como en su propia definición.

Un caso particular de parámetros espectrales lo constituyen los *Mel Frequency Cepstral Coefficients*: MFCCs (Zheng, Zhang, & Song, 2001). Estos parámetros, muy usados en la literatura en procesos de clasificación y procesado de sonidos, se encuentran unívocamente definidos e incluso normalizados (ETSI, 2002) por lo que son una buena solución para resolver la heterogeneidad descrita en el párrafo anterior.

Si bien los MFCCs presentan un conjunto de parámetros normalizados susceptibles de ser aplicados a la clasificación de sonidos, ofrecen una visión unidimensional del segmento de audio al que se refieren. En efecto, todos los parámetros derivan del mismo enfoque: el cepstrum de la señal sonora (función calculada a partir de su espectro de potencia). El uso de los MFCCs, aun teniendo la ventaja de la normalización, limita las posibilidades de exploración de otros parámetros que pudieran llegar a ser más expresivos.

Una alternativa lo constituye el uso de la norma MPEG-7 (ISO, 2001). El objetivo de esta norma no es originalmente la clasificación de sonidos sino otro mucho más

amplio: la descripción normalizada de contenido multimedia (texto, imágenes, audio, vídeo,…). Pero para describir sonidos la norma, en su parte 4, propone un conjunto de parámetros y algoritmos mucho más ricos desde el punto de vista semántico que el proporcionado por los MFCCs.

Los parámetros MPEG-7 poseen, por tanto, una doble característica: normalización y riqueza semántica. Eso los hace especialmente atractivos para explorar técnicas de clasificación de sonidos (Casey, 2001). Por otra parte la comparación entre técnicas de clasificación basadas en MFCCs y técnicas basadas en MPEG-7 muestran diversidad de resultados en función de las condiciones de uso (H. G. Kim & Sikora, 2004; Ntalampiras, Potamitis, & Fakotakis, 2008).

No obstante, sean cuales fueren los parámetros utilizados para caracterizar un sonido, éstos no son más que la base sobre la que actúan los algoritmos de clasificación. En este aspecto hay también una gran variedad de soluciones en la literatura, predominando quizás las soluciones basadas en modelos ocultos de Markov (L. R. Rabiner, 1989). Precisamente esta es la técnica elegida en la norma MPEG-7.

Sin embargo, la clasificación de patrones, sean estos sonidos o no, es un campo de trabajo con una rica tradición y en el que se han hecho numerosas propuestas. Bajo los nombres de aprendizaje automático, *machine learning*, minería de datos o *business intelligence* se pueden encontrar potentes algoritmos especializados en la clasificación de patrones en general (Bishop, 2006; Flach, 2012); así como, en particular, de clasificación de datos secuenciales (Dietterich, 2002; Esling & Agon, 2012), y de clasificación de sonidos (Gopi, 2014; Theodoridis & Chellappa, 2013).

1.4. Estructura de la tesis

El presente documento de tesis está dividido en siete capítulos de acuerdo con la siguiente estructura:

- **Capítulo 1: Introducción;** contextualiza el trabajo y realiza una breve descripción del mismo, destacando sus objetivos y estructura.
- **Capítulo 2: Técnicas clásicas de procesamiento de sonidos;** realiza una descripción del estado del arte en cuanto a las técnicas de procesamiento de sonidos: en el domino del tiempo; en el domino de la frecuencia; procesamiento homomórfico; y codificación predictiva lineal. A través de cada una de estas técnicas se pueden extraer distintas características de los sonidos, lo que permitirá su representación para una ulterior clasificación.
- **Capítulo 3: Caracterización de sonidos mediante el estándar MPEG-7;** realiza un estudio de la estructura de la norma, profundizando en su parte 4 encargada de la descripción del contenido de audio y, más concretamente, en los descriptores del sonido a bajo nivel (timbre, ritmo, distribución espacial,

temporal y espectral o fuente de sonido). Elige y define los parámetros que caracterizarán cada fragmento de un sonido en los capítulos restantes.

- **Capítulo 4: Técnicas de clasificación;** presenta y define distintas técnicas de clasificación dentro de las utilizadas en el ámbito de la minería de datos. Aplica estas técnicas a la clasificación de los fragmentos de sonidos representados mediante los parámetros MPEG-7 definidos en el capítulo anterior. En esta primera aproximación no se tiene en cuenta el carácter secuencial (el orden) de los fragmentos.

- **Capítulo 5: Clasificación de secuencias temporales;** analiza distintas técnicas que permiten tener en cuenta el carácter secuencial de los sonidos. Se consideran especialmente las técnicas basadas en los Modelos Ocultos de Markov, al ser éste el clasificador recomendado por la norma MPEG-7. Se comparan las prestaciones de los clasificadores de secuencias con los resultados obtenidos en el capítulo anterior.

- **Capítulo 6: Clasificación de series derivadas;** introduce el concepto de serie vectorial derivada como aquella secuencia en la que cada fragmento de sonido está representado por un vector cuyos elementos son las probabilidades (o puntuaciones) de pertenecer a cada clase posible de sonido. Se exploran distintas técnicas de clasificación de series derivadas y se comparan sus prestaciones con los resultados obtenidos en los capítulos anteriores. Se realizan finalmente diversas consideraciones de implementación que tratan de proporcionar un carácter industrial al proceso de clasificación propuesto.

- **Capítulo 7: Resumen y conclusiones;** presenta un resumen del trabajo realizado y las conclusiones derivadas del mismo, haciendo especial hincapié en los aspectos más novedosos de la tesis. Se presentan igualmente también las posibles líneas de continuación del presente trabajo de investigación.

CAPÍTULO 2. TÉCNICAS CLÁSICAS DE PROCESAMIENTO DE SONIDOS

2.1. Introducción

Una de las posibles definiciones de sonido es: "Sensación producida en el órgano del oído por el movimiento vibratorio de los cuerpos, transmitido por un medio elástico, como el aire" (Real Academia Española, 2014).

Los sonidos se pueden clasificar, de forma genérica, en sonoros y no sonoros o sordos. Los primero se producen mediante un tren de impulsos casi periódicos, normalmente producido por resonancia. El período o frecuencia fundamental de este tren de impulsos se conoce con el nombre de tono o "*pitch*". Debido a la resonancia, se producen grandes picos en el espectro resultante, a los cuales se les llaman formantes. En cambio, los sonidos sordos presentan una estructura ruidosa.

En este capítulo se repasarán las técnicas clásicas usadas en el procesamiento y caracterización de sonidos, basadas en propiedades del dominio del tiempo, dominio de las frecuencias, derivadas del procesamiento homomórfico y de la codificación predictiva lineal. A través de cada una de estas técnicas, se pueden extraer distintas características de los sonidos que pueden servir, entre otros propósitos, para la clasificación.

La selección de la mejor representación paramétrica de la señal sonora es una tarea de suma importancia en el diseño de cualquier sistema de reconocimiento vocal. El objetivo principal de estas representaciones paramétricas es comprimir los datos de la señal sonora, eliminando la información no pertinente para el análisis y extraer las características que contribuyen de manera significativa con la detección de las diferentes fonéticas, las cuales no son apreciables mediante un simple análisis en tiempo o frecuencia.

Debido a la naturaleza cambiante y partiendo de la suposición que las propiedades de la señal vocal varían relativamente despacio en el tiempo, resulta más conveniente, realizar el procesamiento y análisis a porciones de la señal, pudiendo observar la evolución de los distintos parámetros calculados. A estas porciones de señal se les denominan trama o "*frame"* (L. R. Rabiner & Schafer, 1978).

Las señales vocales varían lentamente en periodos cortos de tiempo, comportándose como una señal estacionaria en *frames* de entre cinco y cien milisegundos. Sin embargo, si el *frame* contiene periodos más largos de tiempo, del orden de un quinto del segundo o más, las características de la señal cambian reflejando diferentes sonidos (L. L. Rabiner & Juang, 1993).

Cada uno de estos *frames* se procesa para obtener un valor o valores. A este proceso se le denomina **extracción de características**, y a los valores **características**. Este proceso se realiza en intervalos fijos recorriendo todo el sonido, denominado longitud de desplazamiento, solapamiento o "*hop size*".

Según la longitud del *frame* y el *hop size* se pueden dar los siguientes casos:

- *frame* < *hop size*. En este caso no hay solapamiento entre *frames* sucesivos, perdiéndose parte de la señal.
- *frame* = *hop size*. En este caso no hay pérdida de señal pero no existe correlación entre *frames* consecutivos.
- *frame* > *hop size*. Es el caso más habitual, donde los *frames* adyacentes se solapan teniendo una cierta correlación. Cuanto menor sea el *hop size*, más suaves serán las fluctuaciones sufridas por los parámetros obtenidos.

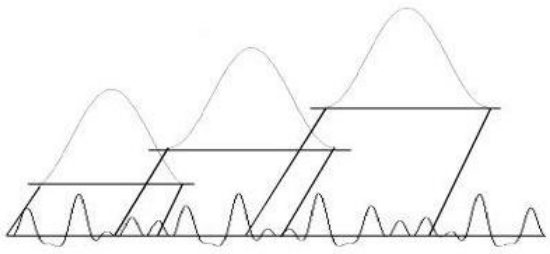


Figura 2-1 Mecanismo de establecimiento de ventanas (Bernal Bermúdez, Bobadilla Sancho, & Gómez Vilda, 2000)

Como se puede observar en la Figura 2-1, la colación de las ventanas puede realizarse de tal forma que existan solapamientos. Aunque esto repercutirá negativamente en los tiempo de respuestas del procesamiento, proporcionará una mejor calidad en los resultados obtenidos.

El procesamiento por *frame* se puede conseguir aplicando una ventana que se desplaza a lo largo de la señal. Para señales definidas por un conjunto de puntos en el dominio del tiempo discreto $x(n)$, la función ventana $w(n)$ será una secuencia

discreta real de tamaño finito. Esta ventana selecciona la muestra de una pequeña sección de señal original mediante un proceso de multiplicación punto a punto,

$$x_w(n) = x(n) \cdot w(n) \quad n = 0,1, \dots, \mathrm{N} - 1 \,. \tag{2-1}$$

Existen varios tipos de ventanas, siendo las más comunes:

- Rectangular

$$w(n) = \begin{cases} 1 & para\ n = 0, 1, \dots, N-1 \\ 0 & para\ el\ resto\ de\ casos \end{cases} \tag{2-2}$$

- Hamming

$$w(n) = \begin{cases} 0.54 - 0.46 \cos\left(\frac{2\pi n}{N-1}\right) & para\ n = 0, 1, \dots, N-1 \\ 0 & para\ el\ resto\ de\ casos \end{cases} \tag{2-3}$$

- Hanning

$$w(n) = \begin{cases} 0.5 - 0.5 \cos\left(\frac{2\pi n}{N-1}\right) & para\ n = 0, 1, \dots, N-1 \\ 0 & para\ el\ resto\ de\ casos \end{cases} \tag{2-4}$$

La correcta elección del tipo de ventana resulta de vital importancia para analizar el efecto de la misma sobre la resolución espectral de la señal, la cual depende del lóbulo principal de la ventana y la atenuación de los lóbulos secundarios respecto al principal. Se profundizará más sobre este tema en el apartado “Procesamiento en el dominio de la frecuencia”.

2.2. Procesamiento en el dominio del tiempo

Las técnicas de procesamiento en el dominio del tiempo involucran directamente la forma de onda de la señal. Aunque el mayor esfuerzo del tratamiento del sonido se centra en análisis espectrales, existen métodos para el procesamiento en el dominio del tiempo que pueden resultar muy útiles.

Representando una señal como conjunto de puntos en el dominio del tiempo discreto $x(n)$, uno de los parámetros más sencillos de calcular en el dominio del tiempo es la energía, que se puede definir como

$$E \equiv \sum_{m=-\infty}^{\infty} x^2(m) \ . \tag{2-5}$$

Este parámetro da poca información sobre la evolución de la señal en el tiempo. Para disponer de la evolución de la señal, se suele utilizar la energía de *frame* o "*short-time energy*" definida como

$$E = E_n \equiv \sum_{m=-\infty}^{\infty} [x(m)w(n-m)]^2 \ . \tag{2-6}$$

En adelante, cuando se hable de energía se estará refiriendo a la energía de *frame*.

Si se usa una ventana rectangular, la expresión anterior se simplifica como

$$E = \sum_{m=1}^{N} x^2(m) \ . \tag{2-7}$$

La ventana se desliza seleccionando el *frame* para el cálculo, como la convolución discreta entre $x^2(m)$ y la ventana, quedando simplemente la energía de la muestra n-ésima como la suma de los cuadrados de las N muestras siguientes. Siendo N el número de muestras consideradas para calcular la energía, establecido por la longitud de la ventana.

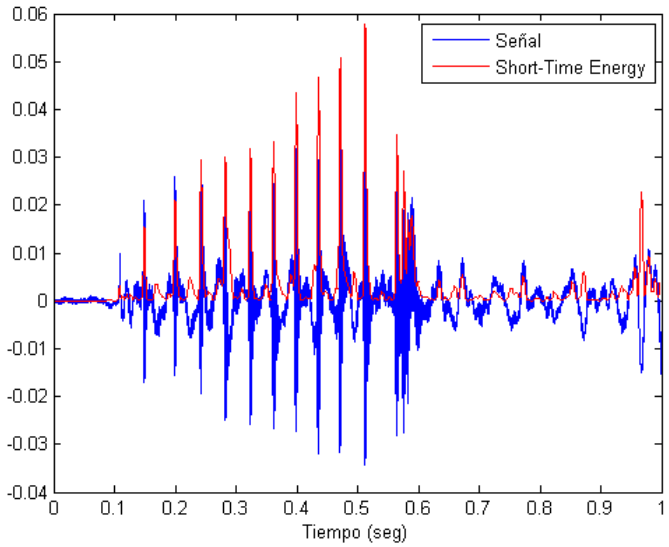


Figura 2-2 Función energía superpuesta sobre la señal vocal en el dominio del tiempo

En la Figura 2-2 se puede observar una señal vocal representada en el dominio del tiempo correspondiente al canto de un sapo corredor, proporcionado por la Fonoteca Zoológica del Museo Nacional de Ciencias Naturales. De forma superpuesta también se ha representado la función energía.

Es importante utilizar un tamaño apropiado de ventana pues con una ventana muy pequeña, del orden de un periodo o menos, la energía fluctuará rápidamente. En cambio, si la longitud de la ventana es grande, la energía cambiará lentamente y no reflejará las propiedades cambiantes de la señal. Es habitual elegir una longitud de la ventana comprendida entre los diez y los veinte milisegundos. El efecto que produce el tamaño de la ventana se puede observar en la Figura 2-3, donde se ha representado la energía para diferentes tamaños de ventana del tipo Hamming.

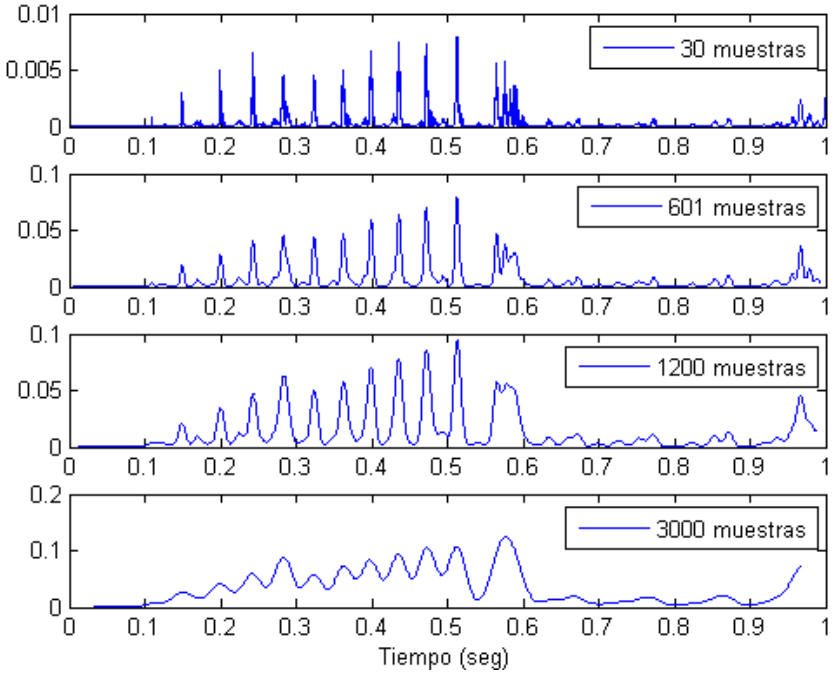


Figura 2-3 Función energía con distintos tamaños de ventanas de Hamming

La energía, entre otras cualidades, permite distinguir con cierta fiabilidad el sonido del silencio y los sonidos sordos de los sonoros.

El rango de valores de la energía, dado que se calcula utilizando una suma de cuadrados, existen grandes diferencias entre la energía de señales de gran y baja amplitud. Por este motivo, la energía es muy sensible a niveles altos de señal, enfatizando las diferencias de amplitud. Una forma sencilla de aliviar este problema es definir la magnitud media como

$$M_n \equiv \sum_{m=-\infty}^{\infty} |x(m)| w(n-m) \ . \qquad \textbf{(2-8)}$$

Como en el caso de la energía, si se utiliza una ventana rectangular la expresión se simplifica a

$$M_n = \sum_{m=1}^{N} |x(m)| \ . \qquad \textbf{(2-9)}$$

En la Figura 2-4 se compara las funciones energía y magnitud media para la misma señal vocal.

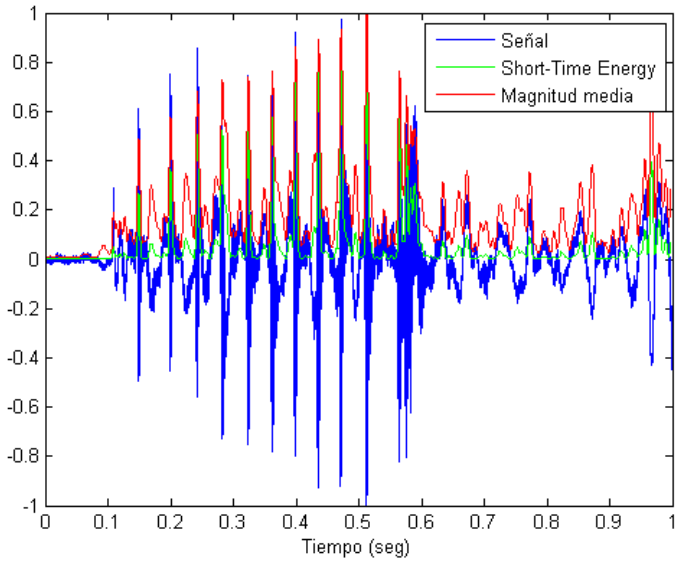


Figura 2-4 Comparación entre la energía y la magnitud media

Otro parámetro interesante, que se obtiene del tratamiento de la señal en el dominio del tiempo, es el número de cruces por cero. Este ratio tiene una correspondencia directa con las frecuencias que contiene la señal. Las señales vocales son de banda ancha y el ratio de cruces por cero es menos preciso. Sin embargo, usando *frames* se puede observar información de las propiedades espectrales.

Un cruce por cero se produce cuando dos muestras consecutivas de una señal tienen diferente signo. Una característica de la señal basada en el cruce por cero es la densidad de cruces por cero, que consiste en el número de cruces por cero que se producen en un cierto segmento de señal, expresado como

$$Z_n \equiv \sum_{m=-\infty}^{\infty} |sgn[x(m)] - sgn[x(m-1)]| \cdot w(n-m) \ . \qquad \textbf{(2-10)}$$

Si se usa una ventana rectangular, la expresión se simplifica a

$$Z_n = \sum_{m=1}^{N-1} |sgn[x(m)] - sgn[x(m-1)]| \ . \qquad \textbf{(2-11)}$$

Como se puede observar en la Figura 2-5, si la medida de Z_n es elevada existen componentes de alta frecuencia, y viceversa. Este parámetro es bastante fiable para la discriminación de los sonidos sordos y sonoros. Una alta tasa de cruces por cero se puede corresponder generalmente a *frames* sordos, mientras que un ratio pequeño corresponde con *frames* sonoros.

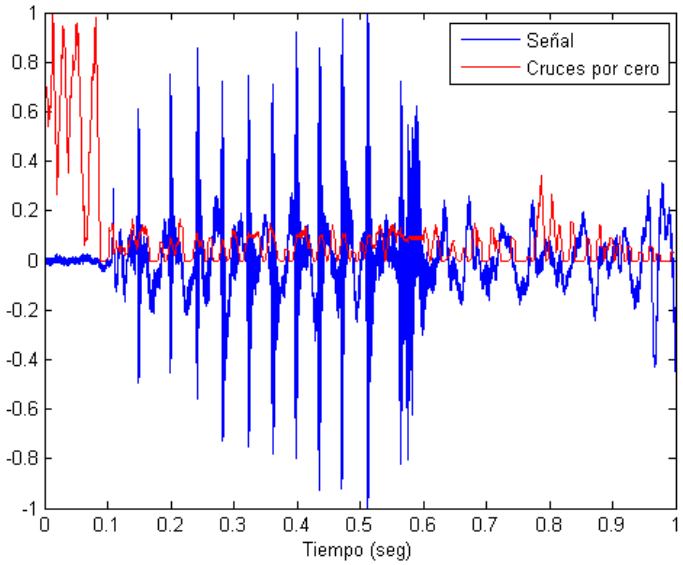


Figura 2-5 Ratio de cruces por cero superpuesto sobre la señal vocal en el dominio del tiempo

Hay que tomar con cautela este parámetro pues es fuertemente influenciable por múltiples factores como pueden ser el nivel de continua (DC) de la señal y ruidos del sistema de digitalización.

La energía o magnitud media y el ratio de cruces por cero se suelen usar de forma combinada para localizar el inicio y final de las señales de voz y los tramos sordos y sonoros.

Otro parámetro importante de las señales en el dominio del tiempo es la frecuencia fundamental, tono o *pitch*, que puede ser usado en múltiples problemas de clasificación de sonidos.

Las señales sonoras se caracterizan por tener un valor de *pitch* muy claro. Sin embargo, las señales sordas carecen de periodicidad. Es en las transiciones donde aparecen los problemas. Existen numerosos algoritmos para calcular el *pitch*, sin embargo, no existe algoritmo conocido que lo determine con un 100% de fiabilidad, ya que no siempre es fácil determinar si existe periodicidad en una señal o no. El *pitch* varía mucho entre diferentes individuos, e incluso para un mismo individuo puede variar en función de diversos factores.

Uno de los métodos usados para la identificación del *pitch* es la función de autocorrelación, definida como

$$\varphi(k) \equiv \sum_{m=-\infty}^{\infty} x(m) \cdot x(m+k) \ . \qquad \textbf{(2-12)}$$

La función de autocorrelación de una señal periódica es también periódica con el mismo período.

Sean $x(n)$ e $y(n)$ dos señales de longitud finita. Una medida de similitud es el error cuadrático medio entre ambas. Si se considera un posible desplazamiento entre ambas señales, se obtiene que el error cuadrático medio es

$$P(k) = \sum_{m=-\infty}^{\infty} \big(x(m) - y(m+k)\big)^2 \ . \qquad \textbf{(2-13)}$$

Desarrollando la expresión anterior se llega a

$$P(k) = \sum_{m=-\infty}^{\infty} x^2(m) + \sum_{m=-\infty}^{\infty} y^2(m+k) - 2\sum_{m=-\infty}^{\infty} x(m)y(m+k) \ . \qquad \textbf{(2-14)}$$

Los dos primeros términos de la expresión anterior se corresponden con la energía ambas señales, independen del desplazamiento entre ambas señales. Por tanto, la función P se hará mínima cuando el último miembro de la expresión se haga máximo. Si ambas señales son la misma señal el último sumando coincide con la definición de la función de autocorrelación.

El valor estimado de *pitch* se corresponde con el mayor máximo local resultante de la función autocorrelación, excluyendo el máximo global. En la Figura 2-6 se puede observar la función de autocorrelación correspondiente al sonido, anteriormente usado, del sapo corredor.

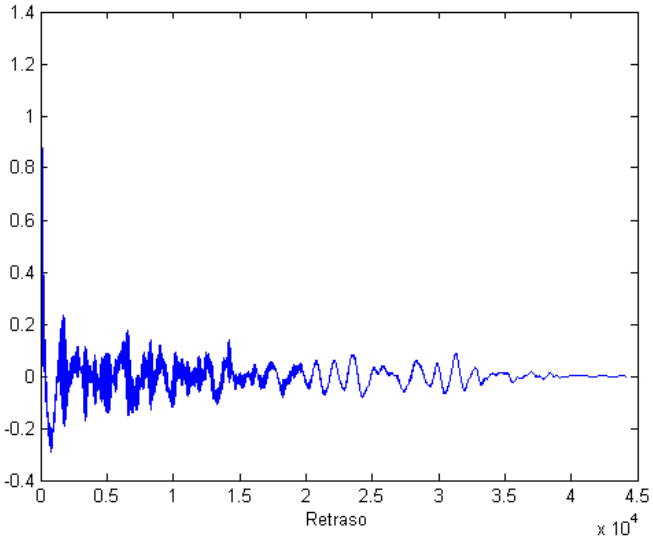


Figura 2-6 Función de autocorrelación

Un problema de la función de autocorrelación es la aparición de picos extraños. Este efecto se puede aliviar, si previo al cálculo de la autocorrelación, se pasa la señal por un filtro *center-clipping*. Este filtro, consiste en realizar una transformación no-lineal de la señal, según el siguiente la Figura 2-7(a). Esta función produce la eliminación de toda la señal que no supera un cierto umbral, que normalmente se fija en un 30% del máximo de la señal en el *frame* en cuestión. El coste computacional de la aplicación de

este filtro es alto por lo que se propone una modificación del mismo, el filtro *3-level center-clipping* representando en la Figura 2-7(b).

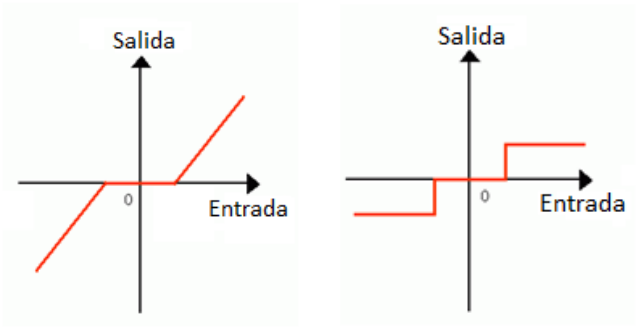


Figura 2-7 (a) Filtro *center-clipping* (b) Filtro *3-level center-clipping*

2.3. Procesamiento en el dominio de la frecuencia

La representación del dominio de tiempo muestra las amplitudes de una señal en el instante de tiempo que ocurre el muestreo. En muchos casos se desea saber el contenido de la frecuencia de una señal antes que las amplitudes de señales individuales.

El algoritmo que se utiliza para transformar datos en el dominio del tiempo al dominio de la frecuencia se conoce como Transformada de Fourier. La transformada de Fourier es una herramienta muy útil cuando se trabaja con modelos matemáticos. En el caso de estudio, las señales son discretas y mediante el uso de ventanas se limita el número de puntos. Para este conjunto de datos se utiliza la transformada discreta de Fourier (DFT).

La DFT es siempre compleja por lo que estará formada por amplitud y fase. La fase tiene poco importancia en el reconocimiento vocal, por tanto, a partir ahora cuando se refiera a DFT se referirá a la amplitud de la DFT. La transformada rápida de Fourier es un algoritmo que permite calcular de forma eficiente la DFT. Para poder aplicar este algoritmo, la secuencia de entrada debe ser múltiplo de 2. Una técnica empleada, para hacer que el tamaño de la secuencia de entrada sea múltiplo de 2, es agregar ceros al final de la secuencia de modo que el número total de muestras sea igual al siguiente múltiplo de 2. Como se verá a continuación, la inclusión de ceros aumenta el tamaño de la muestra lo que ayuda a incrementar la resolución de la frecuencia.

Cuando se trabaja con señales vocales, al no ser no estacionaria, se toman ventanas relativamente pequeñas para que dentro de cada ventana se pueda considerar cuasiestacionaria (Bernal & Gómez, Pedro, Bobadilla, 1999). El efecto que provoca el enventadano es convolucionar el espectro de la señal muestreada con el espectro de la ventana, produciendo una distorsión de la transformada de la señal original. Por esto es conveniente elegir un tipo de ventana que produzca menor distorsión, minimizando las discontinuidades que la señal tiene al comienzo y final de cada *frame*. Esta ventana

debe tener un valor próximo a cero en sus extremos, y normalmente es simétrica respecto del centro de la misma. La multiplicación de la señal por la ventana tiene dos efectos:

- Atenúa de forma gradual la señal a ambos lados del *frame* seleccionado.
- Produce una convolución de la transformada de Fourier de la función ventana y el espectro de la señal.

Debido al segundo efecto, la ventana debe satisfacer dos características para reducir la distorsión espectral introducida por el enventanado:

- Lóbulo principal estrecho y agudo, con buena resolución en alta frecuencia.
- Gran atenuación de los lóbulos secundarios.

Estas dos características, generalmente, son contrapuestas, y por tanto es preciso buscar un compromiso entre ambas.

El uso de una ventana grande permite analizar muy bien los armónicos, pero la envolvente espectral se "camuflada" en cierto modo. Por el contrario, el uso de ventana pequeña, permite ver la envolvente espectral de forma más limpia (Galindo Riaño, 1996). Este efecto se puede observa en la Figura 2-8.

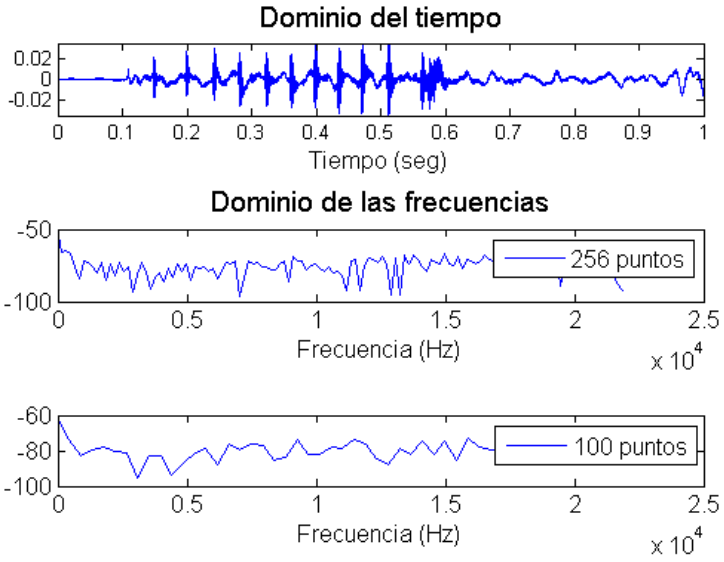


Figura 2-8 Espectros de amplitud de una señal para distintos tamaños de ventana

Una buena opción para enventanar señales sonoras puede ser la ventana rectangular con una duración igual a un período del *pitch*. Esto producirá un espectro de salida muy cercano a la respuesta al impulso. La dificultad estriba en la localización exacta y fiable del período de *pitch*.

Por otra parte, si se toma una ventana pequeña la resolución frecuencial resulta pobre, pues la DFT estudia tantas frecuencias como puntos se indiquen. La frecuencia máxima estudiable para N puntos es

$$f_{\text{maxestudiable}} = \frac{\frac{N}{2}-1}{N \cdot T} \text{ para } N \text{ par con } T = \frac{1}{2 \cdot f_{max}} , \quad \textbf{(2-15)}$$

donde f_{max} es la frecuencia máxima de la señal.

Según la ecuación (2-15), si se aumenta el tamaño de la ventana (N) el número de frecuencias que se estudian es mayor, a costa de disminuir la distancia entre ellas y obteniendo una mejor precisión. La razón matemática indica que cuanto más ancha sea la ventana, más estrecha resulta la función transformada de la ventana, produciendo una menor distorsión en el espectro de la función original.

Para reducir el rango dinámico de las señales espectrales, se suele alisar el espectro para compensar los valores de altas y bajas frecuencias mediante un filtro preénfasis. Idealmente, el filtro preénfasis sólo se debe aplicar a señales sonoras. Sin embargo, por la pequeña distorsión que se introduce en las señales aperiódicas, y por simplificar el sistema de análisis, la práctica totalidad de los sistemas actuales aplican el filtro preénfasis a todo tipo de señales.

En las envolventes de los espectros se pueden observar unos picos en torno a los múltiplos de la frecuencia fundamental. Estos picos armónicos se denominan formantes. Realmente son zonas de resonancia en las que se pone de relieve un conjunto determinado de armónicos. Al valor de *pitch* se le denomina formante 0, F_0. La utilidad de los formantes es determinante en la discriminación de numerosos sonidos. Así por ejemplo, simplemente utilizando la posición de los formantes F_1 y F_2, se puede realizar un mapa que permita diferenciar los fonemas vocálicos de la voz humana.

Mediante la FFT se calcula el espectro $S_i(f)$ de cada *frame*. Si se generaliza para todos los *frames* que componen la señal se obtiene el espectro en cada instante, como una función $S(t,f)$ compleja de dos variables, tiempo y frecuencia. A partir de esta función, se puede obtener el espectro de potencia del sonido en cada *frame*,

$$P(t,f) = S(t,f) \cdot S^*(t,f) = |S(t,f)|^2 \ . \quad \textbf{(2-16)}$$

Se denomina espectrograma a la representación gráfica de esta función, $P(t,f)$. El espectrograma se puede realizar de dos formas:

- Gráfico de 3 dimensiones $[t,f,P]$ (Figura 2-9).
- Gráfico de 2 dimensiones $[t,f]$, señalando con distinta intensidad de color el valor de la potencia, P (Figura 2-10).

La opción más común es la segunda que es más fácil interpretar y los formantes aparecen como franjas horizontales (Faúndez Zanuy, 2000).

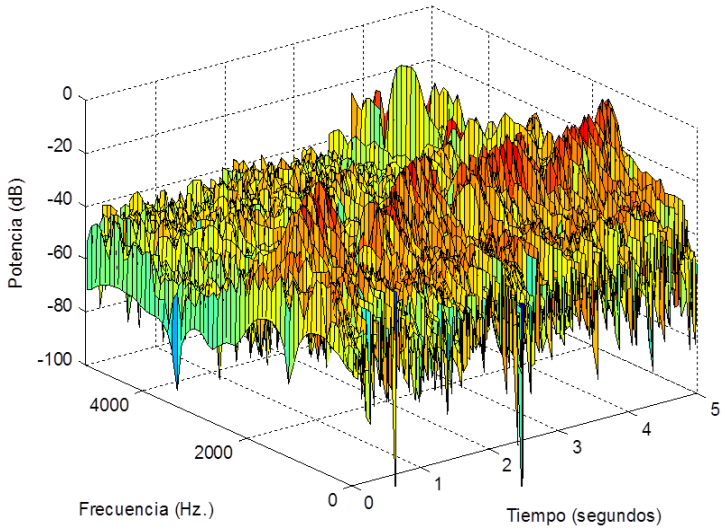


Figura 2-9 Representación espectro-temporal de la potencia (3D)

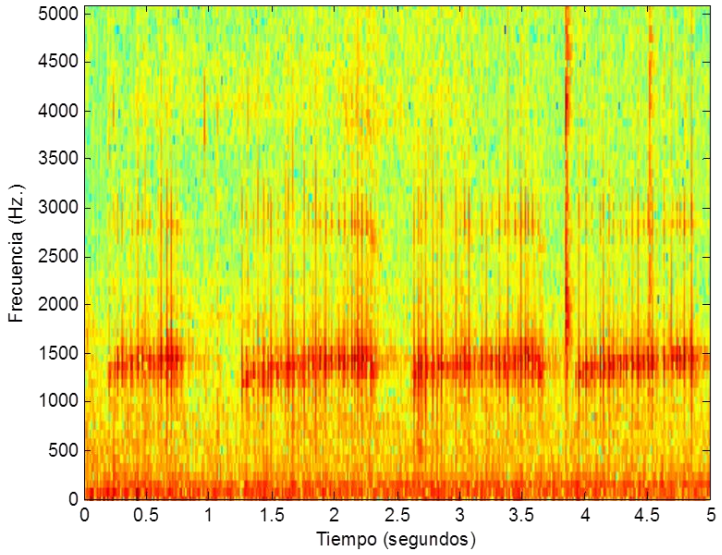


Figura 2-10 Espectrograma

En estas representaciones, algunos valores de potencia pueden quedar "camuflados" para lo cual resulta más conveniente usar una escala logarítmica para las magnitudes de amplitud, representando dB.

El espectrograma muestra los valores de las amplitudes en un rango dinámico que hay que calibrar. La forma habitual es usar un filtro pre-énfasis, que independiza el tono del emisor (Fulop, 2011).

Sobre el espectrograma se puede definir un amplio conjunto de parámetros, entre los cuales se pueden señalar el centroide de potencia y la dispersión espectral.

El Centroide de potencia es una forma de expresar resumidamente la forma del espectro. Indica si el espectro está dominado por altas o bajas frecuencias,

$$C_p \equiv \frac{\sum_i f_i \cdot P_i}{\sum_i P_i} \ . \tag{2-17}$$

La dispersión espectral de potencia expresa de forma simple la forma del espectro e indica si la potencia está concentrada alrededor del Centroide o, por el contrario, se encuentra dispersa a lo largo de todo el espectro,

$$D_e \equiv \sqrt{\frac{\sum_i \left(f_i - C_p\right)^2 \cdot P_i}{\sum_i P_i}} \ . \tag{2-18}$$

2.4. Procesamiento homomórfico

En la producción de la voz intervienen diversos órganos que forman el aparato fonador. Es importante analizar cómo se produce la voz y obtener un modelo para que en base a éste se puedan diseñar reconocedores de voz que aprovechen las características de este modelo. Una simplificación de la realidad es considerar consiste en separar la influencia de los distintos factores, en concreto separando la influencia de la excitación de la del tracto vocal. De esta forma, la producción de la voz puede ser modelada en tiempos cortos como la convolución de una función de excitación con la respuesta al impulso del tracto vocal.

La descomposición homomórfica, es una técnica diseñada para separar componentes de una señal convolucionada por medio de una transformación.

Sea

$$s(t) = x(t) * y(t) \tag{2-19}$$

una señal sonora resultante de la convolución de otras dos señales. Si se aplica la transformada de Fourier se tiene

$$S(f) = X(f) \cdot Y(f) \ , \tag{2-20}$$

en el espacio de la frecuencia.

Tomando el logaritmo en ambos lados de la expresión anterior

$$\ln S(f) = \ln X(f) + \ln Y(f) \ . \tag{2-21}$$

De esta forma, una convolución en el dominio del tiempo, se ha transformado en una suma de componentes logarítmicas en el dominio de la frecuencia. Finalmente, para separar las componentes x e y, se debe aplicar una transformada inversa de Fourier para el logaritmo del espectro

$$F^{-1}\{\ln S(f)\} = F^{-1}\{\ln X(f)\} + F^{-1}\{\ln Y(f)\} \ . \tag{2-22}$$

Esta última transformación se encuentra en el dominio del tiempo, pero no es el mismo tiempo que el de la señal original. Es una medida de la velocidad del cambio de la magnitud espectral. Este dominio es llamado cepstral y existe una correspondencia entre las medidas del espacio espectral y el cepstral:

Espectral	Cepstral
Espectro	Cepstro
Frecuencia	Quefrencia
Armónico	Ramónico
Magnitud	Gamnitude
Fase	Saphe
Filtro	Liftro

Tabla 2-1 Correspondencia entre el espacio espectral y el Cepstral

Los coeficientes cepstrales describen la periodicidad del espectro. Un pico en el cepstrum indica que la señal es una combinación lineal de múltiplos de la frecuencia del *pitch*. El *pitch* se puede encontrar en el número del coeficiente donde se produce el pico.

El cepstrum es un número complejo y, por tanto, tiene parte real e imaginaria. La parte compleja contiene la información sobre la magnitud y fase inicial del espectro, permitiendo la reconstrucción de una señal. Mientras que su parte real utiliza solamente las magnitudes del espectro.

Los cepstrum reales de una señal $x(n)$ están definidos como

$$c(n) \equiv \frac{1}{2\pi}\int_{\pi}^{-\pi} \ln\left|X\left(e^{j\omega}\right)\right| e^{j\omega n} d\omega \ , \qquad \textbf{(2-23)}$$

donde $X\left(e^{j\omega}\right)$ es la transformada de Fourier de $x(n)$.

De la misma forma, los cepstrum complejos de $x(n)$ se definen como

$$\hat{x}(n) \equiv \frac{1}{2\pi}\int_{\pi}^{-\pi} \ln X\left(e^{j\omega}\right) e^{j\omega n} d\omega \ , \qquad \textbf{(2-24)}$$

donde se usa el logaritmo complejo dado por

$$\hat{X}\left(e^{j\omega}\right) = \ln\left|X\left(e^{j\omega}\right)\right| + j\theta(\omega) \qquad \textbf{(2-25)}$$

y la fase

$$\theta(\omega) = arg\left[X\left(e^{j\omega}\right)\right] \ . \qquad \textbf{(2-26)}$$

Si la señal $x(n)$ es real, el cepstrum real $c(n)$ y el complejo $\hat{x}(n)$ son también señales reales. Por tanto el término cepstrum complejo, no se refiere a que es una señal

compleja, sino que se toma el logaritmo complejo. Para el propósito de este capítulo, cuando se indique cepstrums[1] se referirá a los cepstrums reales.

Si se aplica este análisis a los *frames* de una señal, se pueden calcular los cepstrums con el uso de la DFT. No obstante, el cepstrum complejo obtenido no es exacto, debido a que el logaritmo complejo usado en el cálculo de la DFT, es una versión muestreada de $X(e^{j\omega})$. De esta forma, el resultado de la transformación inversa es una versión con *aliasing* del verdadero cepstrum complejo. Para minimizar el efecto de *aliasing* se necesita un valor grande para N, aproximadamente mayor o igual que 512.

Para poder realizar una comparación con la captación del sonido por el ser humano, hay que tener en cuenta que el sistema auditivo no es constante a lo largo del eje de la frecuencia. Este, puede fácilmente distinguir tonos entre 200 y 250Hz pero no entre 2000 y 2500Hz. Para imitar esta característica espectral se debe usar el análisis espectral con una resolución fija en una escala de frecuencia subjetiva llamada, escala en frecuencia Mel, donde Mel es una unidad de la frecuencia subjetiva. Existe una relación monótona entre la escala Mel y la escala en frecuencia física en Hz (Quariteri, 2002),

$$f_{mel} = 2595 \log_{10}\left(1 + \frac{f(Hz)}{700}\right) . \qquad \textbf{(2-27)}$$

La escala en frecuencia Mel es aproximadamente lineal hasta 1000Hz y logarítmica en frecuencias superiores, Figura 2-11.

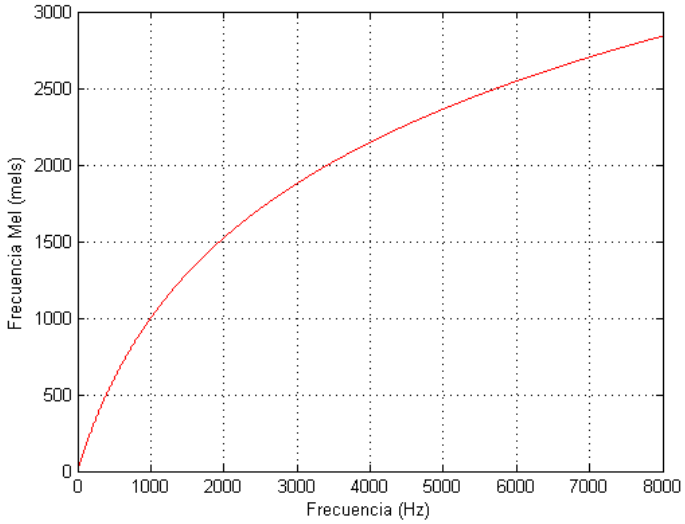


Figura 2-11 Relación entre la frecuencia lineal (Hz) y la frecuencia Mel

Si se representa el cepstrum real en la frecuencia Mel, de una señal analizada en *frames*, se tienen los Coeficientes Mel Cepstral (MFCC, Mel-Frequency Cepstrum Coefficients).

[1] Cepstrums o Ceptra son la forma plurar de cepstrum

Para calcular los coeficientes MFCC se realizan los siguientes pasos (Nieto, 2006):

- Se calcula el espectro de Fourier de la señal analizada en *frames*,

$$X_a(k) = \sum_{k=0}^{N-1} x(n) e^{-j2\pi} \ . \qquad \textbf{(2-28)}$$

- Se calcula la energía por bandas de frecuencia haciendo el espectro obtenido, en el paso anterior, por un banco de filtros. El banco de filtros es una técnica clásica en el análisis espectral, que consiste en representar el espectro de una señal, por la energía logarítmica en la salida de un banco de filtros formado por filtros paso banda a lo largo del eje de las frecuencias. Esta representación es una burda aproximación del espectro de la señal.
  Hay que considerar el fenómeno del enmascaramiento de las frecuencias del sistema auditivo, hecho por el cual un sonido no puede ser percibido por el oído si existe otro muy cercano en frecuencia con un nivel energético baste alto. Considerando este fenómeno, se implementa un banco de filtros con respuesta en frecuencia triangular, superpuestos y con una separación uno del otro por un intervalo constante en frecuencia Mel, Figura 2-12.

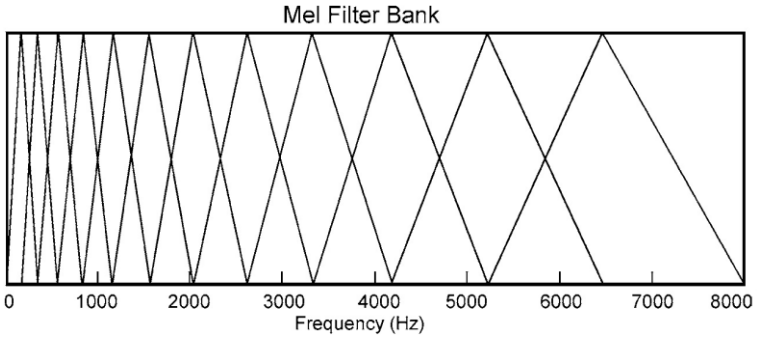


Figura 2-12 Banco de filtros utilizado por Davis y Mermelstein (Nieto, 2006)

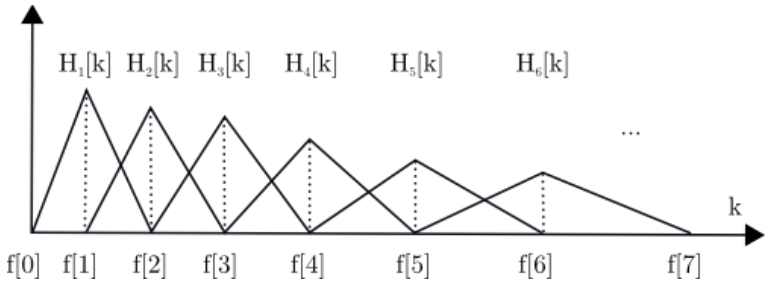


Figura 2-13 Banco de filtros triangulares centrado en frecuencias (Nieto, 2006)

Existen otras alternativas para el diseño del bando de filtros, por ejemplo calcular el promedio del espectro alrededor de la frecuencia central en cada filtro, como se ilustra en la Figura 2-13.

La respuesta en frecuencia de este último tipo de banco de filtros es

$$H_m(k) = \begin{cases} 0 & si \quad k < f(m-1) \\ \dfrac{2(k - f(m-1))}{(f(m+1) - f(m-1))(f(m) - f(m-1))} & si \;\; f(m-1) \le k \le f(m) \\ \dfrac{2(f(m+1) - k)}{(f(m+1) - f(m-1))(f(m+1) - f(m))} & si \;\; f(m) \le k \le f(m+1) \\ 0 & si \;\; k > f(m+1) \end{cases} \quad \textbf{(2-29)}$$

$$S(m) = \ln\left[\sum_{k=0}^{N-1} |X_a(k)|^2 \cdot H_m(k)\right], \quad \textbf{(2-30)}$$

donde $m = 0, 1, 2, \dots, M-1$, siendo M el número de filtros.

- Posteriormente, se obtienen los coeficientes mel-cepstral, aplicando la transformada coseno discreta (DCT) a las salida de los M filtros,

$$c(n) = \sum_{m=0}^{M-1} S(m) \cos\left(\frac{\pi n (m + \frac{1}{2})}{M}\right). \quad \textbf{(2-31)}$$

Para aplicaciones de reconocimiento de vocal, típicamente se usan los primeros 13 coeficientes MFCC.

Es importante notar que la representación de los MFCC ya no es una transformación homomórfica, pero podría serlo si el logaritmo y el sumatorio se intercambian quedando

$$S(m) = \sum_{k=0}^{N-1} \ln(|X_a(k)|^2 H_m(k)) \;. \quad \textbf{(2-32)}$$

2.5. Codificación predictiva lineal LPC

La codificación predictiva línea, LPC, es una de las técnicas más potentes de análisis de señales sonoras, y uno de los métodos más útiles para codificar con baja tasa de bits. Se ha convertido en la técnica predominante para la estimación de los parámetros básicos, como por ejemplo, pitch, formantes y espectro. Su función representa la envolvente espectral de una señal en forma comprimida, utilizando la información de un modelo lineal, con lo cual se proporcionan unas aproximaciones a los parámetros de la voz muy precisas. El nombre de este modelo es debido a la extrapolación del

valor de la siguiente muestra de la señal, $s(n)$, como la suma ponderada de las muestra anteriores $s(n-1)$, $s(n-2)$,…, $s(n-p)$,

$$s'^{(n)} = \sum_{k=1}^{p} a_k \cdot s(n-k)\,. \qquad \textbf{(2-33)}$$

El conjunto de coeficientes óptimo será aquel que haga que el error cuadrático sea mínimo,

$$J = \sum_{n=-\infty}^{\infty} \big(s(n) - s'(n)\big)^2\ , \qquad \textbf{(2-34)}$$

$$J = \sum_{n=-\infty}^{\infty} \left(s(n) - \sum_{k=1}^{p} a_k \cdot s(n-k)\right)^2\ . \qquad \textbf{(2-35)}$$

Estos mínimo se calculan haciendo que las derivadas parciales sean cero, obteniendo un sistema de ecuaciones que permite calcular el conjunto de coeficientes,

$$\frac{\partial J}{\partial a_m} = 0 \qquad 1 \le m \le p\ . \qquad \textbf{(2-36)}$$

En notación matricial, el sistema de ecuaciones se puede escribir en función de la función de autocorrelación como

$$\begin{vmatrix} \varphi(0) & \varphi(1) & \varphi(2) & \dots & \varphi(p-1) \\ \varphi(1) & \varphi(0) & \varphi(1) & \dots & \varphi(p-2) \\ \varphi(2) & \varphi(1) & \varphi(0) & \dots & \varphi(p-3) \\ \dots & \dots & \dots & \dots & \dots \\ \varphi(p-1) & \varphi(p-2) & \varphi(p-3) & \dots & \varphi(0) \end{vmatrix} * \begin{vmatrix} a_1 \\ a_2 \\ a_3 \\ \dots \\ a_n \end{vmatrix} = \begin{vmatrix} \varphi(1) \\ \varphi(2) \\ \varphi(3) \\ \dots \\ \varphi(p) \end{vmatrix}\ . \qquad \textbf{(2-37)}$$

Siendo φ la función de autocorrelación, que como se ha visto anteriormente

$$\varphi(m) = \sum_{i=0}^{N-1-m} x(i) \cdot x(i+m) \quad m = 0, 1, \dots p\ , \qquad \textbf{(2-38)}$$

donde p es el número de coeficientes LPC que se desea calcular. En el coeficiente $\varphi(0)$ se acumula la energía de la señal analizada.

Para la obtención de los coeficientes LPC se desarrolla el algoritmo de Levison-Durbin, que resuelve la matriz optimizando las operaciones (Levinson, 1947)

$$E_0 = \varphi(0)\ , \qquad \textbf{(2-39)}$$

$$k_i = \frac{r(i) - \sum_{j=1}^{L-1} \alpha_j^{i-1} \cdot \varphi(|i-j|)}{E_{i-1}} \quad 1 \le i \le p \ ,$$

$$\alpha_i^i = k_i \ ,$$

$$\alpha_j^i = \alpha_i^{i-1} - k_i \cdot \alpha_{i-j}^{i-1} \ ,$$

$$E_i = \left(1 - {k_i}^2\right) \cdot E_{i-1} \ .$$

El número de coeficientes determina la resolución con la que el análisis LPC va a representar la envolvente espectral de la señal. Un valor reducido implica poca resolución, pero un valor excesivo implica cierta distorsión debido a que no sólo se tiene en cuenta la envolvente espectral, sino la estructura final del mismo. En la Figura 2-14 se puede observar como varia la envolvente espectral en función del número de coeficientes.

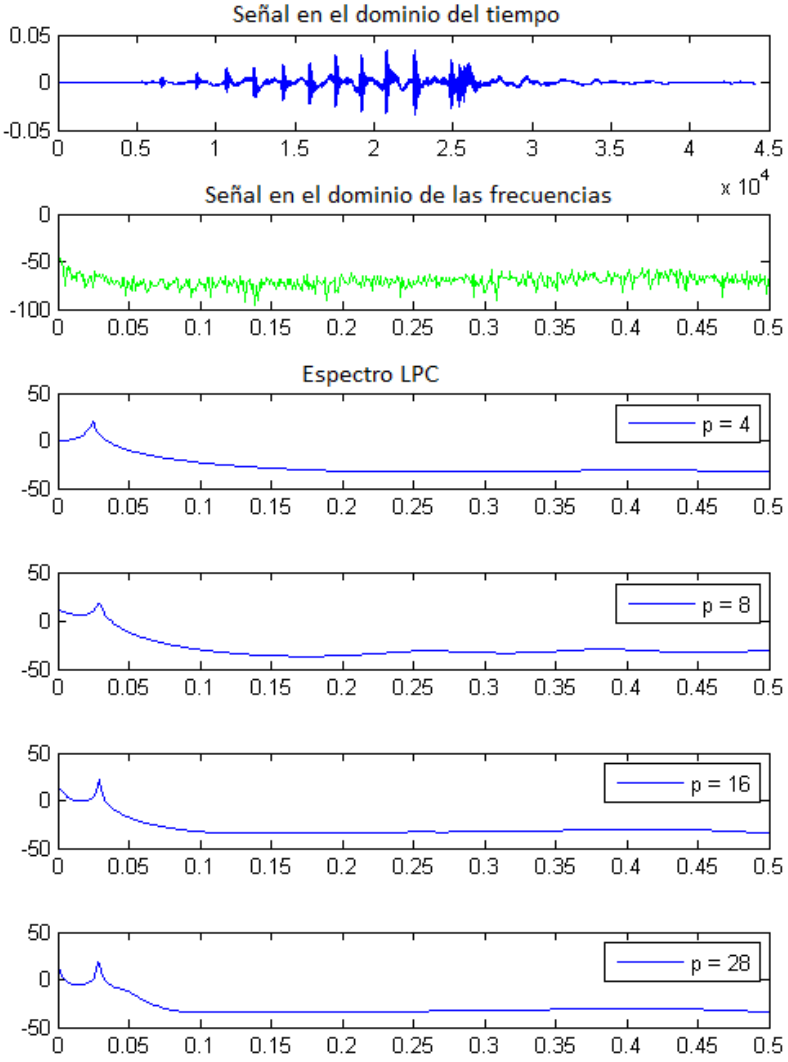


Figura 2-14 Variación del espectro LPC en función del número de coeficientes

Los formantes se pueden definir como las bandas de frecuencia donde se concentra la mayor parte de la potencia sonora de un sonido. Estos formantes se visualizan como máximos relativos en la envolvente espectral. Utilizando la técnica LPC tal como se describe en (Snell & Milinazzo, 1993) se puede calcular los formantes de una señal.

Este cálculo puede resumirse en los siguientes pasos:

- A partir de los datos del sonido $s(n)$ se modela su fuente de producción, utilizando para el desarrollo de esta tesis un modelo del aparato fonador del anuro. Este modelo toma la forma de: un generador de sonido armónico (a la frecuencia fundamental); un generador de ruido aleatorio; y un filtro digital $H(z)$. El filtro puede expresarse como una ganancia G y un filtro inverso $A(z)$

$$\mathrm{H(z)} = \frac{\mathrm{G}}{\mathrm{A(z)}} \ . \qquad \textbf{(2-40)}$$

  El filtro inverso $\mathrm{A(z)}$ toma la forma

$$\mathrm{A(z)} = 1 - \sum_{\mathrm{k=1}}^{\mathrm{p}} \mathrm{a_k\, z^{-k}} \ . \qquad \textbf{(2-41)}$$

  El grado p del polinomio $\mathrm{A(z)}$ determina la precisión del modelo.

- Se calculan las raíces z_i del polinomio $A(z)$. Estas raíces son, en general, números complejos que pueden expresarse en la forma

$$\mathrm{z_i} = \mathrm{r_i e^{j\theta_i}} \ . \qquad \textbf{(2-42)}$$

- La frecuencia del formante se calcula mediante la expresión

$$\mathrm{f_i} = \frac{\mathrm{f_s}}{2\pi}\theta_\mathrm{i} \ , \qquad \textbf{(2-43)}$$

  en la que $\mathrm{f_s}$ representa la frecuencia de muestreo del sonido.

- El ancho de banda del formante se calcula mediante la expresión

$$\mathrm{B_i} = \frac{\mathrm{f_s}}{\pi}\mathrm{Ln\ r_i} \ . \qquad \textbf{(2-44)}$$

- Se calcula el error $\varepsilon(n)$ entre el sonido real $s(n)$ y la estimación $\tilde{s}(n)$ obtenida mediante el modelo LPC,

$$\mathrm{\varepsilon(n) = s(n) - \tilde{s}(n) = s(n) - \sum_{k=1}^{p} a_k\, s(n-k)} \ . \qquad \textbf{(2-45)}$$

- Se calcula la función de autocorrelación del error $\varepsilon(n)$ como

$$\varphi(\mathrm{k}) = \sum_{\mathrm{m}=-\infty}^{+\infty} \varepsilon(\mathrm{m})\, \varepsilon(\mathrm{m}+\mathrm{k}) \ . \qquad \textbf{(2-46)}$$

- Se calculan los picos φ_i de la función de autocorrelación del error y el valor k_i para el que ocurren.
- El tono (*pitch*) del sonido se calcula como

$$Pitch = \frac{\mathrm{f_s}}{\mathrm{k_1}} \ , \qquad \textbf{(2-47)}$$

expresión en la que $\mathrm{k_1}$ es la posición del primer máximo de la función de autocorrelación del error.

La Figura 2-15 recoge el resultado del análisis de formantes aplicando técnicas LPC. En ella pueden apreciarse los espectros para dos *frames* distintos de una señal sonora. Sobre cada espectro se traza la estimación obtenida por LPC, la posición de los formantes y su ancho de banda. En general, los formantes más significativos son los primeros.

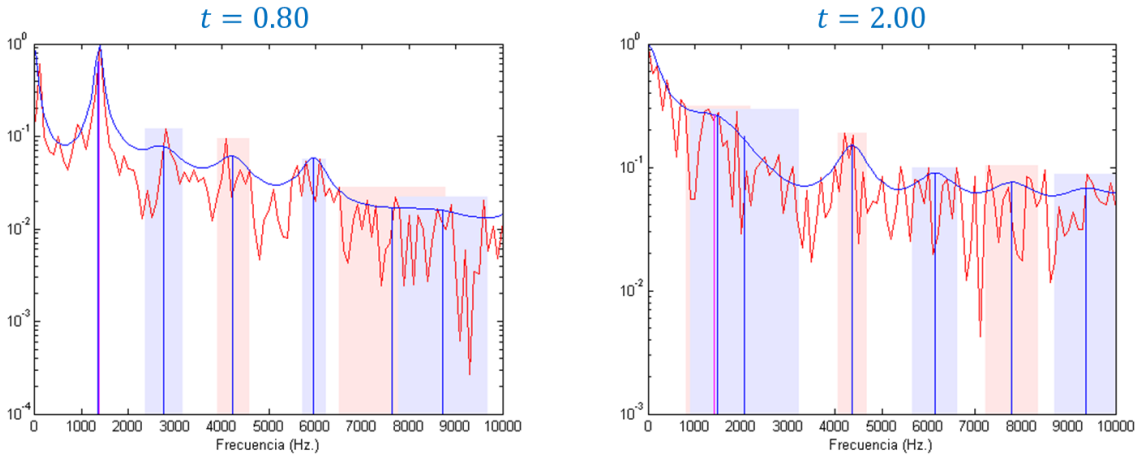


Figura 2-15 Análisis de formantes por técnicas LPC

CAPÍTULO 3. CARACTERIZACIÓN DE SONIDOS MEDIANTE EL ESTÁNDAR MPEG-7

Se entiende por caracterización: La determinación de los atributos peculiares de alguien o de algo, de modo que claramente se distinga de los demás (Real Academia Española, 2014).

En el campo de la clasificación e identificación de sonidos el primer paso debe ser la caracterización de los sonidos. Como se ha visto en el capítulo anterior, existen un gran número de técnicas usadas en la caracterización de los sonidos. Dependiendo de las aplicaciones, tipos de sonidos y, en muchos casos, la elección de los autores, estas técnicas carecen de homogeneidad, tanto en la selección de características como incluso en la definición del parámetro.

Dada la aparente falta de homogeneidad en las técnicas utilizadas parece conveniente realizar el uso de características normalizadas como los *Mel Frequency Ceptral Coefficiente* (MFCCs). Estas características, ampliamente utilizadas en los procesos de clasificación de sonidos, están unívocamente definidas e incluso estandarizadas (ETSI, 2002), por lo que son una buena solución para hacer frente a la heterogeneidad descrita anteriormente. Mientras los MFCC son un conjunto de características estándar que pueden ser aplicados a la clasificación de sonidos, ofrecen una visión unidimensional al derivarse todas las características del mismo enfoque: el cepstrum de la señal de sonidos. Aunque el uso de los coeficientes MFCC tiene la ventaja de la normalización, limitan la búsqueda de rasgos semánticos más expresivos.

Una alternativa entre los distintos estándares existentes, es el estándar MPEG-7, de la Organización Internacional para la Estandandarización (ISO/IEC) (ISO/MPEG, 1998) y desarrollado por el grupo MPEG (*Moving Picture Experts Group*). El grupo MPEG se encarga de desarrollar normas para la representación codificada del audio y video digital y datos relacionados.

La cantidad de información audiovisual disponible en formato digital está alcanzando cifras inconmensurables gracias a las nuevas tecnologías de comunicación y al uso masivo de Internet en la sociedad. El crecimiento de este tipo de información parece no tener límites. El valor de esta información depende en buena medida de lo sencillo que sea encontrarla, recuperarla, acceder a ella y gestionarla (Vicente, 2005). De ahí el interés de este tipo de normas, que normalizan un conjunto de parámetros confiriendo carácter industrializador.

El MPEG7 en comparación con los MFCCs presenta una mayor riqueza semántica como se verá a lo largo de este capítulo.

3.1. Objetivo y estructura de la norma

MPEG-7, formalmente denominado "Interfaz para la descripción de contenido multimedia (*Multimedia Content Description Interface*)", es un estándar para la descripción de material multimedia: habla, audio, vídeo, imágenes y modelos 3D (ISO/MPEG, 1998).

No está dirigido a ninguna aplicación en particular. Por el contrario, los elementos que MPEG-7 estandariza apoyan un amplio abanico de aplicaciones. Es bastante diferente de los otros estándares MPEG, porque no define una forma de representar datos con el objetivo de reconstruirlos fielmente, como hizo MPEG-1, 2 y 4 (Koenen & Pereira, 2000). Su objetivo se aparta de estos estándares, que se ocupan principalmente de la representación del contenido (codificación), y añade una capa semántica por encima, dependiendo de ellos (u otros estándares similares) para proporcionar acceso a los contenidos mismos.

Ofrece un conjunto completo de herramientas de descripción audiovisual: elementos de metadatos, su estructura y relaciones, que son definidos por el estándar en forma de descriptores y esquemas de descripción. Estas herramientas sirven para crear un conjunto de descripciones formados por esquemas de descripción y sus descriptores correspondientes, que constituirá la base para aplicaciones permitiendo el necesario acceso eficaz y eficiente (búsqueda, filtrado y navegación) al contenido multimedia. Esta es una tarea difícil, dado el amplio espectro de necesidades y aplicaciones multimedia, selectiva y amplio número de funciones audiovisuales de importancia en este contexto (Martínez, 2004).

Como todos los estándares producidos por MPEG, la especificación de MPEG-7 está orientada a un modelo de decodificador. Es decir, MPEG-7 detalla la sintaxis y semántica de las descripciones, de tal forma que cualquier decodificador acorde con la norma sea capaz de interpretar una descripción MPEG-7. Por tanto, lo que luego el descodificador haga con ella (filtrado, búsqueda, navegación o acceso), como el método usado en el codificador para generarla no están especificados, dejando espacio para desarrollos o mejoras posteriores, así como para la posibilidad de ofrecer

valor añadido por parte de empresas o instituciones que produzcan aplicaciones basadas en MPEG-7 (Núñez & Fernández, 2002).

El estándar MPEG-7 (ISO/IEC 15938) ha sido dividido en secciones, que tratan con diferentes aspectos de la descripción del material multimedia (Chang, Sikora, & Puri, 2001; Martínez, 2004):

- **Parte 1:** ***MPEG-7 Systems***. Esta sección abarca aspectos globales de acceso a la información, mecanismos de transmisión, almacenamiento, sincronización del contenido y descripción de la información, formatos de fichero, multiplexado de descripciones y calidad de servicio. Otro aspecto que engloba son las herramientas relacionadas con la gestión y protección de la propiedad intelectual.
- **Parte 2:** ***MPEG-7 Description Definition Language***. Describe el lenguaje de definición de descripciones. Se basa en el lenguaje XML de metadatos en un intento de favorecer la interoperabilidad y la creación de aplicaciones. Para poder dar al estándar mayores funcionalidades se han añadido algunas extensiones de las que *XML Schema* puro carece, ya que no fue diseñado para describir material audiovisual. Con el fin de evitar un problema de exceso de datos se ha creado un compresor llamado BiM (*Binary Format for MPEG-7*). Entre una de sus propiedades, este compresor presenta la ventaja de ser más robusto que XML ante los errores de transmisión (Avaro & Salembier, 2001) y cuenta con una alta tasa de compresión al eliminar la redundancia estructural del documento. Cada documento es dividido y transmitido por partes. A cada parte del documento se le llama unidad de datos y llevará la información completa de un descriptor. De esta forma, en el caso de modificarse la información de un descriptor, sólo se tendrá que transmitir la información de éste y no de todo el documento, lo que permite ahorrar ancho de banda.
- **Parte 3:** ***MPEG-7 Visual***. Comprende la descripción de elementos visuales, por sus propiedades espaciales (color, textura o forma), propiedades temporales (movimiento o actividad), localización espacio-temporal (posición o trayectoria) o por características específicas (reconocimiento de rostros).
- **Parte 4:** ***MPEG-7 Audio***. Especifica los métodos de descripción de contenidos de audio basados en sus características de bajo nivel (timbre, ritmo, distribución espacial, temporal y espectral o fuente de sonido), y herramientas de alto nivel como: la transcripción a texto (para voz o letras de canciones), reconocimiento de sonido e indexación.
- **Parte 5:** ***MPEG-7 Multimedia Description Schemes***. Sección que especifica las herramientas de descripción que son genéricas, así como la organización de los elementos individuales, de forma tal que se puedan aplicar a la creación de descripciones completas de la estructura física y del contenido semántico del material (Salembier & Smith, 2001). Dentro de MDS se encuentran también los

esquemas de descripción para los usuarios de los servicios audiovisuales, que permiten especificar las preferencias personales y los patrones de uso para ofrecer servicios de personalización.

- **Parte 6: *MPEG-7 Reference Software****: the eXperimentation Model*. Pretende proporcionar una implementación de referencia de las partes pertinentes de la norma MPEG-7, y es conocida como XM *Experimentation Software* o Modelo de Experimentación. El XM permite realizar pruebas de funcionamiento de las partes desarrolladas, asimismo sirve de orientación para la construcción de sistemas MPEG-7. Las aplicaciones XM están divididas en dos tipos: las aplicaciones servidores (extracción) y las aplicaciones cliente (búsqueda, filtrado y/o transcodificación).
- **Parte 7: *MPEG-7 Conformance***. Pretende proporcionar directrices y procedimientos para probar que las implementaciones de MPEG-7 se ajustan a las normas establecidas.
- **Parte 8: *MPEG-7 Extraction and use of the descriptions***. Agrupa material informativo sobre la extracción y uso de algunas de las herramientas de descripción.
- **Parte 9: *MPEG-7 Profiles***. Recoge distintos perfiles y niveles estandarizados para MPEG-7, especificados a través de las distintas partes del estándar ISO/IEC 15939.
- **Parte 10: *MPEG-7 Scheme Definition***. Recoge todos los esquemas MPEG-7, reuniéndolos desde diferentes estándares, correcciones y enmiendas.

Profundizando en el lenguaje de descripción que usa MPEG-7, los principales elementos del estándar son (Pereira, 1996):

- **Descriptores (D):** son las representaciones de características que definen la sintaxis y la semántica de cada representación característica. Estas representaciones no serán más que un determinado conjunto de metadatos MPEG-7. Se utilizan para describir los siguientes tipos de información:
    - Características audiovisuales de bajo nivel, como son el color, la textura, el movimiento, la energía, etc.
    - Características de alto nivel de objetos semánticos, eventos y conceptos abstractos.
    - Procesos de gestión del contenido.
    - Información sobre el contenido.
- **Esquemas de Descripción (DS)**: especifican la estructura y semántica de las relaciones entre sus componentes. Estos componentes pueden ser tanto D's como DS's.
- **Lenguaje de Definición de Descripciones (DDL)**: proporciona las herramientas para permitir la creación de nuevos DS's y D's. También permite la extensión y modificación de los DS's existentes.

- **Herramientas del sistema**: permiten el multiplexado de diferentes descripciones así como la sincronización de las mismas con el contenido. Se ocupan de los mecanismos de transmisión y de representaciones codificadas (tanto textuales como binarias) para almacenarlas y transmitirlas de forma eficiente. También incluyen los procesos de gestión y protección de la propiedad intelectual.

En la Figura 3-1 se puede observar las relaciones existentes entre los elementos anteriormente indicados.

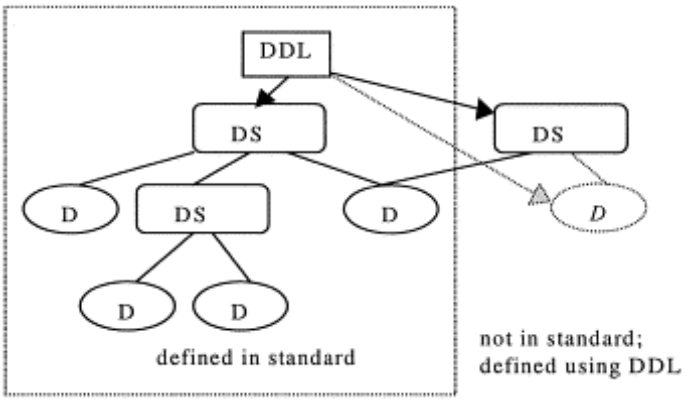


Figura 3-1 Elementos del estándar MPEG-7 (Koenen & Pereira, 2000)

En principio, la gran mayoría de los D's son utilizados para describir características de bajo nivel y podrán ser extraídos de los contenidos de una forma automática. Por el contrario, será necesaria la intervención de un agente humano para extraer los D´s de alto nivel.

Los DS's de MPEG-7, pueden ser clasificados en tres grandes grupos: audio, dominio visual y descripción genérica de datos multimedia. En este último grupo se describen los metadatos relacionados con la creación, producción, uso y gestión de los datos multimedia; así como el contenido multimedia a distintos niveles de abstracción (incluyendo estructura de la señal, características modelos y semántica).

Los esquemas de descripciones multimedia (DMS) utilizados en MPEG-7 se dividen en seis partes (Martínez, 2002):

1. Elementos básicos y herramientas. Los elementos básicos se pueden agrupar en dos grandes grupos: información temporal y anotaciones textuales. Los DS's que describen **información temporal** se basan en el estándar ISO8601 (*Data elements and interchange formats — Information interchange — Representation of dates and times*) que especifica una notación estándar utilizada para representar instantes, intervalos e intervalos recurrentes de tiempo evitando ambigüedades. Esta notación facilita la migración entre distintas plataformas. Estos elementos

representan la información temporal del contenido o "*streams*". Sirven para describir el tiempo a nivel de muestras del contenido multimedia, como pueden ser el periodo de muestreo de una señal digital. Las **anotaciones textuales** aportan una serie de construcciones básicas como texto libre, texto estructurado y anotaciones estructuras relacionadas. Se utiliza para la descripción textual del contenido multimedia. Además de los elementos básicos, se dispone de un conjunto de **herramientas de esquema** que facilitan la creación y agrupación de las descripciones.

2. Descripción del contenido. Se utilizan DS's que describen información sobre la **estructura** (regiones, cuadros de vídeo, segmentos de audio) y la **semántica** del contenido multimedia (objetos y eventos). Los DS's de estructura se organizan en torno a un segmento o "*segment*", que representa las características espaciales y temporales del contenido. El segmento puede organizarse en una estructura jerárquica que permita el acceso, o indexado, del contenido para facilitar las búsquedas. Los segmentos se describen utilizando DS's del tipo anotaciones textuales y D's de la parte de audio y vídeo (Salembier, Llach, & Garrido, 2002).

3. Gestión del contenido. Se utilizan DS's que describen la información sobre la creación, producción, codificación, almacenamiento, formatos de archivos y uso de los contenidos multimedia. Estos DS's se agrupan en: ***creation infomation*** que describen la información sobre la creación del contenido: autor, título, lugares, fechas de creación, género, tema, propósito, relaciones con otros contenidos multimedia, lenguaje...; ***usage infomartion*** que describen el uso que se le puede dar al contenido multimedia, como puede ser los derechos de autor, disponibilidad e información financiera; y ***media information*** que describen información sobre el almacenamiento del contenido, la compresión, codificación y formato de los datos.

4. Organización del contenido. Se utilizan DS's que permiten organizar y formar colecciones de contenido multimedia. Permite describir una colección como un todo basándose en propiedades comunes de entre todos los contenidos.

5. Navegación y acceso. Se utilizan DS's para facilitar la navegación y obtención del contenido multimedia.

6. Interacción con el usuario. Describe las preferencias del usuario para, por ejemplo, facilitar la personalización del acceso y presentación del

contenido. También permite registrar las acciones realizadas por el usuario en el sistema.

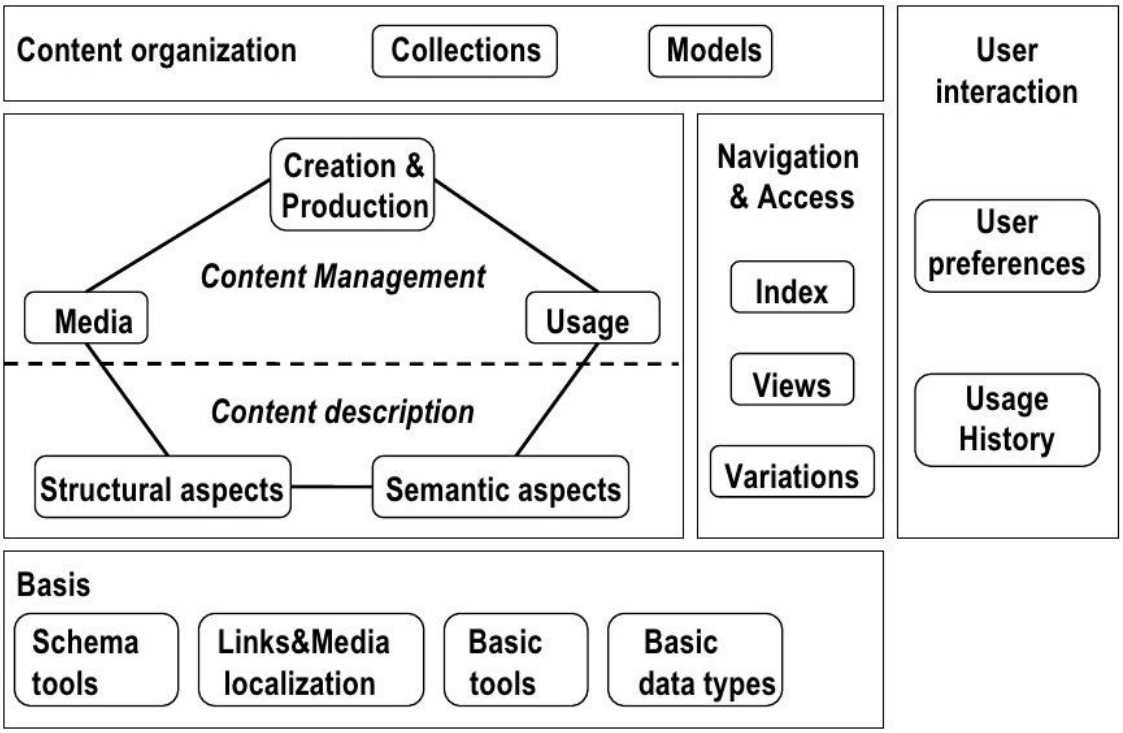


Figura 3-2 Visión general de los esquema de descripción (DS's) (Day & Martinez, 2001)

3.2. Definición y determinación de parámetros de audio

Como se ha comentado anteriormente, la parte 4 del estándar MPEG-7 es la que se encarga de la descripción de los contenidos de audio. Los descriptores de audio se dividen en dos áreas generales: descripciones de alto nivel (*HLDs*) y descripciones de bajo nivel (*LLDs*) que realizan una descripción genérica del audio (Quackenbush & Lindsay, 2001).

Una manera de realizar descripciones de una señal de audio bajo el estándar MPEG-7 (ISO, 2001) es extrayendo características en intervalos regulares, los *frames*. Las características pueden ser escalares y vectoriales y para cada caso existe un descriptor tipo, *AudioLLDScalarType* y *AudioLLDVectorType*, respectivamente. Ambos tipo de datos pueden ser instanciados como valores muestreados en un descriptor *ScalableSeries,* que es una forma estandarizada de representar una serie de características *LLDs* (escalares o vectoriales) extraídas de los *frames* de sonido a intervalos de tiempo regulares. Esta serie puede describirse: completa o de forma escalada. En este último caso, la serie original se descompone en sub-secuencias consecutivas de las muestras. En la Figura 3-3 se puede observar la relación entre los distintos tipos de estructuras que se definen en el estándar. Existen tipos de datos abstractos (*AudioDType* y *AudioDSType*) que son definidos en la parte 5 de la norma MPEG-7.

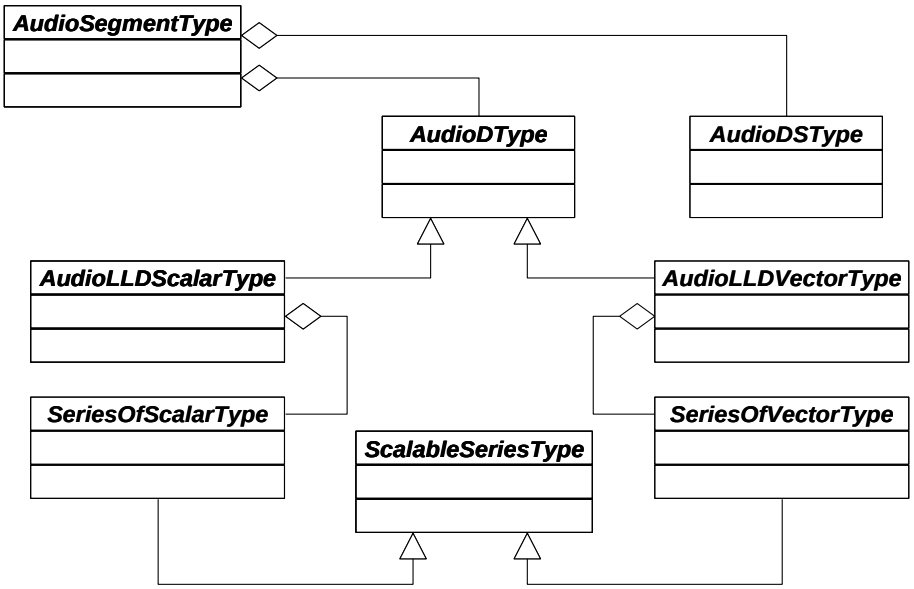


Figura 3-3 Relaciones entre distintos tipo de estructuras para la descripción de audio (ISO, 2001)

En este apartado se realizará un exhaustivo repaso de los puntos de la norma que tienen relación con la descripción a bajo nivel de los audios. De entre todos los parámetros de bajo nivel que la norma define, para el desarrollo de esta tesis y sus siguientes capítulos, sólo se tendrán en cuenta los parámetros basado en *frames*.

3.2.1 Clases escalables

Los elementos que se describen en este punto no son parámetros. Son clases que ofrecen una estructura para la representación de los distintos parámetros, de cara al almacenamiento y posterior reproducción y tratamiento. Aunque queda fuera de las necesidades de la tesis, se estudian para tener una visión global de la norma.

3.2.1.1 Series escalables - *ScalableSeriesType*

Como se puede observar en la Figura 3-3 *ScalableSeriesType* es un tipo abstracto heredado por *SeriesOfScalarType* y *SeriesOfVectorType*. Sus atributos definen las dimensiones y escala de la serie:

- Escalado (*scaling*): específica cómo se ha escalado la señal original. En caso de no escalar la señal original se omite este atributo.
- Número de muestras (*totalNumOfSamples*): indica el número total de muestras de la señal original sin tener en cuenta el escalado, si lo tuviese.
- Radio (*ratio*): es un número que indica cuantas muestras contiene cada *frame*, como se puede observar en la Figura 3-4, cada *frame* (asociada con el número de índice) comprende un radio que es igual al número de muestras originales contenidas dentro de esta misma.
- Número de elementos (numOfElements): índica el número total de *frames* de la serie escalable. Cuando no se escala la señal el número de elementos es igual al número de muestras.

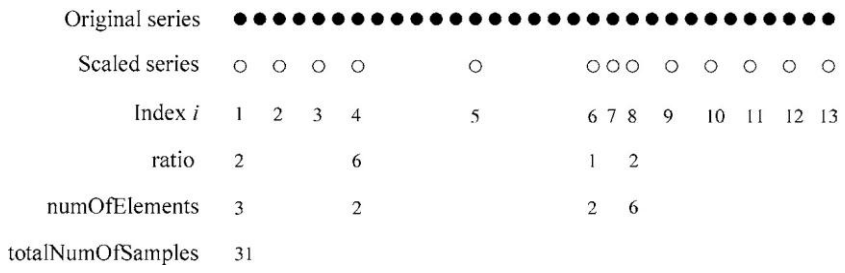


Figura 3-4 Ilustración de *ScalableSeries* (ISO, 2001)

En la Figura 3-4 se muestra una ilustración de una *ScalableSeries* que consta 31 muestras que están condensadas en 13 muestras escaladas. Las tres primeras muestras escaladas, cada una, contienen dos muestras originales, las siguientes dos contienen seis muestras originales, etc. La última serie escalable tiene un radio de dos, aunque sólo contiene una muestra original. Esta situación no ocasiona problemas pues es detectable mediante

$$numElementos = \frac{totalNumMuestras}{Radio} \ . \qquad \textbf{(3-1)}$$

3.2.1.2 Serie de valores escalares - *SeriesOfScalarType*

Cuando las propiedades a describir son de tipo escalar se usa el descriptor *SeriesOfScalarType* cuyos atributos son:

- Bruto *(Raw)*: contiene la serie original en caso de que no se haya escalado dicha señal. Como se mencionó anteriormente, el atributo *scaling* indica si existe una operación de escalado.
- Peso (*Weight*): es opcional y se utiliza para controlar el escalado,

$$\bar{w}_k = \frac{1}{N} \sum_{i=1+(k-1)N}^{k \cdot N} w_i \ . \qquad \textbf{(3-2)}$$

- Mínimo, máximo y media (*Min*, *Max* y *Mean*): son tres vectores que caracterizan los valores de un *frame* de la serie escalada. En caso de no escalar la señal original se omiten estos atributos. Para *Min* se toma el valor mínimo del total de muestras de un *frame*,

$$m_k = min_{i=1+(k-1)N}^{k \cdot N} [x_i] \ . \qquad \textbf{(3-3)}$$

Para *Max* se toma el valor máximo de las mismas muestras,

$$M_k = max_{i=1+(k-1)N}^{k \cdot N} [x_i] \ . \qquad \textbf{(3-4)}$$

Y para *Mean* se toma el valor promedio,

$$\bar{x}_k = \frac{1}{N} \sum_{i=1+(k-1)N}^{k \cdot N} x_i \ . \quad \textbf{(3-5)}$$

- Varianza (*Variance*): es un vector que contiene la varianza calculada de cada *frame*. En caso de no escalar la señal original se omite este atributo. En caso de existir los pesos (*weight*). estos son considerados,

$$z_k = \frac{1}{N} \sum_{i=1+(k-1)N}^{k \cdot N} x_i^2 - \bar{x}_k^2 \ . \quad \textbf{(3-6)}$$

- Aleatorio (*Random*): es un vector donde cada elemento contiene un valor aleatorio del *frame* correspondiente. En caso de no escalar la señal original se omite este atributo.
- Primero (*First*): es un vector que contiene el valor de la primera muestra de cada *frame*. En caso de no escalar la señal original se omite este atributo.
- Último (*Last*): es un vector que contiene el valor de la última muestra de cada *frame*. En caso de no escalar la señal original se omite este atributo.

En las formulas anteriores k es el índice de la serie escalable, i el índice de la serie original y N es el número de muestras contenidas en cada muestra escalada.

3.2.1.3 Serie de valores vectoriales - *SeriesOfVectorType*

Cuando las propiedades a describir son de tipo vectorial se usa el descriptor *SeriesOfVectorType* cuyos atributos son:

- *Raw*, *Min*, *Max*, *Mean*, *Random*, *First*, *Last*, *Variance* y *Weight*: son atributos iguales a los del descriptor *SeriesOfScalarType*.
- Tamaño del vector (*VectorSize*): es el número de elementos de cada vector dentro de la serie.
- Covarianza (*Covariance*): es una matriz de dimensión 3, donde el número de filas es igual al *numOfElements*,

$$\sigma_k^{jj'} = \frac{1}{N} \sum_{i=1+(k-1)N}^{k \cdot N} \left(x_i^j - \bar{x}^j\right)\left(x_i^{j'} - \bar{x}^{j'}\right) \ . \quad \textbf{(3-7)}$$

  En caso de no escalar la señal original se omite este atributo.
- *VarianceSummed*: series de la suma de los coeficientes de varianza de los grupos de la muestra. En caso de no escalar la señal original se omite este atributo. El tamaño del vector debe ser igual *numOfElements*,

$$z_k = \frac{1}{N}\sum_{j=1}^{D}\sum_{i=1+(k-1)N}^{k\cdot N}\left(x_i^j - \bar{x}_i^j\right)^2 \ . \quad \textbf{(3-8)}$$

- *MaxSqDist*: es una serie de coeficientes de distancia cuadrática máxima (*MSD*), que representa un límite superior de la distancia entre grupos de muestras y su media. En caso de no escalar la señal original se omite este atributo. El tamaño de la matriz debe ser igual *numOfElements*,

$$MSD_k = max_{i=1+(k-1)N}^{k\cdot N}\|x_i - \bar{x}_k\|^2 \ . \quad \textbf{(3-9)}$$

En las formulas anteriores D es el tamaño de cada vector, j el índice de cada vector y $\bar{x}_i^j$ la media de las N muestras.

3.2.2 Clases para descriptores de bajo nivel

Los descriptores de bajo nivel (*LLDs*) representan las variaciones de las propiedades del audio en tiempo y frecuencia y pueden clasificarse en los siguientes grupos (Hyoung Gook Kim, Moreau, & Sikora, 2005):

- Descriptores básicos. Muestran la forma de onda y la potencia del audio en el dominio del tiempo: forma de onda y potencia sonora.

- Descriptores espectrales básicos. Representan el análisis en tiempo y frecuencia que describen el espectro del audio en términos de su envolvente, centroide, propagación y llanura: envolvente espectral, centroide de potencia, dispersión espectral de potencia y planitud del espectro.

- Descriptores de parámetros de la señal. Describen la frecuencia fundamental de una señal de audio y las frecuencias armónicas, por lo que sólo se utilizan en señales periódicas o cuasi-periódicas: armonicidad y frecuencia fundamental.

- Descriptores temporales de timbre. Se aplican en segmentos de sonidos donde se requiere conocer un timbre o tono característico: tiempo de ataque y centroide temporal.

- Descriptores espectrales de timbre. Representan características espectrales en un espacio lineal de frecuencia. Utilizados en la percepción de timbres musicales: centroide armónico, desviación armónica, dispersión armónica, variación armónica y centroide espectral.

- Descriptores de base espectral. Representan proyecciones en dimensiones más simples de un espacio espectral con dimensiones más complejas. Utilizados en

la clasificación de sonidos e indexado descriptivos: bases del espectro y proyección del espectro.

Además de estos descriptores, se añade un descriptor más para el silencio.

Estos descriptores, al igual que las del punto anterior, no son parámetros sino clases que permiten modelar los datos en bases a los distintos parámetros.

Todos los descriptores de audio de bajo nivel se definen como subtipos de *AudioLLDScalarType* o *AudioLLDVectorType* como puede observarse en la Figura 3-5.

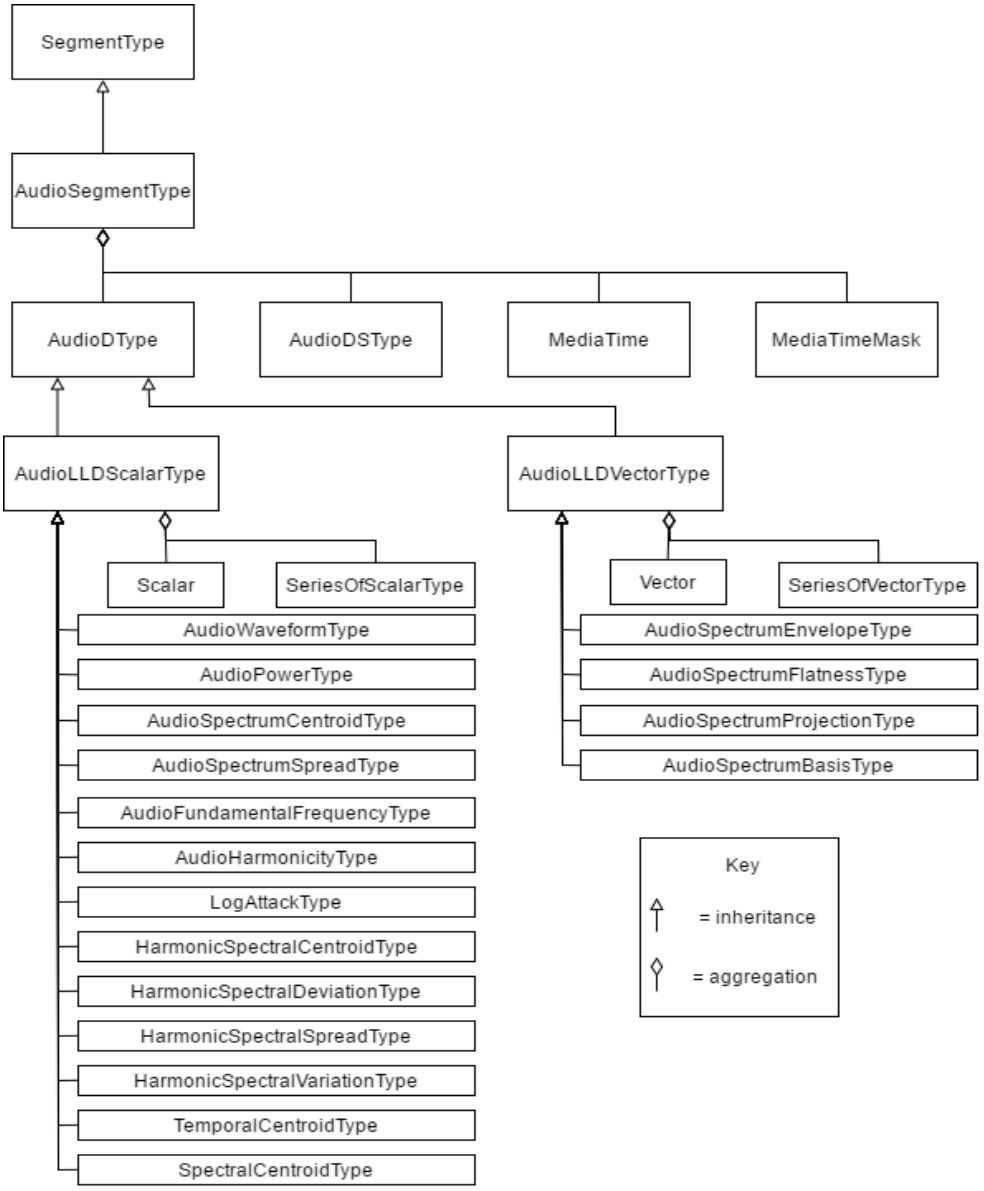


Figura 3-5 Diagrama de clases para los descriptores de bajo nivel de audio (MPEG, 2005)

Hay dos estrategias de descripción utilizando estos tipos de datos: resumen de un solo valor y la descripción en *frames*. Estas dos estrategias de descripción se ponen a disposición para los dos tipos de datos: *Scalar* / *SeriesOfScalarType* y *Vector* / *SeriesOfVectorType*, y se implementan como una opción en DDL (MPEG, 2005).

Al usar descripciones resumidas (que contiene un único valor escalar o vectorial) no existen métodos normativos para calcular el valor. Sin embargo, cuando se utiliza descripciones basadas en *frames*, los valores se calcularán utilizando los métodos de escala proporcionados por los *SeriesOfScalarType* y *SeriesOfVectorType* descriptores, como los atributos de mínimo, máximo y media.

Las clases AudioLLDScalar y AudioLLDVector son abstractas y por lo tanto nunca son instanciadas directamente.

3.2.2.1 AudioLLDScalarType y AudioLLDVectorType

Estos descriptores tienen como atributos:

- *Scalar* o *Vector*. El valor escalar o vectorial según el tipo de dato.
- *SeriesOfScalar* o *SeriesOfVector*. Los valores de *frame* según el tipo de dato.
- *Hopsize*. Como se explicó en el capítulo 2, el tiempo entre dos *frames*. El valor por defecto indicado en la norma es de 10 milisegundos. En caso de no usarse el valor por defecto debería usarse un múltiplo o submúltiplo del valor por defecto para asegurar la compatibilidad.

3.2.3 Parámetros temporales

3.2.3.1 Forma de onda - AudioWaveformType

El descriptor *AudioWaveformType* describe la envolvente de la señal utilizando los valores mínimos y máximos de cada *frame* de la señal. Estos valores son declarados con las variables: *minRange* y *maxRange*, con el valor mínimo y máximo de la señal para cada *frame* respectivamente.

Este descriptor permite mostrar de forma económica una forma de onda del audio. Por ejemplo, una aplicación de edición de sonido puede mostrar un resumen de un archivo de audio de inmediato sin el procesamiento de los datos de audio. Cualquiera que sea el número de muestras, la forma de onda puede visualizarse utilizando un pequeño conjunto de valores que representan los extremos (*min* y *max*) de los *frames*. Los valores mínimos y máximos se almacenan como serie temporal escalable. También pueden ser utilizados para la comparación rápida entre formas de onda.

Este parámetro no está basado en *frames* y por lo tanto no será utilizado en la tesis.

3.2.3.2 Potencia sonora - *AudioPowerType*

Describe de manera temporal la potencia instantánea de cada *frame*. La potencia instantánea se calcula tomando el cuadrado de la forma de onda de cada *frame*, que se promedian en intervalos de tiempo correspondiente al *hopSize*.

La potencia instantánea es una medida útil de la amplitud de una señal como una función del tiempo. En asociación con los descriptores AudioSpectrumCentroid y AudioSpectrumSpread, AudioPower proporciona una descripción económica del

espectro de potencia (la difusión de la energía sobre el rango espectral especificada por el centroide y propagación) que puede ser comparado con la frecuencia del espectro en escala logarítmica. Otra posibilidad es almacenar potencia instantánea a alta resolución temporal, en asociación con un espectro de potencia de alta resolución espectral en baja resolución temporal, para obtener una representación "barata" del espectro de potencia que combina tanto la resolución espectral y temporal.

La potencia sonora, tal y como se ha definido, abarca toda la banda de frecuencia y, en adelante, para referirse a este parámetro se hará como potencia total. Está puede expresarse en valor absoluto ("watios") o en dB con respecto a la potencia unidad. También puede expresarse con respecto a un valor de potencia de referencia. Para este valor de referencia suele utilizarse el valor máximo o el valor mínimo de la grabación. En ocasiones para evitar un pico de máximo o de mínimo, se referencia al percentil 95 o al percentil 5 del valor de la potencia.

Sin embargo, dada la naturaleza de los audios que se han usado en desarrollo de este trabajo, se puede considerar como ruido a una buena parte de las bajas frecuencias. Por este motivo, resulta interesante definir una variante de la potencia total como la potencia dentro de una determinada banda de frecuencias que se puede considerar como relevantes, potencia relevante. El valor por defecto utilizado como banda relevante es la de 500Hz-5kHz.

En la Figura 3-6 se muestra la potencia total en decibelios para uno de los archivos proporcionados por la Fonoteca Zoológica del Museo Nacional de Ciencias Naturales.

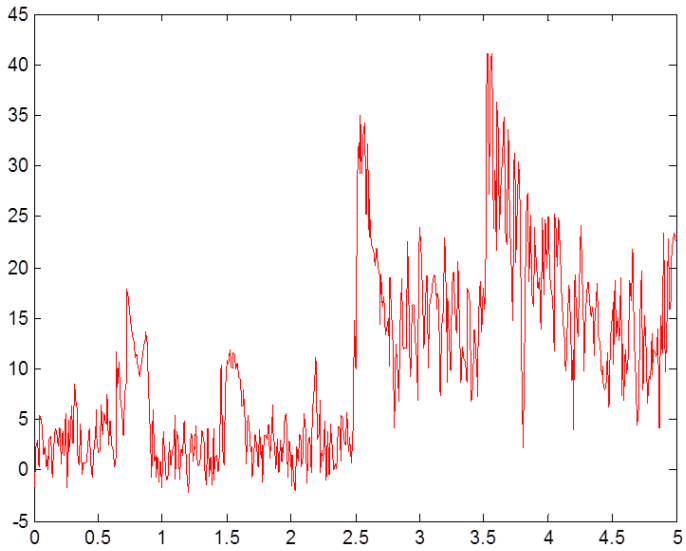


Figura 3-6 Potencia total en dB

3.2.4 Parámetros espectrales

3.2.4.1 Envolvente espectral - AudioSpectrumEnvelopeType

Este parámetro realmente es un vector de parámetros que ofrecen una descripción del espectro completo y no por *frame*. Aunque no formará parte de los parámetros que se usan en la tesis, es importante ya que de él se toman algunos resultados que sirve para el cálculo de otros parámetros.

Es un vector qué describe el espectro de potencia de cada una de las bandas establecidas de acuerdo a una resolución espectral en una distribución logarítmica (base 2) de una señal de audio, por lo que puede ser empleado para generar un espectrograma de dicha señal.

La resolución espectral (r) está definida como el número de bandas de frecuencia por octava, dentro del intervalo entre el borde inferior y superior de las bandas de frecuencia logarítmica (*loEdge* a *hiEdge*)

$$r = 2^j \text{ octavas } -4 \leq j \leq 3 \ , \tag{3-10}$$

existiendo 8 posibles resoluciones: 1/16, 1/8, 1/4, 1/2, 1, 2, 4 y 8 octavas.

El valor predeterminado de *hiEdge* es 16 kHz, que corresponde al límite superior de la audición humana. El valor por defecto de *loEdge* es 62,5 Hz para que el rango predeterminado [*loEdge*,*hiEdge*] corresponda a un intervalo de 8 octavas, logarítmicamente con centro en un frecuencia de 1 kHz. El número de bandas existentes en este intervalo viene dado por

$$B_{in} = \frac{8}{r} \ . \tag{3-11}$$

Los límites de frecuencia inferior y superior para cada banda son

$$loF_b = loEdge \cdot 2^{(b-1)r} \ , \qquad 1 \leq b \leq B_{in} \tag{3-12}$$

$$hiF_b = loEdge \cdot 2^{br} \ .$$

Para disponer de suficiente resolución espectral la norma recomienda una longitud de *frame* tres veces la longitud del *hopsize*. Si se usa el *hopsize* recomendado de 10 milisegundos la longitud de *frame* recomendado es de 30 milisegundos.

Como se vio en el capítulo 2, la multiplicación del *frame* por una ventana para analizar una porción de la señal, equivale en el dominio de la frecuencia a una convolución de la señal con la ventana. Esta convolución reduce en mayor o menor grado las frecuencias armónicas no deseadas debidas a las discontinuidades de la señal. De esta

forma se obtiene una mejor respuesta en frecuencia. Mediante la aplicación de la transformada discreta de Fourier (DFT) se obtiene

$$S_i(k) = \sum_{n=0}^{N_{FT}-1} s\left(n + lN_{hop}\right) \cdot w(n) \cdot e^{-j\frac{2\pi nk}{N_{FT}}} \quad 0 \le l \le L-1 \ , \quad 0 \le k \le N_{FT}-1 \ , \tag{3-13}$$

donde $s\left(n + lN_{hop}\right)$ corresponde al *frame* de señal i analizado, $w(n)$ es la ventana aplicada (el estándar recomienda el uso de la ventana de Hamming), N_{FT} es el número de coeficientes de la DFT, que para simplificar los cálculos deberá ser la siguiente potencia de 2 respecto el número de muestra de la ventana N_w.

Aplicando el Teorema de Parseval, se obtiene la potencia promedio de cada *frame* i

$$\overline{P_i} = \frac{1}{N_{FT} \cdot E_w} \sum_{k=0}^{N_{FT}-1} |S_i(k)|^2 \ , \tag{3-14}$$

donde E_w es la energía de la ventana, que se calcula como

$$E_w = \sum_{n=0}^{N_w-1} |w(n)|^2 \ . \tag{3-15}$$

El espectro de potencia de cada coeficiente de la transformada de Fourier en cada *frame* es

$$P_i(k) = \frac{1}{N_{FT} \cdot E_w} |S_i(k)|^2 \quad para\ k = 0\ y\ k = \frac{N_{FT}}{2} \ , \tag{3-16}$$

$$P_i(k) = 2\frac{1}{N_{FT} \cdot E_w} |S_i(k)|^2 \quad para\ \ 0 \le k \le \frac{N_{FT}}{2} \ . \tag{3-17}$$

Cada coeficiente representa un cierto rango de frecuencia dado por

$$\Delta F = \frac{F_s}{N_{FT}} \ . \tag{3-18}$$

Para encontrar el coeficiente al que pertenece una frecuencia f se utiliza expresión

$$k = round\left(\frac{f}{\Delta F}\right) \quad 0 \le f \le \frac{F_s}{2} \ . \tag{3-19}$$

Por tanto

$$f(k) = k \cdot \Delta F \quad 0 \le k \le \frac{N_{FT}}{2} \ . \tag{3-20}$$

La suma de los coeficientes de energía en cada banda b de cada *frame*, son los coeficientes del *AudioSpectrumEnvelope* (ASE),

$$ASE(b) = \sum_{k=loK_b}^{hiK_b} P(k) \quad 1 \leq b \leq B_{in} \ . \tag{3-21}$$

Adicionalmente se agregan dos coeficientes más para la suma de los coeficientes de energía de las frecuencias entre 0 Hz y 62.5 Hz y desde los 16 KHz hasta la frecuencia de Nyquist $\left(\frac{f_s}{2}\right)$.

3.2.4.2 Centroide de potencia - AudioSpectrumCentroidType

Describe el centro de gravedad del espectro de potencia. Proporciona donde se concentra la mayor parte de la energía del espectro de potencia referenciado a 1kHz en cada *frame*.

Se calculan los coeficientes del espectro de potencia como se ha explicado para el descriptor *AudioSpectrumEnvelope*. Los coeficientes por debajo de los 62.5Hz se sustituyen por un solo coeficiente con una potencia igual a su suma y una frecuencia nominal de 31.25Hz, para evitar una componente de DC,

$$bound = floor\left(\frac{62.5}{\Delta F}\right) \ . \tag{3-22}$$

Esto da como resultado un nuevo espectro de potencia,

$$P'_i(0) = \sum_{k=0}^{bound} P(k) \quad con\ f'^{(0)} = 31.25 \ , \tag{3-23}$$

$$P'_i(k') = P(k + bound) \quad con\ f'^{(k')} = f(k + bound) \ , \quad 1 \leq k \leq \frac{N_{FT}}{2} - bound \ . \tag{3-24}$$

El *AudioSprectrumCentroid* se calcula como

$$C = \frac{\sum_{k'} \log_2\left(\frac{f'(k')}{1000}\right) \cdot P'_i(k')}{\sum_{k'} P'_i(k')} \ . \tag{3-25}$$

Para poner de manifiesto la interpretación de este parámetro (y de otros subsiguientes basados en el espectro de un *frame*) se toma como ejemplo un fragmento del espectrograma de un archivo de audio proporcionado por la Fonoteca Zoológica del Museo Nacional de Ciencias Naturales que contiene el canto de un sapo partero tal como aparece en la Figura 3-7. En ese espectrograma se han seleccionado dos *frames*: una correspondiente a un sonido del canto del anuro en $t = 0.8$; y otro

correspondiente a un sonido "de fondo" en ausencia de canto en $t = 2$. Los espectros correspondientes pueden verse en la Figura 3-8.

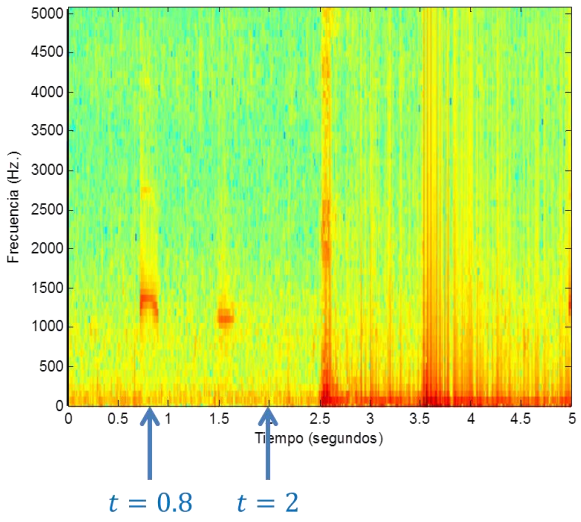


Figura 3-7 Espectrograma de un audio con el canto de un sapo partero

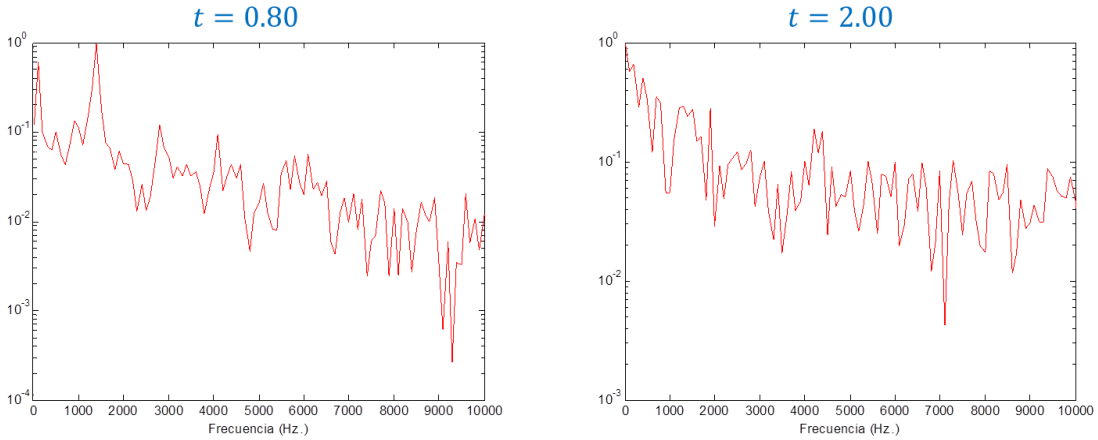


Figura 3-8 Espectros de *frames,* con canto y sin canto de sapo partero

3.2.4.3 Dispersión espectral - audioSprectrumSpreadType

Describe como se encuentra distribuida la energía espectral alrededor de su Centroide. Una valor bajo significa que la mayor parte de la energía del espectro se encuentra concentrado en su Centroide, mientras que un valor alto representa que la distribución de energía se encuentra en un amplio rango de frecuencias. Con este descriptor se puede reconocer si existe un tono puro o ruido.

A partir de los cálculos del *AudioSprectrumCentroid*, el *AudioSprectrumSpread* se calcula como la desviación RMS respecto al centroide en escala de Octava,

$$S = \sqrt{\frac{\sum_{k'} \left(\log_2 \left(\frac{f'(k')}{1000} \right) - C \right)^2 \cdot P'(k')}{\sum_{k'} P'(k')}} \ . \quad \textbf{(3-26)}$$

3.2.4.4 Planitud del espectro - AudioSprectrumFlatnessType

Este descriptor consiste en una serie de valores que expresan la desviación que existe entre el espectro de potencia de la señal y una señal de ruido blanco. Valores alto reflejan ruido en la señal o que no existe un tono particular en la banda. Valores bajo indican una estructura armónica del espectro o la existencia de un tono particular en la banda.

Se realiza el análisis espectral de la señal utilizando el mismo procedimiento especificado para el descriptor AudioSpectrumEnvelope pero sin solapamiento, por lo que el estándar recomienda un *hopSize* de $30\ ms$ para este descriptor.

Los límites de cada banda vienen dado por

$$loEdge = 2^{\frac{n}{4}} \cdot 1KHz \ , \quad \textbf{(3-27)}$$

$$hiEdge = 2^{\frac{B}{4}} \cdot loEdge \ , \quad \textbf{(3-28)}$$

donde n es un número entero.

En vista de las limitaciones en la resolución de frecuencia, no se recomienda el uso de *AudioSpectrumEnvelope* por debajo de 250Hz. Una resolución de frecuencia logarítmica de un cuarto de octava se utiliza para todas las bandas. Por lo tanto, todas las bandas son proporcionales a las bandas de frecuencia empleadas por *AudioSpectrumEnvelope*. Con el fin de reducir la sensibilidad contra las desviaciones en la frecuencia de muestreo, las bandas se definen de una manera solapada: el borde inferior y superiores de cada banda se multiplican por los factores de 0,95 y 1,05, respectivamente. En consecuencia, cada banda se superpone con su banda de vecina un 10%

$$loF_b = 0.95 \cdot loEdge \cdot 2^{\frac{b-1}{4}} \ , \quad \textbf{(3-29)}$$

$$hiF_b = 0.95 \cdot loEdge \cdot 2^{\frac{b}{4}} \quad 1 \leq b \leq B \ . \quad \textbf{(3-30)}$$

Para disminuir el procesamiento, se hace un promedio de cada $2n+1$ de los coeficientes de energía $P(k)$, donde n indica el número de banda a partir de 1kHz, obteniendo un sólo coeficiente de energía $P_g(k')$.

Para cada banda b el coeficiente de *AudioSpectrumFlatnes* (SFM) es

$$SFM_b = \frac{\sqrt[hiK'_b - loK'_b + 1]{\prod_{k'} P_g(k')}}{\frac{1}{hiK'_b - loK'_b + 1}\sum_{k'} P_g(k')} \quad . \tag{3-31}$$

Si no hay señal de audio presente (la potencia media es cero), se devuelve un valor de planitud 1.

3.2.5 Parámetros para reducción de dimensionalidad

La norma MPEG7 contempla dos clases que se usan para la reducción de dimensionalidad. La reducción de dimensionalidad reduce los tiempos que son necesarios en el procesamiento de la información de la misma forma que reduce espacio requerido de almacenamiento sin perder información fundamental. Esta reducción pasa por buscar un cambio de ejes dentro de la distribución de los valores de un parámetro para que estos se puedan identificar con una distancia mínima los ejes. Buscando poder despreciar uno de los dos ejes y quedarse con una de las dos dimensiones como puede verse en la Figura 3-9 como ejemplo de reducción usando el método PCA (*Principal component analysis*). Este proceso no es trivial, ni mucho menos rápido, quedando fuera del alcance de la tesis aunque si se debe tener en cuenta como unos de los trabajos futuros que surgen a partir de este estudio.

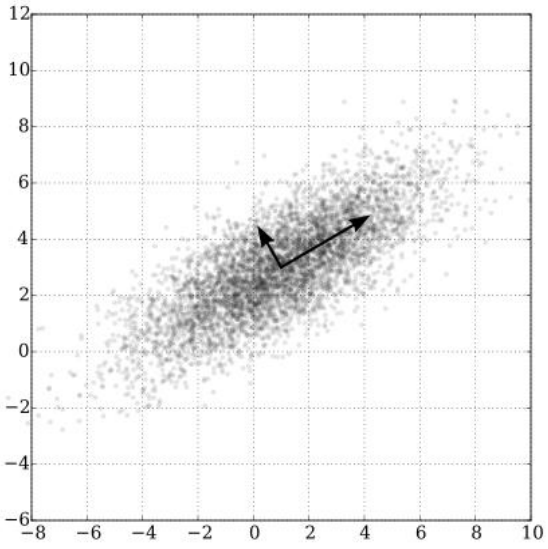


Figura 3-9 Análisis de componentes principales de una distribución normal multivariante (Wikipedia, 2016)

3.2.5.1 Bases del espectro - AudioSprectrumBasisType

Este descriptor se utiliza para proyectar las descripciones de espectro de alta dimensionalidad en una representación de pocas dimensiones. La reducción de dimensionalidad juega un papel importante en las aplicaciones de clasificación automática mediante la representación compacta de información estadística relevante en los segmentos de audio. Estas características han demostrado funcionar bien para aplicaciones de clasificación y recuperación automáticas.

Para realizar una reducción de dimensión sobre una descripción del tipo *AudioSpectrumEnvelope*, que es una representación de *SeriesOfVectors* con M segmentos y N frecuencias, para cada vector, x, se pasa a escala de decibelios

$$\chi = 10 \log_{10}(x_t) \ . \quad \textbf{(3-32)}$$

Estos valores se normalizan usando la norma euclídea (*L2-norm)*

$$r = \sqrt{\sum_{k=1}^{N} \chi_k^2} \ . \quad \textbf{(3-33)}$$

Los vectores normalizados se pueden organizar en una matriz de M x N

$$\tilde{\chi} = \begin{bmatrix} \tilde{x}_1^T \\ \tilde{x}_2^T \\ \vdots \\ \vdots \\ \tilde{x}_M^T \end{bmatrix} , \quad \textbf{(3-34)}$$

donde $\tilde{x}$ es el vector normalizado y se calcula

$$\tilde{x} = \frac{\chi}{r} \ . \quad \textbf{(3-35)}$$

La matriz (3-34) se puede factorizar, usando el método de descomposición en valores singulares (SVD), como

$$\tilde{\chi} = USV^T \ , \quad \textbf{(3-36)}$$

donde U es una matriz fila con bases ortogonales, S es una matriz “diagonal” con los valores singulares y V es la traspuesta de una matriz columna con bases ortogonales. Este método elimina información redundante sin que el efecto por la pérdida de información sea notable.

La reducción espectral se realiza usando sólo las primeras k bases, siendo el rango de valores típicos para clasificación de sonidos de 3 a 10 bases,

$$V_k = [v_1 \quad v_2 \quad \cdots \quad v_k] \ . \quad \textbf{(3-37)}$$

Para calcular la porción de información que se tiene para las k bases utilizadas se usa la matriz de valores singulares

$$I(k) = \frac{\sum_{i=1}^{k} S_{ii}}{\sum_{j=1}^{N} S_{jj}} , \quad \textbf{(3-38)}$$

donde N es el número total de bases que coincide con el número total de frecuencias.

3.2.5.2 Proyección del espectro - AudioSprectrumProjectionType

Este descriptor es el complemento al descriptor *AudioSpectrumBasis* y se utiliza para representar características de baja dimensionalidad de un espectro después de la proyección contra unas bases reducidas. Estos dos descriptores se utilizan siempre juntos. Las características de baja dimensionalidad del *AudioSpectrumProjection* consisten en una serie de vectores, un vector para cada *frame*, t, del espectrograma de entrada normalizado, $\tilde{\chi}$, que produce un vector proyectado, y_t, que se almacena en el *SeriesOfVector*,

$$y_t = \lfloor r_t \quad \tilde{X}_t{}^T V_1 \quad \tilde{X}_t{}^T V_2 \quad \cdots \quad \cdots \quad \tilde{X}_t{}^T V_k \rfloor \ . \tag{3-39}$$

Estos descriptores, *AudioSpectrumBasis* y *AudioSpectrumProjectin* se pueden utilizar para la clasificación automática de sonidos y para realizar un resumen del espectrograma.

3.2.6 Parámetros de armonicidad

3.2.6.1 Tono - AudioFundamentalFrequencyType

Se usa para describir la frecuencia fundamental de la señal de audio que es una buena aproximación al *pitch* y la entonación. La frecuencia fundamental es complementaria al espectro de frecuencia logarítmica, y junto con el descriptor *AudioHarmonicity*, especifica los aspectos detallados de la estructura armónica de sonidos periódicos que el espectro logarítmico no puede representar por falta de resolución. La inclusión de una medida de confianza, utilizando el campo de peso de la serie de escalar es una parte importante del diseño, que permite el manejo adecuado y la escala de las partes de la señal que carecen de una periodicidad clara.

La norma no especifica ningún algoritmo para calcularlo. En este trabajo se ha usado una técnica de LPC (*Linear Prediction Coding*), descrita en el capítulo anterior.

3.2.6.2 Armonicidad – AudioHarmonicityType

Describe el grado de armonicidad de una señal de audio y permite distinguir entre sonidos que tienen un espectro de armónicos (sonidos musicales, voz, etc.) y los que tienen un espectro no armónicas (ruido, sonido sordo, mezclas de instrumentos, etc.). Junto con AudioFundamentalFrequecy, AudioHarmonicity describe la estructura armónica del sonido. Estas características son ortogonales y complementarias a descripciones del tipo AudioSpectrumEnvelope.

El descriptor está compuesto por dos medidas: HarmonicRatio y UpperLimitOfHarmonicity.

3.2.6.2.1 Razón de armonicidad – HarmonicRatio

Se puede definir como la proporción de los componentes armónicos en el espectro de potencia. Se deriva de la correlación entre la señal y una representación retardado de

la señal, un retraso del período fundamental de la señal. Para evitar la dependencia de la frecuencia fundamental real, el algoritmo se realiza una estimación buscando el valor máximo de la correlación cruzada normalizada de la señal.

Primero se calcula la autocorrelación normalizada del *frame* i con un retardo de k

$$\varphi(i,k) = \sum_{j=m}^{m+n-1} s(j)\cdot s(j-k) \Big/ \left(\sum_{j=m}^{m+n-1} s(j)^2 \cdot \sum_{j=m}^{m+n-1} s(k-k)^2 \right)^{\frac{1}{2}}, \qquad \textbf{(3-40)}$$

donde s es la señal de audio; $m = i \cdot n$ siendo i el índice del *frame* con valores entre 0 y $M-1$; M es el número de *frames*; $n = t \cdot sr$ donde t es el tamaño de la ventana utilizada, usando por defecto un valor de $10ms$, y sr es la tasa de muestreo.

Los valores de k van 1 a $K = \omega \cdot sr$, siendo ω el máximo periodo fundamental esperado que por defecto se utiliza el valor de $40ms$.

Se elige el mayor valor de $\varphi(i,k)$ para cada *frame*, como la relación armónica de la misma $H(i)$. Este valor es 1 para señales puramente periódicas y tenderá a 0 para el ruido blanco.

3.2.6.2.2 Frecuencia límite de armonicidad - UpperLimitOfHarmonicity

Se puede definir como la frecuencia a partir de la cual el espectro no puede considerarse armónico. Se calcula a partir de los espectros de potencia originales y filtrados por filtro del tipo peine (*comb*). La señal filtrada se calcula

$$c(j) = s(j) - \lambda s(j-K)\,, \qquad \textbf{(3-41)}$$

con j con valores entre m y $m+n-1$.

Donde λ es la ganancia óptima

$$\lambda = \sum_{j=m}^{m+n-1} s(j)\cdot s(j-K) \Big/ \sum_{j=m}^{m+n-1} s^2(j-K)\,, \qquad \textbf{(3-42)}$$

siendo K el retardo correspondiente a la razón armónica $H(i)$, y la estimación fundamental período. Si K no es entero, $s(j-k)$ se calcula por interpolación lineal.

Una vez se disponen de las dos señales, original y filtrada, se calcula las DFT usando la técnica descrita en *AudioSpectrumEvelope*. Se calcula los espectros de potencia y los coeficientes por debajo de los 62.5Hz se sustituyen por un solo coeficiente con una potencia igual a su suma y una frecuencia nominal de 31.25Hz, para evitar una componente de DC, como se describió para el descriptor *AudioSpectrumCentroid*.

Para cada frecuencia, f_{min}, se calcula la suma de la potencia de frecuencias superiores para ambas señal y se toma su relación

$$\alpha(f_{\lim}) = \sum_{f=f_{\lim}}^{f_{max}} P'(f) \Big/ \sum_{f=f_{\lim}}^{f_{max}} P(f) \ , \qquad \textbf{(3-43)}$$

donde $P(f)$ y $P'(f)$ son las potencias espectrales de la señal original y la filtrada respectivamente, y f_{max} es la máxima frecuencia del DFT.

Partiendo de $f_{\lim} = f_{max}$ y bajando en frecuencia, se encuentra la mayor frecuencia, $f_{u\lim}$ para los que esta relación es menor que un umbral, por defecto 0,5. Esta frecuencia se convierte a escala de octavas centradas en 1kHz,

$$UpperLimitOfHarmoniciy = \log_2(f_{u\lim}/1000) \ . \qquad \textbf{(3-44)}$$

3.2.7 Descriptores de timbre

Los siguientes descriptores tienen como objetivo describir las características perceptivas de los sonidos. El timbre es una característica que permite distinguir dos sonidos que son iguales en tono, volumen y duración subjetiva. Es una característica multidimensional que incluye entre otros la envolvente espectral, envolvente temporal y las variaciones de cada uno de estos como se pone de manifiesto en la Figura 3-10 (Hyoung Gook Kim et al., 2005). Los siete descriptores tímbricos son de dos tipos:

- Descripción temporal del timbre: LogAttacktTime y Temporal Centroid.
- Descripción espectral del timbre: SpectralCentoid, HarmonicSpectralSpread, HarmonicSpectralCentroid, HarmonicSpectralDeviation y HarmonicSpectralVariation.

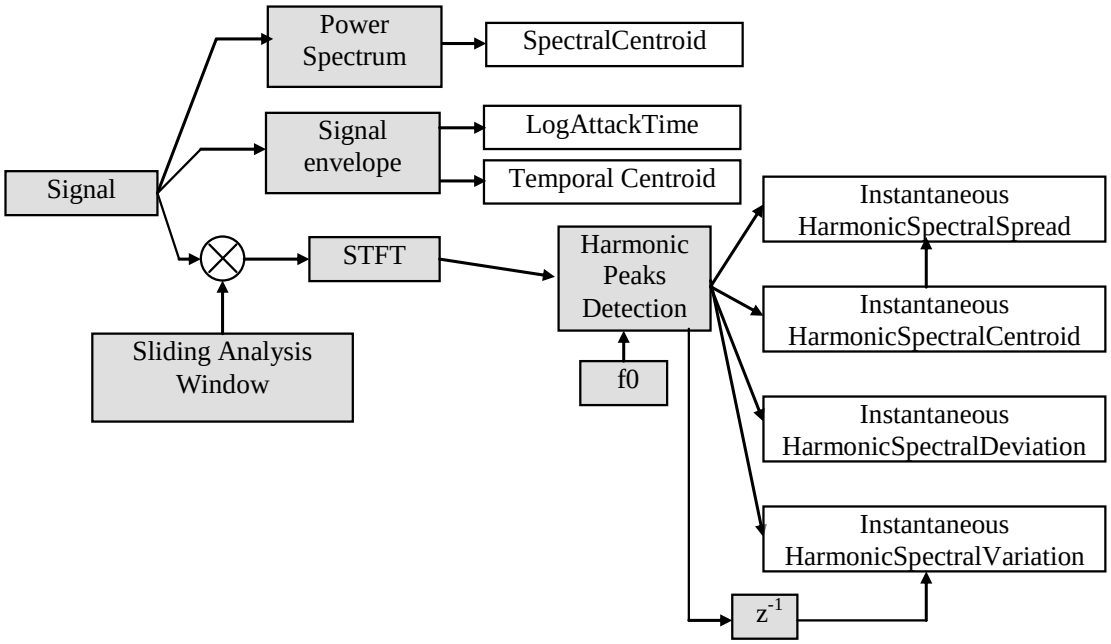


Figura 3-10 Diagrama de bloques para la extracción de los descriptores de timbre (ISO, 2001)

Los descriptores de timbre temporales se extraen de la envolvente de la señal en el dominio del tiempo. La envolvente de la señal describe el cambio de energía de la señal y es generalmente equivalente al ADSR (*Attack*, *Decay*, *Sustain*, *Release*) de un sonido musical. La Figura 3-11 proporciona una representación esquemática de la envolvente de un sonido, mostrando sus diferentes fases y los plazos correspondientes (expresadas en el dominio índice de *frame*). Las fases ADSR de un sonido son:

• Ataque (*Attack*): longitud de tiempo requerido para que el sonido para alcanzar su máximo volumen inicial. Será muy corto para un sonido de percusión.

• Caída (*Decay*): tiempo necesario para el volumen para alcanzar un segundo nivel de volumen conocido como sostenido.

• Sostenido (*Sustain*): nivel de volumen en el que el sonido se mantiene después de la fase de caída. En la mayoría de sonidos es inferior al volumen ataque, pero podría ser el mismo o aún más alto.

• Relajación (*Release*): tiempo que tarda el volumen para reducir a cero.

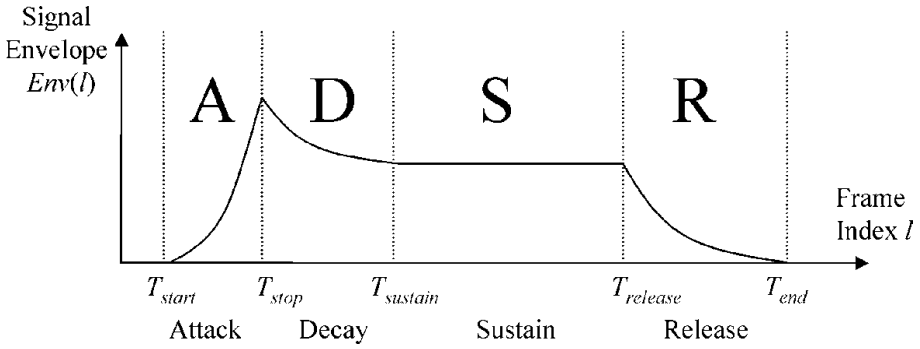


Figura 3-11 Forma general de la envolvente ADSR de un sonido (Hyoung Gook Kim et al., 2005)

El sonido no tiene que tener las cuatro fases. Un órgano tiene una fase de ataque, una fase sostenida y una fase de relajación, pero no la fase de caída.

Las características espectrales de timbre tienen por objeto describir la estructura del espectro armónico y se extraen en un espacio de frecuencia lineal, en lugar de uno logarítmica, derivado de los resultados experimentales en la percepción humana de timbre similares. Se requiere el cálculo de la frecuencia fundamental y los picos armónicos, de acuerdo con una técnica LPC, antes del cálculo de cada una de las características espectrales.

Muchos de los descriptores de timbre se han diseñado para el uso específico en las señales armónicas, como una señal musical monofónica. Cada descriptor describe un segmento de sonido. Un ejemplo de un segmento sonido sería una sola nota tocada en un clarinete.

Los parámetros recomendados para la extracción dependen de si se requieren los valores globales o si también se requieren valores instantáneos. En cualquiera de los dos casos la ventana recomendada por la norma es la de hamming.

Si se requieren sólo los valores globales de los descriptores de Timbre entonces los parámetros de extracción recomendados son de 8 periodos fundamentales para el tamaño de la ventana y 4 periodos fundamentales para el *hopsize*.

Si se requiere serie de valores instantáneos los parámetros de extracción recomendados son de $30ms$ para el tamaño de la ventana y un *hopsize* de $10ms$.

3.2.7.1 Tiempo de ataque - *LogAttackTime*

Se define como el logaritmo en base decimal del tiempo que transcurre entre el comienzo de la señal y el momento en que alcanza su parte estable

$$LogAttackTime = \log_{10}(T_0 - T_0) \ , \qquad \textbf{(3-45)}$$

donde T_0 es el tiempo donde la señal empieza que puede ser estimado cuando la envolvente de la señal sea un 2% del valor máximo; y T_1 corresponde al tiempo donde la señal llega a su fase sostenida o máxima. En la Figura 3-12 se muestra un ejemplo de estos tiempos sobre la envolvente de una señal.

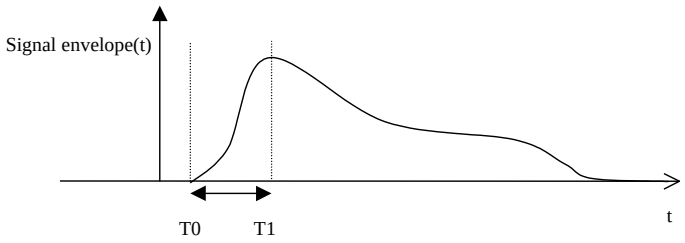


Figura 3-12 Ilustración del tiempo de ataque (ISO, 2001)

Este parámetro está basado en segmentos y no en *frame* por lo que no será utilizado en la tesis.

3.2.7.2 Picos del espectro (Formantes)

Los picos armónicos son los picos del espectro situados en torno a los múltiplos de la frecuencia fundamental. La norma sugiere un algoritmo de detección de estos picos que permite calcular la frecuencia y valor del pico, mediante técnicas LPC.

Por otro lado, los formantes se pueden definir como las bandas de frecuencia donde se concentra la mayor parte de la potencia sonora de un sonido. El concepto de formante amplía y engloba a la de los picos armónicos en un doble sentido:

- Extiende la definición a sonidos poco armónicos, con picos en frecuencias que no son múltiplos de la frecuencia fundamental.
- Permite obtener para cada pico, además de la frecuencia y el valor, su ancho de banda.

Por esta razón, en este trabajo se ha preferido usar el concepto de formante frente al de pico armónico, tomando como parámetros la frecuencia y el ancho de banda de los tres primeros formantes, puesto que de forma general los formantes más significativos son los primeros.

En la Figura 3-13 se recoge el resultado del análisis de formantes utilizando técnicas LPC sobre los dos *frames* utilizados en la Figura 3-8. Sobre cada espectro se ha trazado la estimación obtenida por la técnica LPC, la posición de los formantes y su ancho de banda.

En la norma no se hace referencia expresa a los parámetros elegidos pero sí se utilizan en el cálculo de los siguientes parámetros.

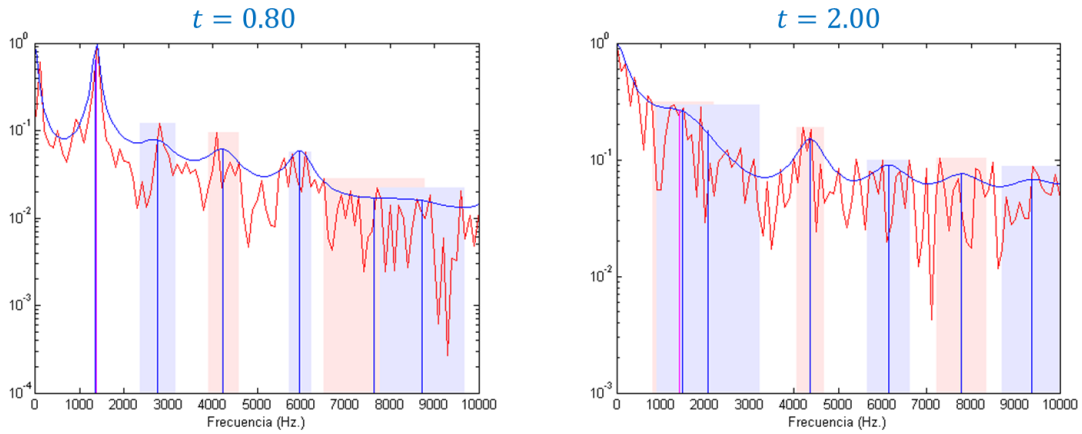


Figura 3-13 Análisis de formantes por técnicas LPC

3.2.7.2.1 Centroide armónico - HarmonicSpectralCentroid

Se podría definir como el centro de gravedad de los picos armónicos o formantes. Se calcula como

$$C_a = \frac{\sum_{i=1}^{n_f} f_i \, v_i}{\sum_{i=1}^{n_f} v_i} \,, \tag{3-46}$$

expresión en la que para el formante i-ésimo, f_i es la frecuencia y v_i el valor de pico.

3.2.7.2.2 Desviación armónica - HarmonicSpectralDeviation

Se podría definir como la desviación de los picos armónicos o formantes con respecto a la envolvente del espectro. Se calcula como

$$D_{ea} = \frac{\sum_{i=1}^{n_f} |\log v_i - \log e_i|}{\sum_{i=1}^{n_f} \log v_i} \,, \tag{3-47}$$

expresión en la que para el formante i-ésimo, v_i es el valor de pico y e_i el valor de la envolvente. Para calcular la este valor de la envolvente la propia norma sugiere hacerlo mediante la media de los 3 formantes adyacentes

$$e_i = \frac{1}{3}\sum_{k=-1}^{+1} v_{i+k} \ . \qquad \textbf{(3-48)}$$

3.2.7.2.3 Dispersión armónica - *HarmonicSpectralSpread*

Se podría definir como la desviación típica de los picos armónicos o formantes con respecto al centroide armónico. Se calcula como

$$D_{ia} = \frac{1}{C_a}\sqrt{\frac{\sum_{i=1}^{n_f} v_i^2 (f_i - C_a)^2}{\sum_{i=1}^{n_f} v_i^2}} \ , \qquad \textbf{(3-49)}$$

expresión en la que para el formante i-ésimo, f_i es la frecuencia y v_i el valor de pico. Donde n_f es el número de *frames* en el segmento de sonido.

3.2.7.2.4 Variación armónica - HarmonicSpectralVariation

Se define como la correlación normalizada entre los valores de los picos armónicos o formantes de dos *frames* adyacentes. Se calcula como

$$V_a = 1 - \frac{\sum_{i=1}^{n_f} v_{i,j} \cdot v_{i,j-1}}{\sqrt{\sum_{i=1}^{n_f} v_{i,j}^2}\sqrt{\sum_{i=1}^{n_f} v_{i,j-1}^2}} \ , \qquad \textbf{(3-50)}$$

expresión en la que $v_{i,j}$ es el valor de pico del formante i-ésimo en el *frame* j-ésimo.

3.2.7.3 Centroide espectral - *SpectralCentroid*

Se define como la media ponderada de los contenedores de frecuencia del espectro de potencia. Se calcula como

$$C_e = \sum_{k=1}^{n} f(k) \cdot S(k) \Big/ \sum_{k=1}^{n} S(k) \ , \qquad \textbf{(3-51)}$$

expresión en la que $S(k)$ es el k-ésimo coeficiente del espectro de potencia, $f(k)$ es la frecuencia de la k-ésimo coeficiente del espectro de potencia y n es el tamaño del espectro de potencia.

Este parámetro no está basado en *frame*, por lo tanto no será utilizado en la tesis.

3.2.7.4 Centroide temporal - *TemporalCentroid*

Se define como el tiempo medio sobre la envolvente de energía. Se calcula como

$$C_t = \sum_{n=1}^{N} \frac{n}{sr} \cdot SEnv\,(n) \Big/ \sum_{n=1}^{N} SEnv(n)\ , \qquad \textbf{(3-52)}$$

expresión en la que $SEnv$ es la envolvente de la señal, sr es la tasa de muestreo y N es la longitud de la envolvente de la señal.

Este parámetro está basado en segmentos y no en *frame*, y por tanto no será utilizado en la tesis.

3.3. Relación entre los parámetros de la tesis y los MPEG7

A modo de resumen, en la Tabla 3-1, se detalla la relación existente entre los parámetros usados en la tesis con los definidos en la norma MPEG-7 los cuales se referencia con el apartado de la norma donde son definidos (ISO, 2001).

	Parámetro		En MPEG-7
1.a	Potencia total		AudioPowerType (5.3.5)
1.b	Potencia relevante		
2	Centroide de potencia		AudioSpectrumCentroidType (5.3.8)
3	Dispersión espectral		AudioSpectrumSpreadType (5.3.9)
4	Planitud		AudioSpectrumFlatnessType (5.3.10)
5	Tono		AudioFundamentalFrequencyType (5.3.12.6)
6	Razón de armonicidad		AudioHarmonicityType (5.3.13)
7	Frecuencia límite de armonicidad		AudioHarmonicityType (5.3.13)
8.a	Frecuencia formante	1	No están descritos directamente aunque si indirectamente descritos en el apartado 5.3.14.3.3 de la norma, como introducción a los parámetros de timbre.
		2	
		3	
8.b	Ancho de banda formante	1	
		2	
		3	
9	Centroide armónico		HarmonicSpectralCentroidType (5.3.16)
10	Desviación armónica		HarmonicSpectralDeviationType (5.3.17)
11	Dispersión armónica		HarmonicSpectralSpreadType (5.3.18)
12	Variación armónica		HarmonicSpectralVariationType (5.3.19)

Tabla 3-1 Relación entre parámetros de la tesis y MPEG-7

Nuevamente, a partir de los *frames* utilizados en la Figura 3-8, en la Tabla 3-2 y la Figura 3-14 se recogen el conjunto de parámetros obtenidos.

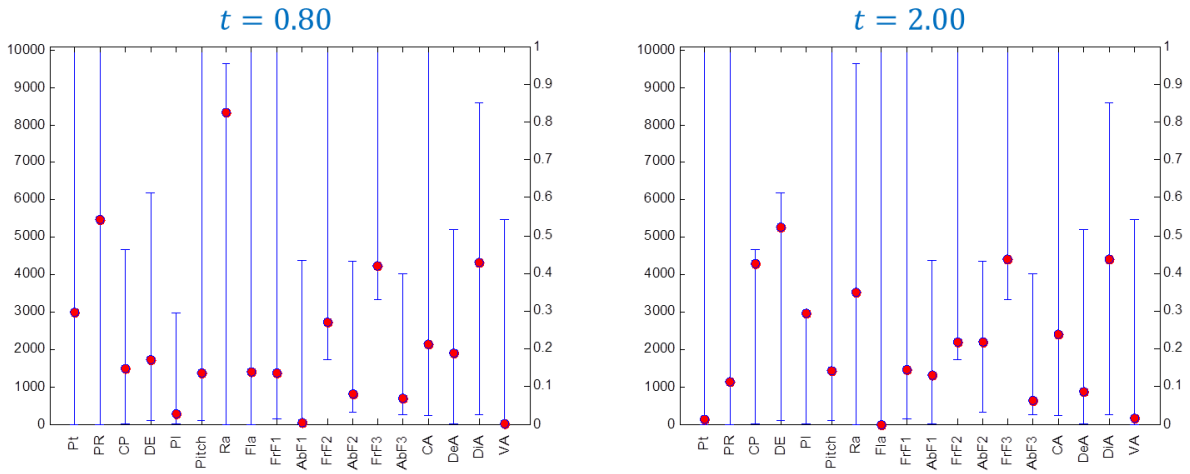


Figura 3-14 Parámetros primeros para ambos *frames*

	Parámetro		$t = 0.8$	$t = 2.0$
1.a	Potencia total		10.8 dB	-1.5 dB
1.b	Potencia relevante		7.1 dB	-7.6 dB
2	Centroide de potencia		1492 Hz	4289 Hz
3	Dispersión espectral		1716 Hz	5258 Hz
4	Planitud		0.0269	0.2942
5	Tono		1378 Hz	1423 Hz
6	Razón de armonicidad		0.83	0.35
7	Frecuencia límite de armonicidad		1403 Hz	0 Hz
8.a	Frecuencia formante	1	1359 Hz	1452 Hz
		2	2740 Hz	2188 Hz
		3	4234 Hz	4417 Hz
8.b	Ancho de banda formante	1	36 Hz	1316 Hz
		2	810 Hz	2191 Hz
		3	682 Hz	647 Hz
9	Centroide armónico		2134 Hz	2405 Hz
10	Desviación armónica		0.187	0.088
11	Dispersión armónica		0.429	0.438
12	Variación armónica		$1.4 \cdot 10^{-5}$	$1.7 \cdot 10^{-2}$

Tabla 3-2 Parámetros primarios obtenidos para ambos frames

3.4. Clasificación de sonidos

Una vez que se han extraído los descriptores de bajo nivel, es posible identificar ciertas características en el dominio del tiempo y de frecuencia, que mediante un método de clasificación posibilite realizar un indexado de audio.

Basándose en las características espectrales es posible una identificación basada en contenido ya que estas características son propias de cada sonido y tienen una variación específica durante el tiempo que puede ser visto como una huella digital.

Utilizando un método de clasificación es posible calcular el nivel de similitud de un sonido o si pertenece a una cierta clase de sonido.

La herramienta de clasificación de sonido estándar MPEG-7 se basa en la proyección de espectro de audio (ASP) como vector de características. Es importante que el vector de característica tenga un tamaño manejable. En la práctica, es necesario reducir el tamaño del vector de características. MPEG-7 emplea la descomposición en valores singulares (SVD) o análisis de componentes independientes (ICA) para este propósito, de donde se obtiene el *Audio Spectrum Basis* (ASB) que consiste en una matriz de dimensión reducida pero que contiene las características principales de ASP.

Un clasificador utiliza el vector de características de dimensión reducida para asignar el sonido a una categoría. Los clasificadores de sonido a menudo se basan en modelos estadísticos. Ejemplos de tales clasificadores incluyen modelos Gaussianos mezcla (MMG), modelos ocultos de Markov (HMMs), redes neuronales (RNAs) y máquinas de vectores soporte (SVMs). El clasificar propuesto por la norma es el de modelos ocultos de Markov, el cual consiste en varios estados. Durante el entrenamiento, los parámetros para cada estado del modelo de audio se estiman mediante el análisis de los vectores de características del conjunto de entrenamiento. Cada estado representa una porción de comportamiento parecido de un proceso de secuencia de símbolos observable. En cada instante de tiempo, el símbolo observable en cada secuencia se queda en el mismo estado o se traslada a otro estado en función de un conjunto de probabilidades de transición de estados. Las diferentes transiciones de estado pueden ser más importantes para el modelado de distintos tipos de datos. Por lo tanto, las topologías de HMM se utilizan para describir cómo se conectan los estados (Hyoung Gook Kim, Moreau, & Sikora, 2004).

La elección del vector de características y la elección del clasificador son fundamentales en el diseño de sistemas de clasificación de sonido.

Los sonidos se modelan de acuerdo con las etiquetas y representadas por un conjunto de parámetros HMM. La clasificación automática de audio usa un conjunto de HMM, etiquetas y funciones bases para encontrar la mejor clase para un sonido de entrada presentándolo a una serie de HMM y seleccionando el modelo con la puntuación máxima verosimilitud.

CAPÍTULO 4. TÉCNICAS DE CLASIFICACIÓN

4.1. Introducción a la clasificación no secuencial

En el capítulo anterior se seleccionaron una serie de atributos que sirven para caracterizar el sonido de cada *frame*. Estas características se usaran como base para distintos métodos de clasificación automática, que es uno de los objetivos de esta tesis como aplicación práctica de la aplicación de técnicas de minería de datos en secuencias temporales. A partir de este capítulo, se evaluaran distintas técnicas de minería de datos y se compararán a fin de concluir cuál es el mejor método en base a los casos estudiados.

En este capítulo se realizará un primer estudio de clasificación de los sonidos a partir de la información de cada *frame* de forma individual, sin tener en consideración los antecesores ni predecesores, por lo que se puede hablar de una clasificación no secuencial.

Los pasos previos a la construcción de un clasificador son el procesamiento de los archivos de audio y la extracción de características a cada *frame*. Como se ha visto en el capítulo anterior, para este trabajo la extracción de características se basa en la norma MPEG-7. De entre todos los parámetros descriptivos de bajo nivel, se han seleccionado sólo aquellos basados en *frames*:

1. Potencia total
2. Potencia relevante
3. Centroide de potencia
4. Dispersión espectral de potencia
5. Planitud del espectro
6. Tono (pitch)
7. Razón de armonicidad
8. Frecuencia límite de armonicidad

9. Picos armónicos o formantes
    a. Frecuencia de los 3 primeros formantes
    b. Ancho de banda de los 3 primeros formantes
10. Centroide armónico
11. Desviación armónica
12. Dispersión armónica
13. Variación armónica

Con ello se consigue una caracterización de cada *frame* mediante un punto en $\mathbb{R}^{18}$. Igualmente, cada archivo de sonido se caracteriza mediante una nube de puntos en $\mathbb{R}^{18}$. La clasificación de cada archivo se consigue mediante la comparación de dicha nube de puntos, con los correspondientes archivos que se seleccionarán como patrones, realizando un "conteo" de la clasificación obtenida de los *frames*.

Esta comparación, denominada en el ámbito de la minería de datos como clasificación supervisada, puede realizarse con numerosas técnicas. Un conjunto amplio y representativo de las mismas es el que se estudiará en este capítulo y está compuesto por los algoritmos siguientes (Gorunescu, 2011):

- Distancia mínima.
- Máxima verosimilitud.
- Árboles de decisión.
- k-vecinos más próximos.
- Máquinas de vectores soporte.
- Regresión logística.
- Redes neuronales.
- Función discriminante.
- Clasificador bayesiano.

Para este estudio se ha contado con archivos de sonidos proporcionados por la Fonoteca Zoológica del Museo Nacional de Ciencias Naturales, correspondientes a 2 especies: el sapo corredor (epidalea calamita) y el sapo partero común (alytes obstetricans). Adicionalmente, de cada especie se recogen varios tipos de canto (vocalización). El resumen se recoge en la Tabla 4-1.

En total se dispone de 6.053 segundos de grabación (1h:40´:53´´). De la simple audición de los sonidos se puede apreciar que, en el sapo corredor, la vocalización en coro no difiere sensiblemente de la vocalización estándar, salvo en el número de individuos que emiten el sonido. Éste es más continuo y quizás más potente pero sus características son similares. Por ello se han agrupado ambas vocalizaciones en un único tipo de sonido.

Especie	Vocalización	Archivos
Sapo corredor	Estándar	20
	Coro	3
	Canto de suelta	10
	Subtotal	**33**
Sapo partero	Estándar	29
	Distress call	1
	Subtotal	**30**
Total		**63**

Tabla 4-1 Tipos de sonidos analizados

Por otra parte, para el sapo partero de la vocalización "distress call" sólo se dispone de una grabación, por lo que no es posible realizar pruebas significativas de clasificación de sonido.

En definitiva, las cinco vocalizaciones anteriores se han reducido a tres tipos de sonido:

1. Sapo corredor con vocalización estándar o de coro.
2. Sapo corredor con vocalización de canto de suelta.
3. Sapo partero con vocalización estándar.

Sonido	Archivos
Sapo corredor	23
Sapo corredor: suelta	10
Sapo partero	30
Total	**63**

Tabla 4-2 Tipos de sonidos analizados por tipo de sonido

Una característica común a todas las grabaciones es que han sido realizadas en el hábitat natural, por lo que están acompañadas de importantes "ruidos" (viento, agua, lluvia, tráfico, voz,…).

Para la fase de entrenamiento de los clasificadores es necesario contar con un conjunto de patrones. Los archivos patrones han sido seleccionados de entre todos los disponibles por su buena calidad relativa. No todos los *frames* dentro de un archivo patrón pueden considerarse representativos del sonido en cuestión. Tan sólo algunos fragmentos, más o menos extensos, tienen ese carácter. Por ello no basta con indicar cuáles son los "archivos patrón" sino, para cada uno de ellos, hay que identificar las regiones de interés (ROI: region of interest).

El proceso de identificación de las ROIs se realiza manualmente mediante la audición de la grabación, la observación detallada del espectrograma y el análisis de los parámetros de *frame*. La Figura 4-1, la Figura 4-2 y la Figura 4-3 muestran los espectrogramas de los patrones con indicación de las respectivas ROIs.

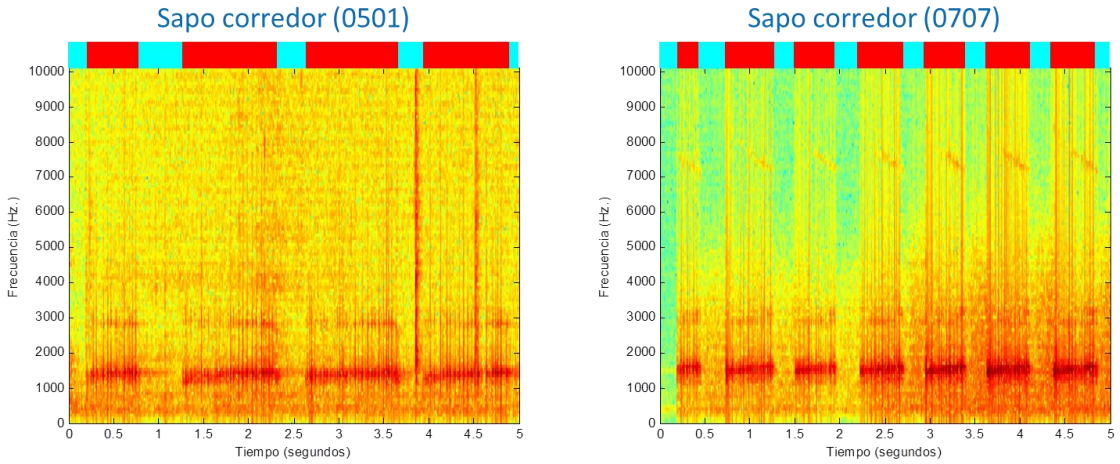


Figura 4-1 ROIs en los patrones del sapo corredor

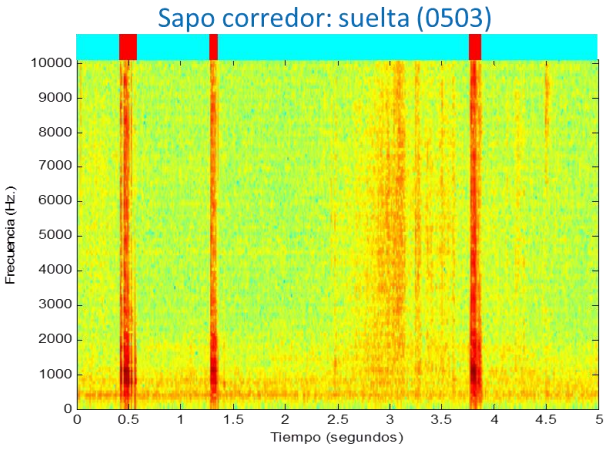


Figura 4-2 ROIs en los patrones del sapo corredor (canto de suelta)

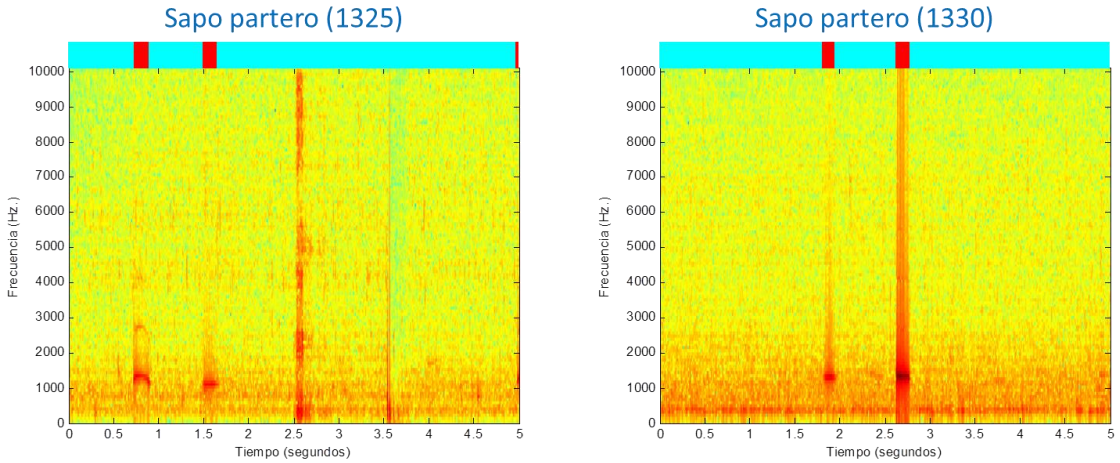


Figura 4-3 ROIs en los patrones del sapo partero

Una vez determinadas las regiones de interés de los archivos patrones, se calculan los valores de los parámetros de *frame* para cada uno de los 4 tipos de regiones posibles: las de interés de los 3 tipos de sonido y las regiones de ruido o ausencia de sonido. El

objetivo será encontrar pautas de comportamiento que ayuden en la clasificación de los sonidos.

Para la mayoría de los algoritmos de clasificación que se utilizarán en este capítulo, bastará con un valor central y una dispersión de la nube de puntos en $\mathbb{R}^{18}$. Sin embargo, para otras técnicas de clasificación es necesario caracterizar estadísticamente los parámetros de *frame* con mayor precisión. Para ello se obtendrá la función de distribución de probabilidad o la función de densidad de probabilidad de los mismos.

El resultado de cada algoritmo estudiado se representará en una gráfica donde el eje horizontal representa los archivos de sonidos ordenados por tipo: sapo corredor (zona azul); sapo corredor en canto de suelta (zona verde); sapo partero (zona roja). Por cada archivo existe una línea vertical cuyo color se corresponde con la clasificación realizada por el algoritmo (con el mismo código de colores anterior). En una clasificación perfecta, el color de cada línea debería corresponder con el color de la zona. Cada discrepancia de color supone un error de clasificación. La altura de cada línea es la probabilidad que el algoritmo asigna a la clasificación realizada. En la Figura 4-4 se muestra como ejemplo el resultado de la clasificación mediante árbol de decisión.

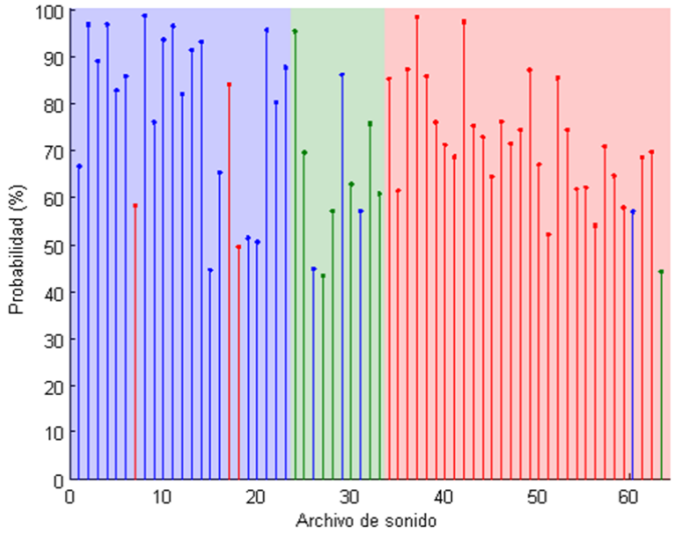


Figura 4-4 Clasificación por árbol de decisión

También se mostrará un resumen del clasificador haciendo uso del porcentaje de acierto de cada especie, así como una estimación del porcentaje de acierto global del clasificador. Siguiendo con el ejemplo del clasificador por árbol de decisión en la Figura 4-5 se muestra como se representará el perfil de éxito.

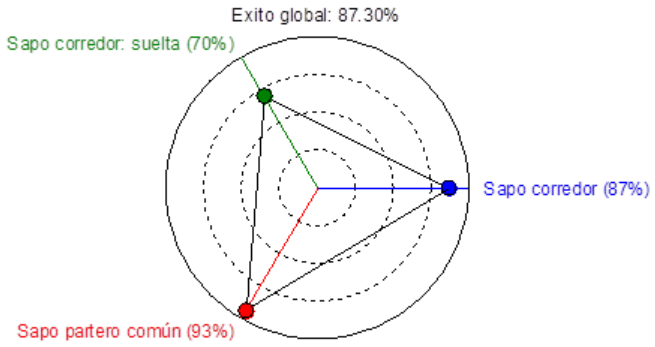


Figura 4-5 Resumen de la clasificación por árbol de decisión

Por último, aunque no menos importante, se representará la matriz de confusión donde se compara el resultado obtenido por el clasificador con la el valor real, conocido al utilizar técnicas de aprendizaje supervisado. En la matriz de confusión se puede observar que todos los datos clasificados en la diagonal han sido clasificados correctamente. Un clasificador perfecto tendría una matriz de confusión con un 100% en la diagonal y un 0% fuera. Siguiendo el ejemplo del árbol de decisión en la Tabla 4-3 se muestra la matriz de confusión de dos formas: conteo del número de archivos clasificados y de forma porcentual.

		Clase obtenida		
		1	**2**	**3**
Clase real	**1**	20	0	3
	2	3	7	0
	3	1	1	28

		Clase obtenida		
		1	**2**	**3**
Clase real	**1**	89.96%	0.00%	13.04%
	2	30.00%	70.00%	0.00%
	3	3.33%	3.33%	93.33%

Tabla 4-3 Matriz de confusión del árbol de decisión

4.2. Distancia mínima

Una de las técnicas más simples, y sin embargo con una importante efectividad, es la de mínima distancia a los patrones. Existen distintos tipos de distancia, aunque la más utilizada es la distancia euclidiana generalizada para espacios multidimensionales (Wacker & Landgrebe, 1971).

Dados 2 puntos i y j en $\mathbb{R}^p$ (un espacio con p parámetros), cuyas coordenadas respectivas son

$$x_i = (x_{i1}, x_{i2}, \dots, x_{ip}) \quad x_j = (x_{j1}, x_{j2}, \dots, x_{jp}) \ , \tag{4-1}$$

se denomina distancia euclídea entre ellos a

$$d_{ij} = \sqrt{\left(x_{i1} - x_{j1}\right)^2 + \left(x_{i2} - x_{j2}\right)^2 + \cdots + \left(x_{ip} - x_{jp}\right)^2} \ , \qquad \textbf{(4-2)}$$

$$d_{ij} = \sqrt{\sum_{k=1}^{p} \left(x_{ik} - x_{jk}\right)^2} \ . \qquad \textbf{(4-3)}$$

No todas estas coordenadas están expresadas en las mismas unidades: algunas lo son en watios (p.e. la potencia), otras en hercios (p.e. el centroide de potencia) y otras son adimensionales (p.e. la razón armónica). Como no todos los valores están expresados en las mismas unidades no es posible realizar la suma que aparece en la expresión anterior. Se requiere una homogeneización o normalización previa de las coordenadas. Para ello se suele utilizar algún indicador de la dispersión de los correspondientes valores o, lo que es lo mismo, del parámetro correspondiente. En este trabajo se utilizará como medida de dispersión el rango de cada parámetro definido como

$$R_k = \max_i x_{ik} - \min_i x_{ik} \ , \qquad \textbf{(4-4)}$$

que tiene la propiedad de proporcionar valores de cada sumando entre 0 y 1. Con esta modificación, la medida de la distancia euclídea queda

$$d_{ij} = \sqrt{\sum_{k=1}^{p} \left(\frac{x_{ik} - x_{jk}}{R_k}\right)^2} \ . \qquad \textbf{(4-5)}$$

Si el punto i corresponde a un *frame* y el punto j es el del valor central de un patrón calculado por la mediana, entonces d_{ij} representa la distancia del *frame* i al patrón j.

En la Figura 4-6 se compara el resultado obtenido aplicando el cálculo de la distancia mínima, a todos los *frames* de cada uno de los archivos, con el valor real que se debería haber obtenido. Puede observarse que las grabaciones de sapo corredor son generalmente bien clasificadas con un porcentaje de acierto del 87% para la vocalización estándar y del 90% para la vocalización de canto de suelta. Sin embargo, el clasificador obtiene un resultado mediocre para las grabaciones de sapo partero con un índice de acierto del 60%. En la Figura 4-7 se presenta el resumen de las prestaciones obtenidas para cada clase y en la Tabla 4-4 la matriz de confusión del clasificador.

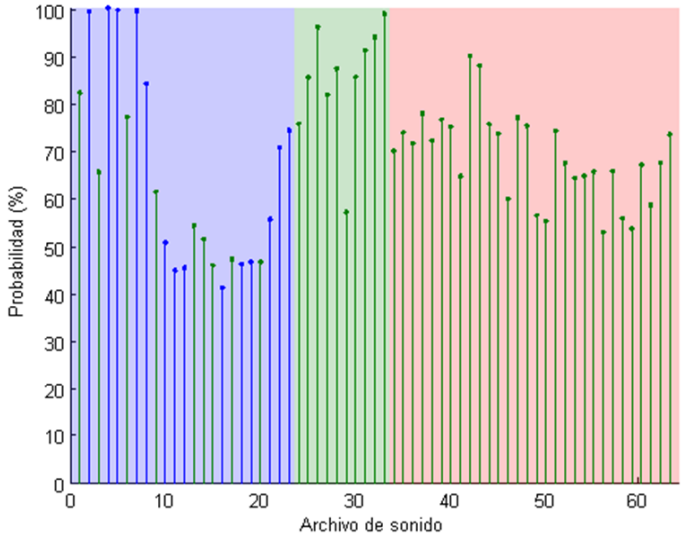


Figura 4-6 Clasificación de todas las grabaciones usando la mínima distancia

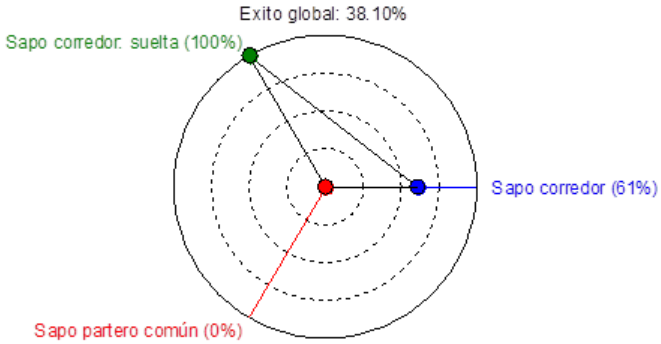


Figura 4-7 Prestaciones del clasificador por mínima distancia

		Clase obtenida		
		1	2	3
Clase real	1	20	0	3
	2	3	7	0
	3	1	1	28

		Clase obtenida		
		1	2	3
Clase real	1	89.96%	0.00%	13.04%
	2	30.00%	70.00%	0.00%
	3	3.33%	3.33%	93.33%

Tabla 4-4 Matriz de confusión del clasificador de mínima distancia

4.3. Máxima verosimilitud

El método de clasificación de mínima distancia analizado en el apartado anterior resume la información sobre los patrones en único indicador por cada parámetro: el valor central del mismo. El valor de la distancia a un valor central de cada clase no aprovecha otra amplia información disponible en la nube de puntos. Es casi intuitivo que una determinada distancia del valor central de un parámetro no tiene la misma importancia si la dispersión del mismo es alta que si es baja.

Una forma de tener en cuenta esta dispersión es considerar no sólo el valor central sino también el rectángulo que lo circunscribe. El rectángulo se convierte en un hiperparalelepípedo (paralelepípedo de n dimensiones) en $\mathbb{R}^n$.

Un paso más en la dirección de aprovechar la información de la nube de puntos es considerar, no sólo el valor central y la dispersión, sino la distribución estadística de los puntos. La estadística de un parámetro x_k puede describirse, bien mediante la función de distribución de probabilidad $F(x_k)$, bien mediante la función de densidad de probabilidad $f(x_k)$. La relación entre estas funciones viene dada por

$$f(x_k) = \frac{dF(x_k)}{dx_k} \ . \tag{4-6}$$

Si se denomina θ a una variable discreta que identifica el tipo de *frame*, pudiendo tomar los valores 1, 2, 3, o 4. Se puede decir que para los sonidos (o *frames*) de tipo θ, el parámetro x_k tiene una función de densidad de probabilidad que denominaremos $f_\theta(x_k)$.

Si inicialmente se considera un único parámetro significativo, el problema de la clasificación es el siguiente: para un *frame* i, cuyo parámetro significativo toma el valor x_i, determinar el valor de θ al que más se asemeja. Para el sonido o patrón j, identificado por $\theta = \theta_j$, la probabilidad de obtener un valor del parámetro x_i es $P(x_i|\theta_j)$. A esa probabilidad condicionada se la denomina verosimilitud (*likelihood*) de que el sonido sea θ_j dado que el valor del parámetro significativo es x_i, y se expresa como

$$\mathcal{L}(\theta_j|x_i) \equiv P(x_i|\theta_j) \ . \tag{4-7}$$

El criterio de clasificación de máxima verosimilitud consiste en obtener el valor de la verosimilitud para todos los valores de θ y clasificar el *frame* con el valor de θ cuya verosimilitud sea máxima (Le Cam, 1979). En el caso de funciones continuas, la verosimilitud se define a partir de la función de densidad de probabilidad de acuerdo a la expresión siguiente

$$\mathcal{L}(\theta_j|x_i) \equiv f_{\theta_j}(x_i) \tag{4-8}$$

o, genéricamente para cualquier *frame* y cualquier patrón de sonido,

$$\mathcal{L}(\theta|x) \equiv f_\theta(x) \ . \tag{4-9}$$

Si se utilizan 2 parámetros significativos x_1 y x_2, la función de densidad de probabilidad es una función bivariante $f_\theta(x_1, x_2)$.

De forma general, utilizando p parámetros, la función de densidad de probabilidad es una función multivariante $f_\theta(x_1, x_2, \dots, x_p)$. Si se denomina vector $\boldsymbol{x}$ a $(x_1, x_2, \dots, x_p)$, se puede escribir la función multivariante como $f_\theta(\boldsymbol{x})$ y la verosimilitud como

$$\mathcal{L}(\theta|\boldsymbol{x}) \equiv f_\theta(\boldsymbol{x}) \ . \tag{4-10}$$

Adicionalmente, considerando que las funciones de densidad de probabilidad gaussianas tienen forma exponencial, es frecuente considerar el logaritmo de la función de verosimilitud de la forma

$$V(\theta|\boldsymbol{x}) \equiv \log \mathcal{L}(\theta|\boldsymbol{x}) = \log f_\theta(\boldsymbol{x}) \ . \tag{4-11}$$

Por las propiedades del logaritmo, cuando la función de verosimilitud $\mathcal{L}(\theta|\boldsymbol{x})$ alcanza el máximo, también lo hace la función de verosimilitud logarítmica $V(\theta|\boldsymbol{x})$.

Se llama verosimilitud diferencial de que un *frame* i sea de una especie j a la diferencia entre la verosimilitud de ser de esa especie j y la verosimilitud de ser un ruido,

$$D_{ij} \equiv V(\theta_j|\boldsymbol{x_i}) - V(\theta_r|\boldsymbol{x_i}) = \log f_{\theta_j}(\boldsymbol{x_i}) - \log f_{\theta_r}(\boldsymbol{x_i}) \ . \tag{4-12}$$

Naturalmente, la máxima verosimilitud de un *frame* se corresponde con la máxima verosimilitud diferencial, y la verosimilitud diferencial de ser un ruido es siempre cero,

$$D_{ir} \equiv V(\theta_r|\boldsymbol{x_i}) - V(\theta_r|\boldsymbol{x_i}) = 0 \ . \tag{4-13}$$

En la Figura 4-8 se compara el resultado obtenido aplicando el cálculo de la máxima verosimilitud, a cada uno de los archivos, con el valor real que se debería haber obtenido.

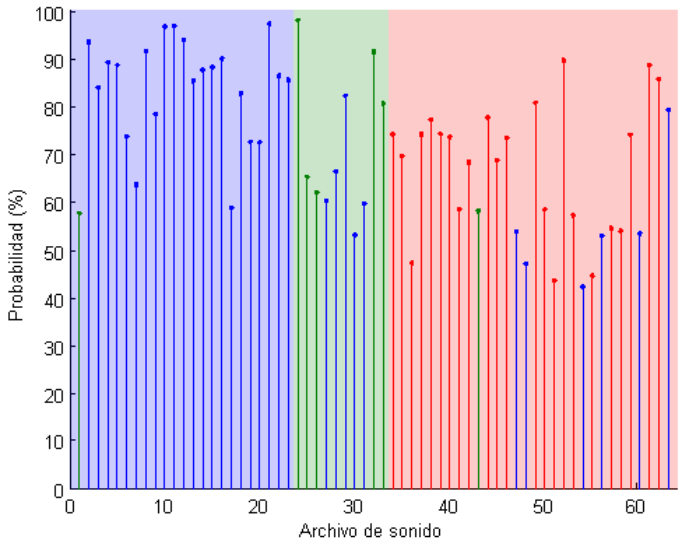


Figura 4-8 Clasificación de todas las grabaciones usando la máxima verosimilitud

Puede observarse que las grabaciones de sapo corredor son generalmente bien clasificadas con un 96% para la vocalización estándar. Sin embargo, ofrece resultados mediocres para las grabaciones de sapo corredor para la vocalización de canto de suelta y sapo partero con índices de acierto del 50% y 77%, respectivamente. La tasa de éxito global es del 79.37%. En la Figura 4-9 se presenta el resumen de las prestaciones obtenidas para cada clase y en la Tabla 4-5 se muestra la matriz de confusión del clasificador.

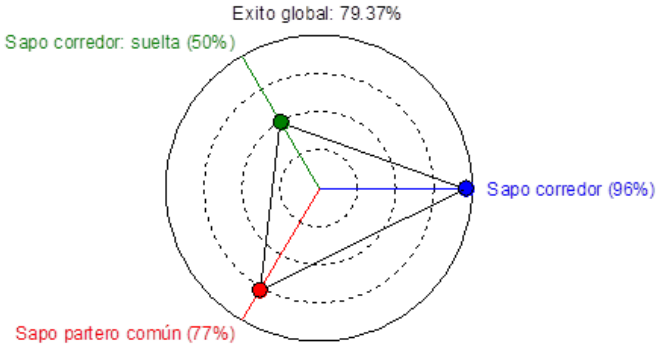


Figura 4-9 Prestaciones del clasificador por máxima verosimilitud

		Clase obtenida		
		1	2	3
Clase real	1	22	1	0
	2	5	5	0
	3	6	1	23

		Clase obtenida		
		1	2	3
Clase real	1	95.65%	4.35%	0.00%
	2	50.00%	50.00%	0.00%
	3	20.00%	3.33%	76.67%

Tabla 4-5 Matriz de confusión del clasificador por máxima verosimilitud

4.4. Árbol de decisión

El árbol de decisión quizás sea uno de los más utilizados en el ámbito del aprendizaje automático debido principalmente a la sencillez del modelo, la rapidez a la hora de clasificar nuevos patrones, la explicación que aporta a la clasificación y la posibilidad de ser representado gráficamente. La construcción se realiza gracias a un proceso de inducción que puede llevarse a cabo mediante distintas implementaciones.

Todo árbol de decisión comienza con un nodo raíz, al que pertenecen todos los casos de la muestra que se quieren clasificar. A partir de este nodo se expande el árbol mediante nodos intermedios y nodos terminales, hojas. Cada nodo hoja se hace corresponder con una categoría concreta, representando las diferentes particiones en las que se ha dividido el espacio de clasificación (Quinlan, 1986).

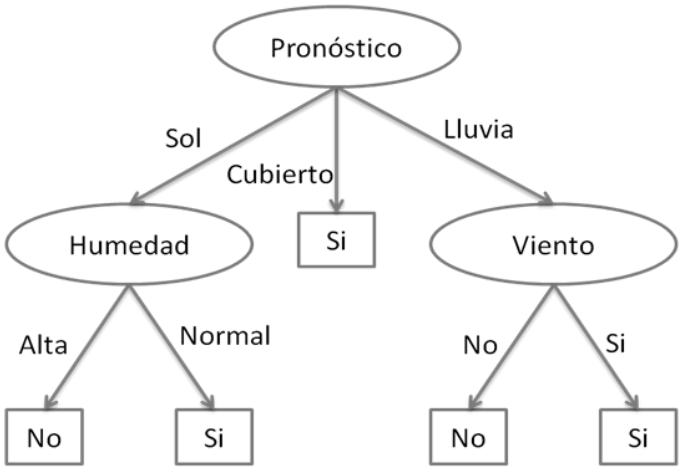


Figura 4-10 Ejemplo de un árbol de decisión sencillo

Como se ha mencionado anteriormente, uno de los motivos del amplio uso de este método es por el hecho de aportar una explicación a la clasificación. Observando la Figura 4-10, donde se ha realizado un árbol de decisión para predecir si jugar o no al tenis en función de las condiciones climáticas, se puede ver que se jugará al tenis cuando el pronóstico sea de cielos cubiertos, o cuando siendo el pronóstico de día soleado la humedad sea normal, o cuando teniendo un pronóstico de lluvia no haya viento. Este aspecto descriptivo es muy importante en muchos ámbitos como pueden ser: diagnósticos médicos, detección de fraudes, mantenimiento preventivo,… ya que proporciona información añadida que puede ser usada por los profesionales en el desarrollo de funciones.

El primer problema al que hay que hacer frente para construir un árbol de decisión es decidir la variable y su división del nodo en cada momento. La división dependerá de la naturaleza de los valores de la variable. Si la variable es discreta se tendrá que evaluar si desarrollar una rama por cada categoría o agrupar categorías. Si la variable es continua se tendrá que evaluar el número de divisiones y las fronteras de estas divisiones. En este sentido, una primera clasificación de los árboles de decisión puede ser: binaria o n-aria. Los árboles binarios sólo permiten dividir un nodo en dos ramificaciones, mientras que los árboles n-aria pueden dividir sus nodos hasta en n ramas.

Para seleccionar la variable del nodo se ha de buscar que la división consiga submuestras lo más homogénea posible respecto a la variable. Para este fin, se necesita una medida de la heterogeneidad o impureza (*impurity*) que sirva para comparar el resultado de las distintas variables y así seleccionar el nodo (Gorunescu, 2011). Al seleccionar un nodo se utiliza la función de ganancia que compara la impureza del nodo padre con la suma de las impurezas de los nodos hijos en los que se divide

$$G(A) = I(A) - \sum_{i=1}^{v} I(A_i)\, p_i \ , \qquad \textbf{(4-14)}$$

donde $G(A)$ es la ganancia en información consecuencia de dividir el nodo padre por la variable predictora A; $I(A)$ y $I(A_i)$ son las impurezas del nodo padre y de los nodos hijos resultantes de la división, respectivamente; A_i los distintos nodos hijos; y p_i es la proporción de casos que se distribuyen en cada uno de los nodos hijos, siendo $\sum p_i = 1$.

Existen diversas formas de calcular la impureza entre las que destacan las basadas en la entropía, el error de clasificación y el índice GINI (expresión (4-15)), que será la función que se usará para calcular la impureza en el algoritmo de clasificación.

$$GINI = \sum_{j=1}^{M} p_{ij}\,(1 - p_{ij}) \ . \qquad \textbf{(4-15)}$$

La variable a seleccionar para el nodo será la que tenga mayor ganancia de información, usando la expresión (4-14). Para cada rama hija se vuelve a realizar el mismo análisis de ganancia. Si uno de los hijos tuviera un resultado homogéneo será un nodo hoja. Existen otros métodos alternativos a la función de ganancia para seleccionar la variable de cada nodo, como por ejemplo, el algoritmo CHAIS donde se cuantifica la correlación existente ente la variable a elegir y la variable dependiente para una división determinada. Este método sólo es aplicable a variables discretas, por lo que si existen variables continuas requieren un paso previo de discretización (Rokach & Maimon, 2008).

Por la forma de construcción, los árboles de decisión se caracterizan por ser muy inestables desde el punto de vista estructural, por lo que también desde un punto de vista discriminante (R.-H. Li & Belford, 2002). El motivo de esta inestabilidad estructural se debe a la visión local en la selección de los nodos, sólo teniendo en cuenta el análisis de la heterogeneidad, por lo que cualquier pequeña variación en el conjunto de entrenamiento puede generar una estructura de árbol diferente.

Además, desarrollar el árbol de decisión hasta que todos los nodos hojas sean homogéneos origina un problema al haberse ajustado demasiado al conjunto de entrenamiento, pudiendo no clasificar correctamente los nuevos casos. Este problema se conoce como sobreentrenamiento u *overfitting*. Para intentar solucionar este problema se aplican distintas técnicas denominadas podas. La poda puede hacerse de dos maneras: según se va construyendo el árbol (pre-poda) y desarrollando el árbol hasta que todos los nodos hojas sean homogéneos y posteriormente eliminar ramas siguiendo un criterio determinado (post-poda). Como se ha mencionado incialmente, un árbol de decisión se puede realizar usando distintas implementaciones. Un ejemplo de pre-poda puede ser la implementación mediante algoritmo CHAID y de post-poda

las implementaciones CART y C4.5. Uno de los criterios más usados en la post-poda se basa en la estimación del error.

Para el estudio particular de este trabajo, se ha aplicado un algoritmo de construcción del árbol de decisión usado el indicie de GINI como criterio para la selección de variable a dividir en cada nodo, e imponiendo un número mínimo de diez elemento en el nodo para poderlo dividir. Lanzado sobre el conjunto de muestras disponibles arroja una tasa de éxito global del 87.30%.

En la Figura 4-11 se compara el resultado obtenido aplicando el modelo obtenido, en MATLAB mediante la función *fitctree*, a cada uno de los archivos con el valor real que se debería haber obtenido. Puede observarse que las tres clases a clasificar tienen buenos resultados, destacando la clasificación del sapo partero.

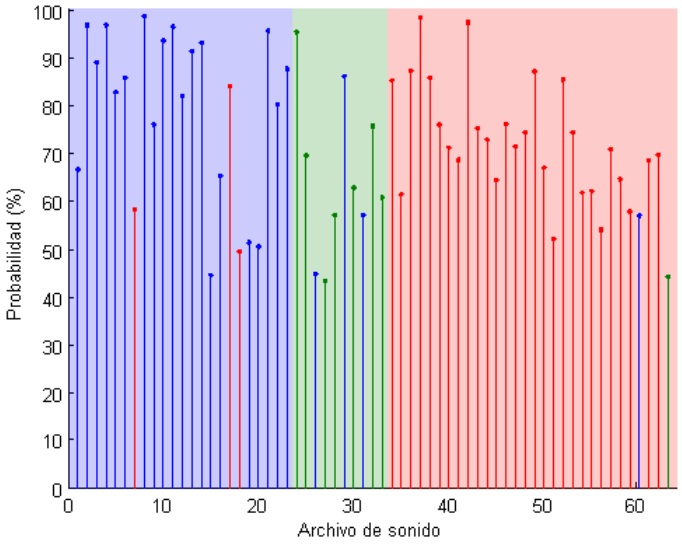


Figura 4-11 Clasificación de todas las grabaciones usando árbol de decisión

En la Figura 4-12 se presenta el resumen de las prestaciones obtenidas para cada clase y en la Tabla 4-6 se muestra la matriz de confusión del clasificador.

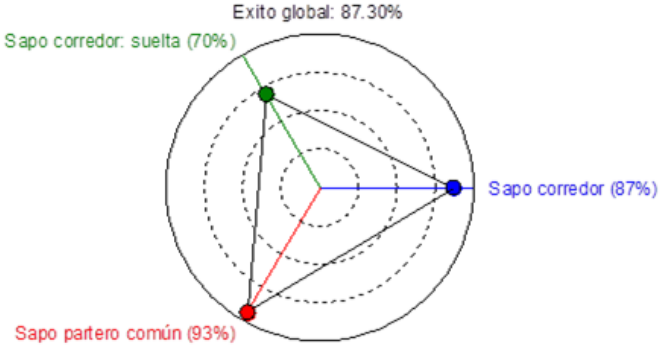


Figura 4-12 Prestaciones del clasificador árbol de decisión

		Clase obtenida		
		1	2	3
Clase real	1	20	0	3
	2	3	7	0
	3	1	1	28

		Clase obtenida		
		1	2	3
Clase real	1	89.96%	0.00%	13.04%
	2	30.00%	70.00%	0.00%
	3	3.33%	3.33%	93.33%

Tabla 4-6 Matriz de confusión del árbol de decisión

4.5. k-vecinos más próximos

Bajo la perspectiva de vecindad es necesaria la definición de una cierta medida de distancia entre los distintos elementos y usar una métrica para la comparación de los distintos objetos. La diferencia más destacable que presentan las técnicas basadas en criterios de vecindad con respecto a otros métodos de clasificación, es su simplicidad conceptual: la clasificación de un nuevo caso se calcula en función de las clases conocidas más próximas.

Si se asume que los casos (x_i, θ_i) ya clasificados se hallan independiente e idénticamente clasificados respecto a la distribución (x, θ), se pueden establecer ciertos argumentos heurísticos para el desarrollo de buenos procesos de clasificación. Por ejemplo, parece razonable asumir que observaciones que se encuentran cercanas, en base a una métrica apropiada, tendrá aproximadamente la misma distribución de probabilidad a posteriori en sus respectivas clasificaciones (Cover & Hart, 1967).

Se pueden distinguir los algoritmos de clasificación por vecindad del resto de algoritmos de clasificación supervisada puesto que, a diferencia del resto, no se genera un modelo inductivo previo y posteriormente se usa para clasificar nuevos casos, sino que en los algoritmos basados en vecindad el modelo se encuentra implícito en los datos.

En la regla de los k vecinos más próximos (*k-NN*), la clase asignada a un nuevo caso será la más votada entre los k vecinos más próximos del conjunto de entrenamiento. En la Figura 4-13 se pueden observar gráficamente distintos casos en función del valor de k. No existe un comportamiento monótono en la relación del porcentaje de aciertos respecto al valor de k. En general, los mejores resultados se obtienen con $k = 3$ o $k = 5$, aunque esto depende, en gran medida, de cada problema.

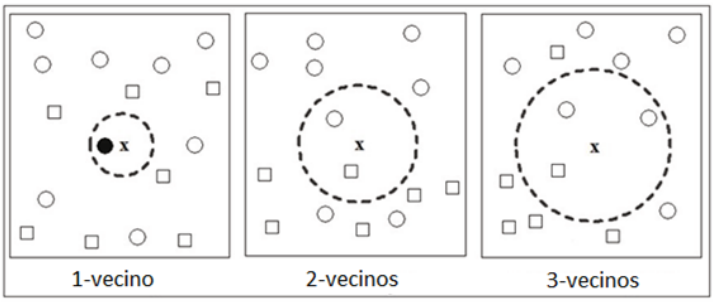


Figura 4-13 Diferentes casos de k-vecinos más próximos (k = 1, 2, 3) (Gorunescu, 2011)

En las primeras versiones del algoritmo se consideraban por igual todas las variables que intervienen en el modelo a la hora de calcular la distancia. Evoluciones posteriores han enriquecido el algoritmo con distintos modificadores útiles en distintas situaciones:

- Introducción de rechazo de muestras que no obtengan una cierta garantía de que la clase asignada sea la correcta.
- Asignación de pesos distintos a las variables en función de su importancia en la determinación de la clase.
- Selección de prototipos con el fin de reducción del conjunto de entrenamiento.

Todas estas técnicas pueden hibridar de muchas maneras, lo que da idea de lo complejo que puede llegar a ser un clasificador por vecindad. No obstante, se puede concluir que el método de los k-vecinos más cercanos es el más simple de todos los algoritmos de aprendizaje automático, ya que simplemente consiste en clasificar un objeto por el voto de la mayoría de sus vecinos (Gorunescu, 2011). Los puntos más negativos de este algoritmo pueden ser el coste computacional elevado y que es muy influenciable por valores extremos y atípicos (*outliers*).

Para el estudio particular de este trabajo se ha aplicado el algoritmo usado una métrica euclidiana y un valor de $k = 1$, que lanzado sobre el conjunto de muestras disponibles arroja una tasa de éxito global del 73.02%.

En la Figura 4-14 se compara el resultado obtenido aplicando el método de los vecinos más próximos, haciendo uso de la función *fitcknn* de MATLAB. Puede observarse que las grabaciones de sapo corredor son generalmente bien clasificadas con un 100% para la vocalización estándar. Sin embargo ofrece resultados mediocres para las grabaciones de sapo corredor para la vocalización de canto de suelta y sapo partero con índices de acierto del 50% y 60%, respectivamente. En la Figura 4-15 se presenta el resumen de las prestaciones obtenidas para cada clase y en la Tabla 4-7 se muestra la matriz de confusión del clasificador.

		Clase obtenida		
		1	**2**	**3**
Clase real	**1**	23	0	0
	2	5	5	0
	3	12	0	18

		Clase obtenida		
		1	**2**	**3**
Clase real	**1**	100.00%	0.00%	0.00%
	2	50.00%	50.00%	0.00%
	3	40.00%	0.00%	60.00%

Tabla 4-7 Matriz de confusión del clasificador k-NN

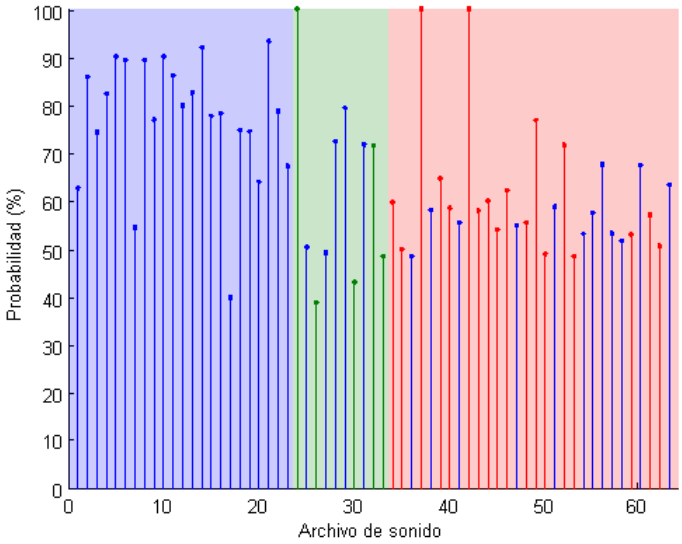


Figura 4-14 Clasificación de todas las grabaciones usando *k-NN*

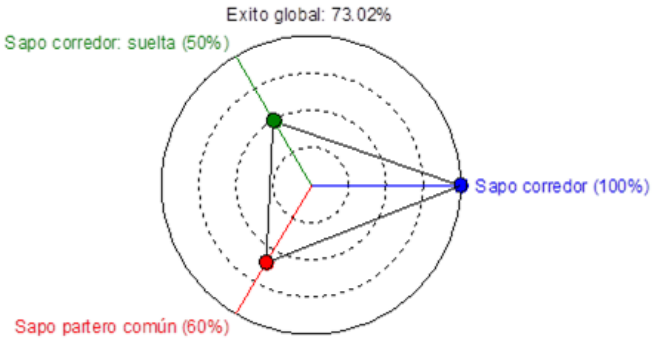


Figura 4-15 Prestaciones del clasificador *k-NN*

4.6. Máquinas de vectores soporte

Las máquinas de vectores soporte, SVM (*Support Vector Machines*) tienen su origen en los trabajos sobre la teoría del aprendizaje estadístico (Boser, Guyon, & Vapnik, 1992; Cortes & Vapnik, 1995). Las SVMs fueron desarrolladas específicamente para la tarea de clasificación binaria, aunque en la actualidad se utilizan para resolver otros tipos de tareas como la regresión y el agrupamiento.

La mayor diferencia entre las SVMs y otros métodos de aprendizaje automático se encuentra en que las SVMs no se centran en construir modelos que cometan pocos errores, sino que, intenta producir predicciones en las que se pueda tener mucha confianza, asumiendo que cometerá errores.

La idea es seleccionar un hiperplano de separación que equidiste de las muestras más cercanas de cada clase, consiguiendo un margen máximo en cada lado del hiperplano. A la hora de definir el hiperplano, sólo se tienen en cuenta las muestras de cada clase

que se encuentran en la frontera. A estas muestras se les denominan vectores soporte. Esta idea es fácilmente aplicable cuando las clases que forman las muestras de entrenamiento están completamente separadas mediante una función lineal, Figura 4-16 (a). En este caso existen infinitos hiperplanos que permiten separar las clases. Para identificar el hiperplano óptimo se define el concepto de margen de un hiperplano de separación, γ, buscando el hiperplano que presente mayor margen (Cristianini & Shawe-Taylor, 2000), Figura 4-17. Para resolver esta búsqueda se utilizan métodos de optimización de funciones.

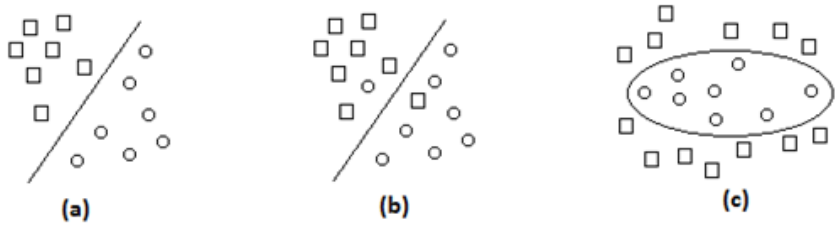


Figura 4-16 (a) Clases perfectamente separables linealmente (b) Clases separables linealmente con error (c) Clases no separables linealmente

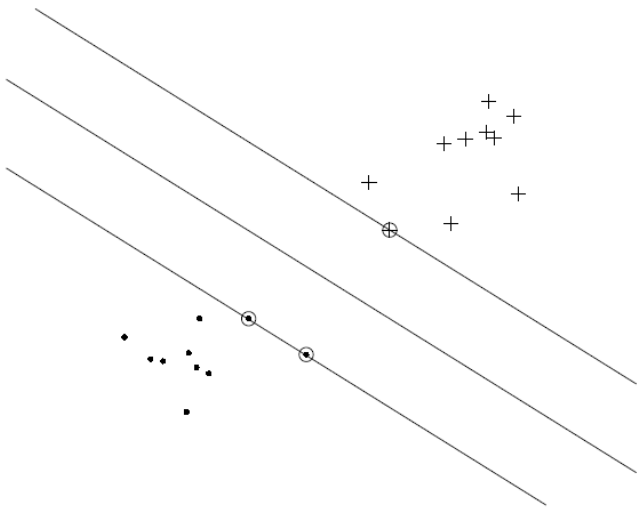


Figura 4-17 Ejemplo de margen óptimo señalando los vectores soporte (Shawe-Taylor & Cristianini, 2004)

En los problemas se caracterizan normalmente por poseer ruido y no pueden ser separados linealmente sin admitir ciertos errores en la clasificación, Figura 4-16 (b). En el problema de optimización para maximizar el margen, admitir errores implica consentir que las restricciones puedan ser violadas. Para contemplar estas situaciones, se introduce en cada restricción una variable de holgura, ξ_i. Cuando las clases no pueden separarse mediante una función lineal, Figura 4-16 (c), se recurre a transformaciones no lineales que llevan del espacio de entrada a otro espacio de mayor dimensionalidad, incluso infinita, en el que las clases sean separables linealmente. A este nuevo espacio se le conoce como espacio de características, Figura 4-18.

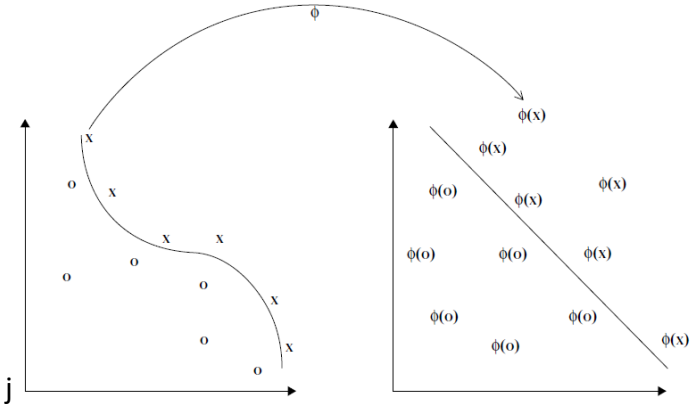


Figura 4-18 Ejemplo de función de transformación del espacio de entrada al especio de las características
(Shawe-Taylor & Cristianini, 2004)

En este nuevo espacio resulta interesante resaltar que siempre se puede clasificar linealmente cualquier conjunto de datos siempre que espacio tenga número de dimensiones apropiado, pudiendo llegar a infinitas (Han, Kamber, & Pei, 2011). En este espacio el hiperplano de separación es $\langle \boldsymbol{w}, \boldsymbol{x} \rangle + b = 0$, donde $\boldsymbol{w}$ es un vector perpendicular al hiperplano. La función ϕ genera vectores en el nuevo espacio a partir de los atributos originales. El margen de las clases a este hiperplano resulta $1/\|w\|$, por lo que el problema se encuentra en minimizar $\|w\| = \langle \phi(x_i), \phi(x_j) \rangle$ (Hastie, Tibshirani, & Friedman, 2005). El producto puede ser expresado como función de los datos de entrada y su expresión dependerá del espacio de entrada, por lo que se simplifica mucho al partir de unas dimensiones más reducidas,

$$K(x_i, x_j) = \langle \phi(x_i), \phi(x_j) \rangle \ . \tag{4-16}$$

A este producto (4-16) se le denomina kernel, siendo el espacio de características un espacio de Hilbert, generalización del espacio euclídeo.

Existen distintos tipos de funciones kernel, las más comunes son:

- Líneal

$$K(x_i, x_j) = x_i^T x_j \tag{4-17}$$

- Gaussiana

$$K(x_i, x_j) = e^{-\frac{\|x_i - x_j\|^2}{2\sigma^2}} \tag{4-18}$$

- Polinómica

$$K(x_i, x_j) = (x_i^T x_j + 1)^p \tag{4-19}$$

De forma genérica son kernels todas aquellas funciones $K(u, v)$ que verifiquen el teorema de Mercer (Riesz & Nagy, 1990)

$$\int_{u,v} K(u,v)\, g(u)\, g(v)\, du\, dv > 0 \ , \qquad \textbf{(4-20)}$$

para toda función g de cuadrado integrable.

Las SVM tiene una naturaleza dicotómica. Sin embargo es habitual necesitar resolver problemas multiclase, como ocurre en el caso de estudio de este trabajo. Existen distintas técnicas para aproximar a SVM multiclase de N clases. Las técnicas más conocidas son:

- **Una frente a todas** que descompone el problema multiclase con N clases en problemas binarios, en los cuales cada una de las clases se enfrenta al resto. Se construyen N clasificadores que definen N hiperplanos que separan la clase i de las N-1 restantes. Para que se considere una clasificación admisible es necesario que el elemento a clasificar sólo sea asociado a una clase de uno de los clasificadores binarios mientras que en el resto de clasificadores binarios el elemento debe pertenecer a la categoría asociada al resto en cada caso.

- **Una frente a una** que descompone el problema de N clases en $N(N-1)/2$ problemas binarios, donde se crean todos los posibles enfrentamientos uno a uno entre clases. Cada elemento a clasificar se somete a todos estos clasificadores binarios, y se añade un voto a la clase ganadora para cada caso, resultando como clase propuesta la que más votos suma.

Para implementar este modelo se hace uso de la función de MATLAB *fitcsvm* usando un kernel gausiano. Como se ha comentado anteriormente, esta función sólo realiza una clasificación binaria, por lo que se ha desarrollado una función propia para implementar la técnica una frente a todas de forma iterativa.

En la Figura 4-19 se compara el resultado obtenido por el modelo creado. Puede observarse que las grabaciones de sapo corredor son generalmente bien clasificadas con un 100% para la vocalización estándar. Sin embargo ofrece resultados mediocres para las grabaciones de sapo corredor para la vocalización de canto de suelta y sapo partero con índices de acierto del 20% y 70%, respectivamente. En la Figura 4-20 se presenta el resumen de las prestaciones obtenidas para cada clase y en la Tabla 4-8 se muestra la matriz de confusión del clasificador.

		Clase obtenida		
		1	**2**	**3**
Clase real	**1**	23	0	0
	2	8	2	0
	3	8	1	21

		Clase obtenida		
		1	**2**	**3**
Clase real	**1**	100.00%	0.00%	0.00%
	2	80.00%	20.00%	0.00%
	3	26.67%	3.33%	70.00%

Tabla 4-8 Matriz de confusión del clasificador SVM

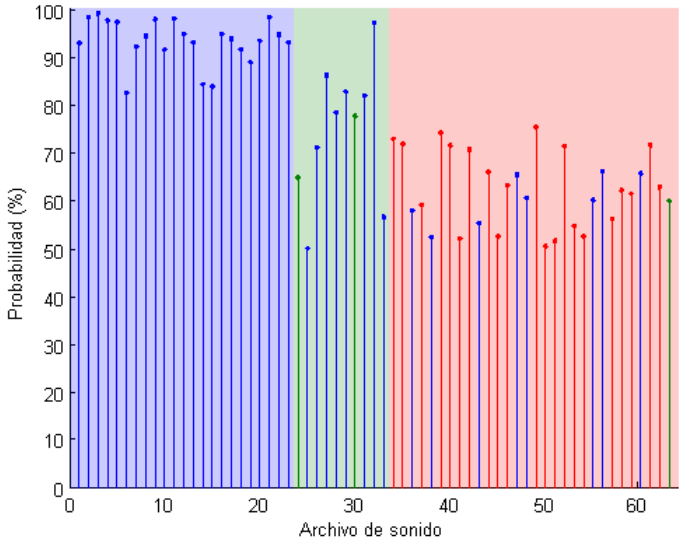


Figura 4-19 Clasificación de todas las grabaciones usando SVM

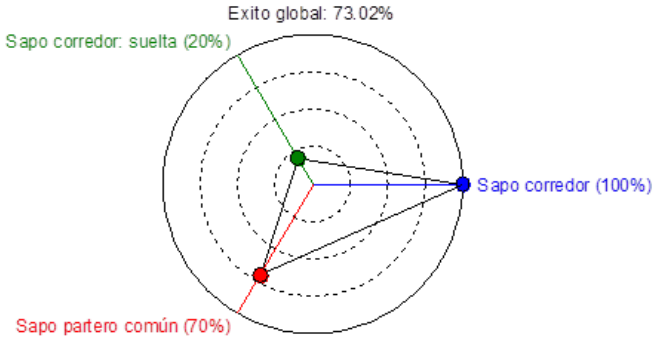


Figura 4-20 Prestaciones del clasificador SVM

4.7. Regresión logística

La regresión logística es un modelo estadístico donde se usa la probabilidad de pertenecer a una clase como función de otras variables predictoras (Hastie et al., 2005). El modelo tiene la forma

$$p_i = \beta_0 + \boldsymbol{\beta}_1' \boldsymbol{x}_i \ , \tag{4-21}$$

donde p_i es la probabilidad de que la muestra i tome el valor de una determinada clase cuando $x = x_i$ (Dobson & Barnett, 2008). El inconveniente principal de esta fórmula (4-21) es que p_i debe estar entre cero y uno, y no hay ninguna garantía de que la predicción $\beta_0 + \boldsymbol{\beta}_1' \boldsymbol{x}_i$ verifique esta restricción. Para garantizar que la respuesta prevista esté entre cero y uno se hace uso de una función que satisfaga esta restricción

$$p_i = F(\beta_0 + \boldsymbol{\beta}_1' \boldsymbol{x}_i) \ . \tag{4-22}$$

Una clase de funciones que satisfacen esta condición son las funciones de distribución, por lo que el problema se puede resolver tomando como F cualquier función de

distribución. Haciendo uso de la función de distribución logística, que tiene la ventaja de ser continua, se obtiene

$$p_i = \frac{1}{1 + e^{-(\beta_0 + \beta_1' x_i)}} \ . \quad \textbf{(4-23)}$$

Particularizando al caso de dos clases se tiene

$$1 - p_i = \frac{e^{-(\beta_0 + \beta_1' x_i)}}{1 + e^{-(\beta_0 + \beta_1' x_i)}} \ . \quad \textbf{(4-24)}$$

Haciendo una transformación se consigue un modelo lineal que se denomina *logit* (Cramer, 2005)

$$g_i = \log \frac{p_i}{1 - p_i} = \beta_0 + \boldsymbol{\beta}_1' \boldsymbol{x}_i \ . \quad \textbf{(4-25)}$$

La variable g representa en escala logarítmica la diferencia entre las probabilidades de pertenecer a ambas clases, y al ser una función lineal se facilita la estimación y la interpretación del modelo.

Generalizando a K clases se obtiene el modelo que se indica en la expresión (4-26).

$$\log \frac{\Pr\,(G = 1|X = x)}{\Pr\,(G = K|X = x)} = \beta_{10} + \boldsymbol{\beta}_1' \boldsymbol{x}_i \ ,$$

$$\log \frac{\Pr\,(G = 2|X = x)}{\Pr\,(G = K|X = x)} = \beta_{20} + \boldsymbol{\beta}_2' \boldsymbol{x}_i \ , \quad \textbf{(4-26)}$$

$$\vdots$$

$$\log \frac{\Pr\,(G = K - 1|X = x)}{\Pr\,(G = K|X = x)} = \beta_{(K-1)0} + \boldsymbol{\beta}_{K-1}' \boldsymbol{x}_i \ .$$

Volviendo a la simplificación para $K = 2$, se tiene que los parámetros del modelo se corresponden con: β_0, la ordenada en el origen y $\boldsymbol{\beta}_1 = (\beta_1, \dots, \beta_n)$. A veces, se utilizan también como parámetros e^{β_0} y e^{β_i}, denominados los *odds ratios* o ratios de probabilidades, que indican cuánto se modifican las probabilidades por unidad de cambio en las variables x,

$$O_i = \frac{p_i}{1 - p_i} = e^{\beta_0} \prod_{j=1}^{n} \left(e^{\beta_i}\right)^{x_j} \ . \quad \textbf{(4-27)}$$

En el caso de tener dos elementos con los mismos valores en todas sus variables menos en una, h, donde $x_{ih} = x_{jh} + 1$ el *odds ratio* para estos dos casos es

$$\frac{O_i}{O_j} = e^{\beta_h} \ . \qquad \textbf{(4-28)}$$

La expresión (4-28) indica cuánto se modifica el ratio de probabilidad cuando la variable x_j aumenta en una unidad.

Si se considera $p_i = 0.5$ se tiene

$$\log \frac{p_i}{1-p_i} = \beta_0 + \beta_1 x_{1i} + \cdots + \beta_n x_{ni} = 0 \ , \qquad \textbf{(4-29)}$$

$$x_{1i} = -\frac{\beta_0}{\beta_1} - \sum_{j=1}^{n} \frac{\beta_j x_{ji}}{\beta_1} \ , \qquad \textbf{(4-30)}$$

donde x_{1i} representa el valor de $\mathrm{x_1}$ que hace igualmente probable que un elemento cuyas restantes variables son $\mathrm{x_{2i}}, \ldots, \mathrm{x_{ni}}$, pertenezca a una de las dos clases.

En la Figura 4-21 se compara el resultado obtenido aplicando el modelo generado en MATLAB mediante la función *mnrfit*. Puede observarse que las grabaciones de sapo corredor son generalmente bien clasificadas con un 100% para la vocalización estándar. Sin embargo ofrece resultados mediocres para las grabaciones de sapo corredor para la vocalización de canto de suelta y sapo partero con índices de acierto del 40% y 47%, respectivamente. En la Figura 4-22 se presenta el resumen de las prestaciones obtenidas para cada clase y en la Tabla 4-9 se muestra la matriz de confusión del clasificador.

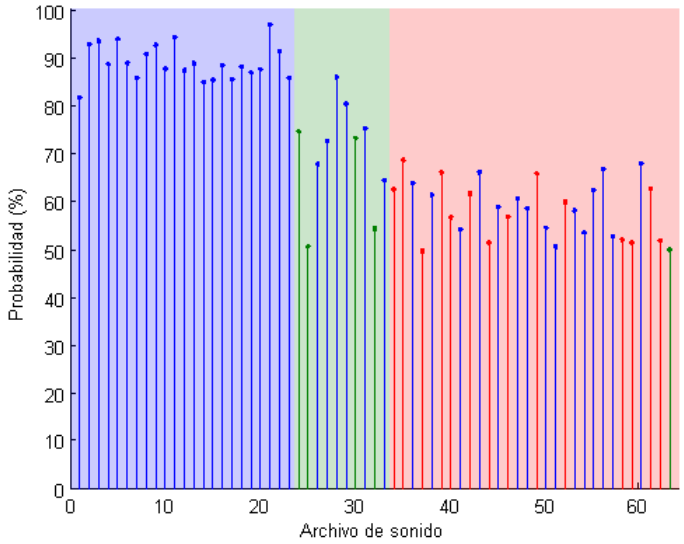


Figura 4-21 Clasificación de todas las grabaciones usando regresión logística

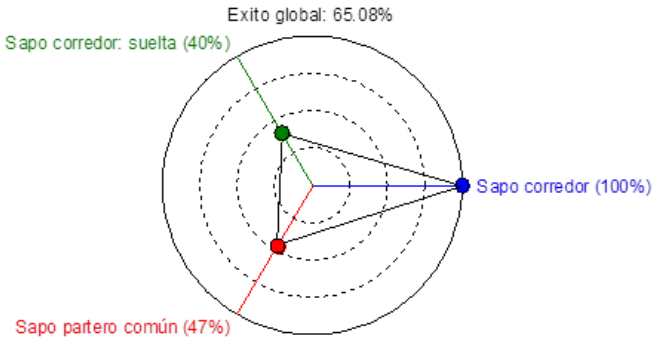


Figura 4-22 Prestaciones del clasificador por regresión logística

		Clase obtenida		
		1	2	3
Clase real	1	23	0	0
	2	6	4	0
	3	15	1	14

		Clase obtenida		
		1	2	3
Clase real	1	100.00%	0.00%	0.00%
	2	60.00%	40.00%	0.00%
	3	50.00%	3.33%	46.67%

Tabla 4-9 Matriz de confusión de la clasificación por regresión logística

4.8. Redes neuronales

Las Redes Neuronales, NN (*Neural Networks*), fueron originalmente una simulación abstracta de los sistemas nerviosos biológicos, constituidos por un conjunto de unidades llamadas neuronas o nodos conectados unos con otros.

El primer modelo de red neuronal fue propuesto en 1943 (McCulloch & Pitts, 1943) en términos de un modelo computacional de actividad nerviosa. Este modelo era un modelo binario, donde cada neurona tenía un escalón o umbral prefijado, y sirvió de base para los modelos posteriores.

Las características principales de las redes neuronales son las siguientes (Du & Swamy, 2013):

- Auto-Organización y Adaptabilidad: utilizan algoritmos de aprendizaje adaptativo y auto-organización, por lo que ofrecen mejores posibilidades de procesado robusto y adaptativo.
- Procesado no Lineal: aumenta la capacidad de la red para aproximar funciones, clasificar patrones y aumenta su inmunidad frente al ruido.
- Procesado Paralelo: normalmente se usan un gran número de nodos de procesado, con alto nivel de interconectividad.

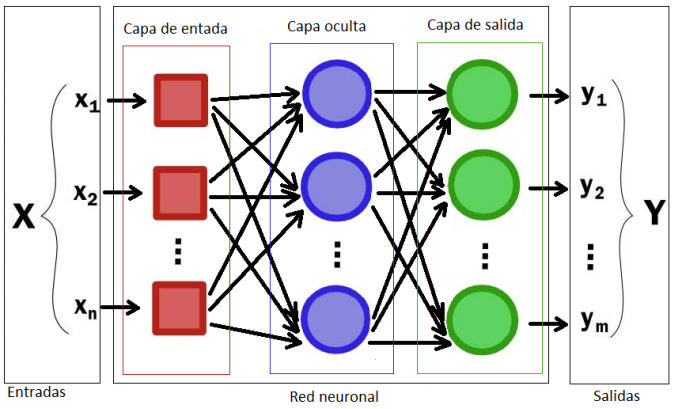


Figura 4-23 Esquema de red neuronal artificial con una capa oculta

Al elemento básico de computación se le llama habitualmente nodo o neurona. Recibe entradas desde otros nodos o de una fuente externa de datos. En la (4-30) se muestra un esquema general de una red neuronal artificial con n entradas, m salidas y una capa oculta. Cada entrada tiene un peso asociado w, que se va modificando en el proceso de aprendizaje. Cada nodo aplica una función f de la suma de las entradas ponderadas mediante los pesos

$$y_i = \sum_j w_{ij} y_j \ . \qquad \textbf{(4-31)}$$

La salida puede servir como entrada en otros nodos.

Hay dos fases en la modelización con redes neuronales (Hastie et al., 2005):

- Fase de entrenamiento: donde se usa un conjunto de datos o patrones de entrenamiento para determinar los pesos (parámetros) que definen el modelo de red neuronal. Se calculan de manera iterativa, de acuerdo con los valores de entrenamiento, con el objeto de minimizar el error cometido entre la salida obtenida por la red neuronal y la salida deseada.
- Fase de prueba: en la fase anterior, el modelo puede que se ajuste demasiado a las particularidades presentes en los patrones de entrenamiento, perdiendo su habilidad de generalizar su aprendizaje a casos nuevos (sobreajuste). Para evitar el problema del sobreajuste, es aconsejable utilizar un segundo grupo de datos diferentes a los de entrenamiento, el grupo de validación, que permita controlar el proceso de aprendizaje.

Normalmente, los pesos óptimos se obtienen optimizando (minimizando) alguna función de energía. Por ejemplo, un criterio muy utilizado es minimizar el error cuadrático medio entre el valor de salida y el valor real esperado.

Una red neuronal típica se puede caracterizar por la función de base y la función de activación. Cada nodo suministra un valor y_j a su salida. Este valor se propaga a través

de la red mediante conexiones unidireccionales hacia otros nodos de la red. Asociada a cada conexión hay un peso sináptico denominado $\{w_{ij}\}$, que determina el efecto del nodo j-ésimo sobre el nodo i-ésimo. Las entradas al nodo i-ésimo que provienen de los otros nodos son acumulados junto con el valor umbral θ_i, y se aplica la función base f, obteniendo u_i. La salida final y_i se obtiene aplicando la función de activación sobre u_i.

La función de base tiene dos formas típicas:

- Función lineal de tipo hiperplano: el valor de red es una combinación lineal de las entradas,

$$u_i(w, x) = \sum_{j=1}^{n} w_{ij} x_j \ . \quad \textbf{(4-32)}$$

- Función radial de tipo hiperesférico: es una función de base de segundo orden no lineal. El valor de red representa la distancia a un determinado patrón de referencia,

$$u_i(w, x) = \sqrt{\sum_{j=1}^{n} \left(x_j - w_{ij}\right)^2} \ . \quad \textbf{(4-33)}$$

El valor de red, expresado por la función de base, $u(w, x)$, se transforma mediante una función de activación no lineal. Las funciones de activación más comunes son la función sigmoidal,

$$f(u_i) = \frac{1}{1 + e^{-\frac{u_i}{\sigma^2}}} \ , \quad \textbf{(4-34)}$$

y gausiana,

$$f(u_i) = Ce^{-\frac{{u_i}^2}{\sigma^2}} \ . \quad \textbf{(4-35)}$$

Una red neuronal se determina por las neuronas y la matriz de pesos. Las neuronas se ordenan en capas existiendo tres tipos de capas: entrada, oculta y salida. Entre dos capas de neuronas existe una red de pesos de conexión, que puede ser de los siguientes tipos: hacia delante, hacia atrás, lateral y de retardo.

El tamaño de las redes depende del número de capas y del número de neuronas ocultas por capa. El número de unidades ocultas está directamente relacionado con las capacidades de la red. Para que el comportamiento de la red sea correcto, se tiene que determinar apropiadamente el número de neuronas de la capa oculta.

De entre las distintas arquitecturas de redes neuronales que existen, para el estudio particular de este trabajo se ha utilizado el perceptrón multicapa que está compuesto por una capa de entrada, una capa de salida y puede disponer de una o más capas ocultas. El número de capas ocultas es un parámetro a optimizar durante el diseño. Una capa oculta es suficiente para realizar una aproximación de cualquier función continua en un intervalo dado (Hornik, Stinchcombe, & White, 1989).

Para la construcción del clasificador se ha usado una sola capa oculta compuesta por diez neuronas mediante el uso de la función MATLAB *feedforwardnet(10)*. La capa de entrada está formada por dieciocho neuronas que se corresponden con cada una de las características extraídas y una capa de salida con una sola neurona.

Las conexiones entre neuronas son siempre hacia delante: las conexiones van desde las neuronas de una determinada capa hacia las neuronas de la siguiente capa; no hay conexiones laterales ni conexiones hacia atrás. Por tanto, la información siempre se transmite desde la capa de entrada hacia la capa de salida.

En la Figura 4-24 se compara el resultado obtenido aplicando el modelo construido en MATLAB mediante la función *train*. Puede observarse que las grabaciones de sapo corredor son generalmente bien clasificadas con un 100% para la vocalización estándar. Sin embargo ofrece bajos resultados para las grabaciones de sapo corredor para la vocalización de canto de suelta y sapo partero con índices de acierto del 40% y 47%, respectivamente. En la Figura 4-25 se presenta el resumen de las prestaciones obtenidas para cada clase y en la Tabla 4-10 se muestra la matriz de confusión del clasificador.

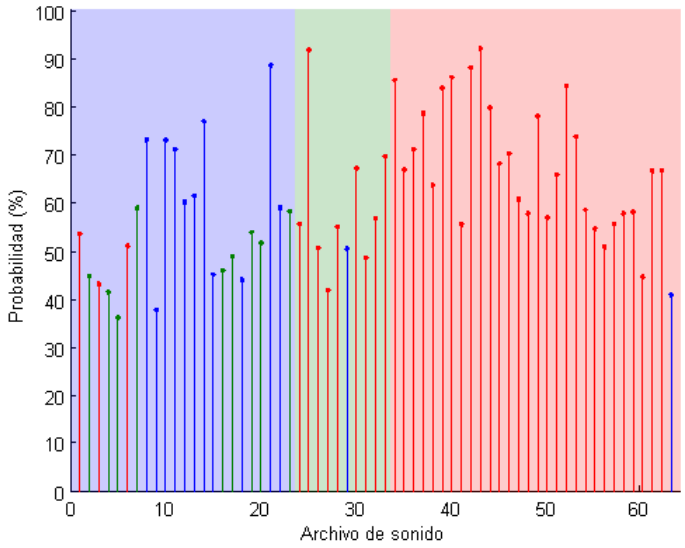


Figura 4-24 Clasificación de todas las grabaciones usando redes neuronales

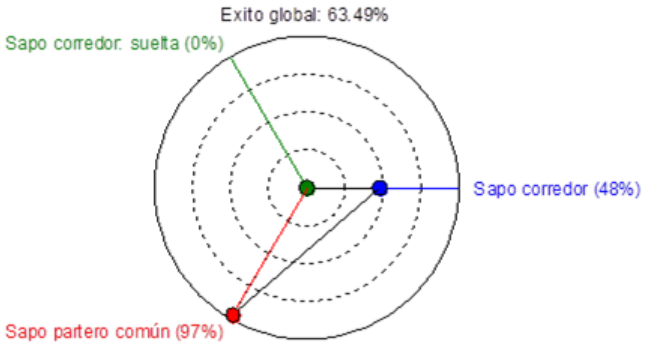


Figura 4-25 Prestaciones del clasificador redes neuronales

		Clase obtenida		
		1	2	3
Clase real	1	14	7	2
	2	1	0	9
	3	0	4	26

		Clase obtenida		
		1	2	3
Clase real	1	60.87%	30.43%	8.70%
	2	10.00%	70.00%	90.00%
	3	0.00%	13.33%	86.67%

Tabla 4-10 Matriz de confusión de la clasificación por redes neuronales

4.9. Función discriminante

El análisis discriminante es parte del análisis estadístico multivariante cuya finalidad es proporcionar técnicas estadísticas de clasificación a partir de la información proporcionada por el conjunto de entrenamiento. La información proporcionada por las variables de entrenamiento se canaliza a través de unas funciones matemáticas, denominadas funciones discriminantes, que son las que finalmente se utilizan en el proceso de clasificación.

El análisis discriminante equivale a un análisis de regresión donde la variable dependiente es categórica y tiene como categorías la etiqueta de cada una de las clases, y las variables independientes son continuas y determinan a qué clase pertenecen las muestras. Se trata de encontrar relaciones lineales entre las variables continuas que mejor discriminen las muestras de entrada en los grupos dados.

Haciendo uso de un enfoque bayesiano para un problema de dos clases, y suponiendo conocidas las probabilidades a priori de que una muestra pertenezca a una de las dos clases (π_1, π_2, tales que $\pi_1 + \pi_2 = 1$) y los costes de clasificación ($c(2|1)$ y $c(1|2)$ donde $c(i|j)$ es el coste de clasificar en la clase i una muestra que pertenece a la clase j), la probabilidad a posteriori de que una muestra este en una clase P_1 (Härdle & Simar, 2015) es

$$P(1|x_0) = \frac{P(x_0|1)P(1)}{P(x_0|1)P(1) + P(x_0|2)P(2)} \quad . \qquad \textbf{(4-36)}$$

Las probabilidades $P(x_0|1)$ y $P(x_0|2)$ vienen dadas por las funciones $\mathrm{f}_1(\mathrm{x})$ y $\mathrm{f}_2(\mathrm{x})$ se puede reescribir (4-36) como

$$P(1|x_0) = \frac{f_1(x_0)\pi_1}{f_1(x_0)\pi_1 + f_2(x_0)\pi_2} \quad . \tag{4-37}$$

De forma análoga para la segunda clase P_2 se tiene que

$$P(2|x_0) = \frac{f_2(x_0)\pi_2}{f_1(x_0)\pi_1 + f_2(x_0)\pi_2} \quad . \tag{4-38}$$

El coste promedio de clasificar mal en la clase 1 será

$$E(d_1) = 0 \cdot P(1|x_0) + c(1|2) \cdot P(2|x_0) = c(1|2)P(2|x_0) \quad . \tag{4-39}$$

De forma análoga para la clase 2 se tiene que

$$E(d_2) = c(2|1) \cdot P(1|x_0) + 0 \cdot P(2|x_0) = c(2|1)P(1|x_0) \quad . \tag{4-40}$$

La asignación de la nueva muestra se realizará en la clase 1 si su coste promedio es menor cumpliéndose que

$$E(d_1) < E(d_2) \rightarrow \frac{f_2(x_0)\pi_2}{c(2|1)} < \frac{f_1(x_0)\pi_1}{c(1|2)} \quad . \tag{4-41}$$

Si la probabilidad a priori es más alta y la verosimilitud de que pertenezca a la clase P_1, el coste de equivocarse al clasificarlo en P_1 es más bajo.

Como se ha supuesto que los costes y que las probabilidades a priori son iguales se tiene que

$$f_1(x_0) > f_2(x_0) \quad . \tag{4-42}$$

Tomando las funciones f_1 y f_2 como distribuciones normales con distintos vectores de medias pero idéntica matriz de varianza (Hastie et al., 2005) se obtiene las funciones

$$f_i(x) = \frac{1}{(2\pi)^{k/2}|V|^{1/2}} e^{\left(-\frac{1}{2}(x-\mu_i)'V^{-1}(x-\mu_i)\right)} \quad , \tag{4-43}$$

donde V es la matriz de covarianzas.

Tomando la expresión (4-43) en (4-42), como ambos miembros siempre son positivos se puede tomar logaritmo resultando

$$(x-\mu_1)'V^{-1}(x-\mu_1) < (x-\mu_2)'V^{-1}(x-\mu_2) \quad . \tag{4-44}$$

Haciendo uso de la definición de la distancia de Mahalanobis (Xiang, Nie, & Zhang, 2008)

$$D_i = (x - \mu_i)' V^{-1} (x - \mu_i) \ , \tag{4-45}$$

se puede decir que la muestra se asignará a la clase cuya media esté más próxima usando esta métrica.

Tomando

$$w = V^{-1}(\mu_2 - \mu_1) \, , \tag{4-46}$$

la frontera entre ambas clases a partir de la expresión (4-44) queda

$$w'x = w' \left(\frac{\mu_2 + \mu_1}{2} \right) \ , \tag{4-47}$$

donde $w'x$ es la función discriminante.

De forma resumida, cuando los costes y probabilidades a priori son iguales y las variables normales, el problema se reduce a definir una nueva variable escalar $z = w'x$ y la clasificación se realiza asignándola a la media más próxima que equivale a la distancia de Mahalanobis en el espacio origen.

En la Figura 4-26 se compara el resultado obtenido aplicando el modelo construido en MATLAB mediante la función *fitcdiscr*. Puede observarse que las grabaciones de sapo corredor son clasificadas correctamente con un 100% para la vocalización estándar. Sin embargo ofrece resultados mediocres para las grabaciones de sapo corredor para la vocalización de canto de suelta y sapo partero con índices de acierto del 40% y 50%, respectivamente. En la Figura 4-27 se presenta el resumen de las prestaciones obtenidas para cada clase y en la Tabla 4-11 se muestra la matriz de confusión del clasificador.

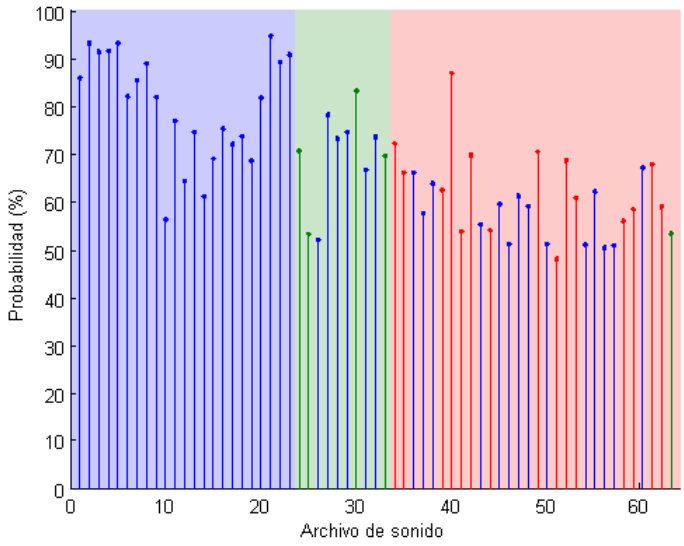


Figura 4-26 Clasificación de todas las grabaciones usando análisis discriminante

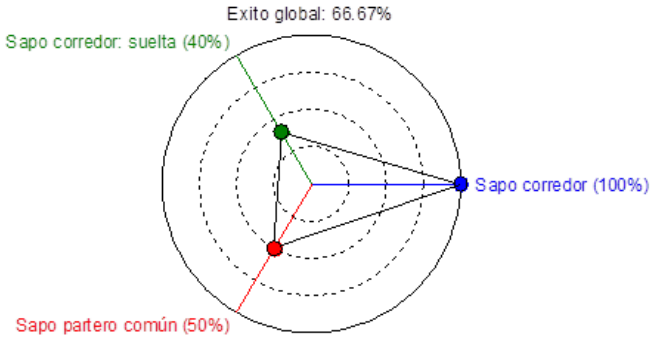


Figura 4-27 Prestaciones del clasificador análisis discriminante

		Clase obtenida		
		1	2	3
Clase real	1	23	0	0
	2	6	4	0
	3	14	1	15

		Clase obtenida		
		1	2	3
Clase real	1	100.00%	0.00%	0.00%
	2	60.00%	40.00%	0.00%
	3	46.67%	3.33%	50.00%

Tabla 4-11 Matriz de confusión de la clasificación por análisis discriminante

4.10. Clasificador bayesiano

El objetivo del clasificador bayesiano es minimizar el coste total de los errores cometidos (Hastie et al., 2005). En el caso particular de la función de pérdida, el problema se convierte en asignar la clase con mayor probabilidad a posteriori. La función de distribución conjunta es desconocida pero puede ser estimada a partir de una muestra aleatoria simple formada por el conjunto de entrenamiento.

Los clasificadores bayesianos son ampliamente utilizados debido a que presentan ciertas ventajas (Araujo, 2006):

- Generalmente, son fáciles de construir y de entender.
- Las inducciones son extremadamente rápidas.
- Es muy robusto.
- Toma evidencia de muchos atributos para realizar la predicción final.

En los clasificadores bayesianos se obtiene la probabilidad posterior de cada clase , θ_i, usando la regla de Bayes, como el producto de la probabilidad a priori de la clase por la probabilidad condicionada de los atributos de cada clase, k, dividido por la probabilidad de los atributos (4-48) (Han et al., 2011),

$$P(\theta_i|k) = \frac{P(\theta_i)P(k|\theta_i)}{P(k)} \quad . \qquad \textbf{(4-48)}$$

El clasificador bayesiano simple, *naive Bayes classifer NBC*, asume que los atributos/características son independientes entre sí dada la clase, por lo que la

probabilidad se puede obtener por el producto de las probabilidades condicionales individuales de cada atributo dado el nodo clase

$$P(\theta_i|k) = \frac{P(\theta_i)P(k_1|\theta_i) \cdot \ldots \cdot P(k_n|\theta_i)}{P(k)} \ , \qquad \textbf{(4-49)}$$

donde n es el número de atributos, incrementándose linealmente el número de parámetros en función de n, en lugar de hacerlo exponencialmente.

Con esta simplificación sólo es necesario estimar el vector de probabilidades a priori de cada clase, $P(\theta)$, y la matriz de probabilidad condicional para cada atributo dado la clase, $P(k_i|\theta)$. Estos parámetros se pueden estimar a partir de los datos de entrenamiento en base a frecuencia. El denominador, $P(k)$, no es necesario al ser un valor constante independiente de la clase.

Una limitación de este clasificador es que normalmente los atributos no son condicionalmente independientes. Una forma de resolver esta limitación es agregando una estructura de dependencias entres los atributos. Existen dos alternativas básicas:

- Clasificador bayesiano simple aumentado con un árbol, TAN.
- Clasificador bayesiano simple aumentado con una red, BAN.

La desventaja de estas soluciones es que aumenta la complejidad y el tiempo de aprendizaje y clasificación.

Otra alternativa para resolver la limitación del NBC es transformar la estructura manteniendo su estructura inicial introduciendo las operaciones de: eliminación, unión/combinación e introducción de un nodo oculto para separar dos atributos. Esta modificación se puede sistematizar calculando la dependencia de pares de atributos mediante la información mutua condicional, IMC,

$$I(X_i, X_j|\theta) = \sum_{X_i, X_j} P(X_i, X_j|\theta) \log\left(\frac{P(X_i, X_j|\theta)}{P(X_i|\theta)P(X_j|\theta)}\right) \ . \qquad \textbf{(4-50)}$$

Para el par con mayor IMC probar las tres operaciones básicas evaluando las tres estructuras y la original para quedarse con la mejor opción. Este proceso se repite hasta que no se mejore el clasificador.

En la Figura 4-28 se compara el resultado obtenido aplicando el modelo construido en MATLAB mediante la función *fitNativeBayes*. Puede observarse que las grabaciones de sapo partero son clasificadas correctamente con un 96.67%. Sin embargo ofrece unos malos resultados para las grabaciones del sapo corredor con índices de acierto del 65.22% y 10%. En la Figura 4-29 se presenta el resumen de las prestaciones obtenidas para cada clase y en la Tabla 4-12 se muestra la matriz de confusión del clasificador.

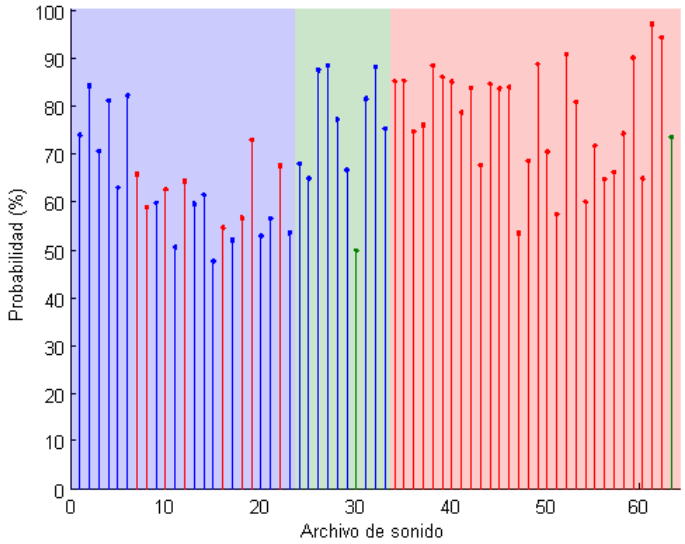


Figura 4-28 Clasificación de todas las grabaciones usando clasificador bayesiano

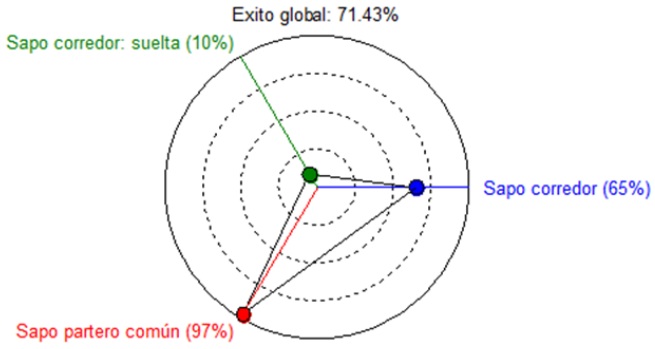


Figura 4-29 Prestaciones del clasificador bayesiano

		Clase obtenida		
		1	**2**	**3**
Clase real	**1**	15	0	8
	2	9	1	0
	3	0	1	29

		Clase obtenida		
		1	**2**	**3**
Clase real	**1**	65.22%	0.00%	34.78%
	2	90.00%	10.00%	0.00%
	3	0.00%	3.33%	96.67%

Tabla 4-12 Matriz de confusión de la clasificación bayesiana

4.11. Comparación de técnicas de clasificación no secuencial

A lo largo de este capítulo se han ido presentando distintas técnicas de clasificación no secuencial. En la Tabla 4-13 se recoge una comparación entre los distintos clasificadores estudiados en función de la tasa de éxito global y de forma individual por cada tipo de canto estudiado.

Algoritmos	Canto 1 (23) Aciertos		Canto 2 (10) Aciertos		Canto 3 (30) Aciertos		Total (63) Aciertos	
Distancia mínima	14	61%	10	100%	0	0%	24	38.10%
Máxima verosimilitud	22	93%	5	50%	23	77%	50	79.37%
Árboles de decisión	20	87%	7	70%	28	93%	55	87.30%
k-vecinos más próximos	23	100%	5	50%	18	60%	46	73.02%
SVM	23	100%	2	20%	21	70%	46	73.02%
Regresión logística	23	100%	4	40%	14	47%	41	65.02%
Redes neuronales	11	48%	0	0%	29	97%	40	63.49%
Función discriminante	23	100%	4	40%	15	50%	42	66.67%
Clasificador bayesiano	15	65%	1	10%	29	97%	45	71.43

Tabla 4-13. Resultados de la clasificación no secuencial

La Figura 4-30 refleja de forma gráfica los mismos resultados expuestos en la Tabla 4-13. La altura de la barra refleja la tasa global de éxito para cada clasificador. Los puntos (con el código de colores habitual) indican la tasa de éxito para cada tipo de canto.

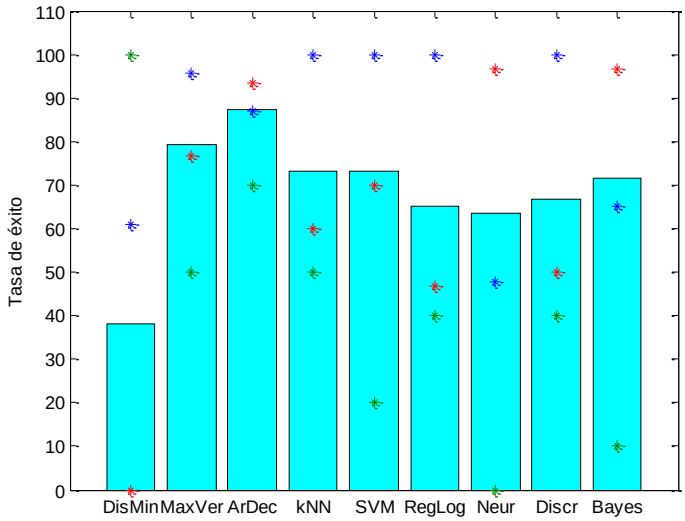


Figura 4-30. Resultados de la clasificación no secuencial

Para la selección de un método de clasificación no sólo es importante la tasa de éxito global sino también que esa tasa esté equilibrada para los distintos tipos de cantos. Para considerar este segundo factor se usará como indicador el rango, R, de la tasa de éxito, definido como

$$R = \max_i E_i - \min_i E_i \ , \qquad \textbf{(4-51)}$$

siendo E_i la tasa de éxito del algoritmo de clasificación para el tipo de canto i-ésimo. La Figura 4-31 se enfrenta para cada clasificador la tasa de error global y el rango.

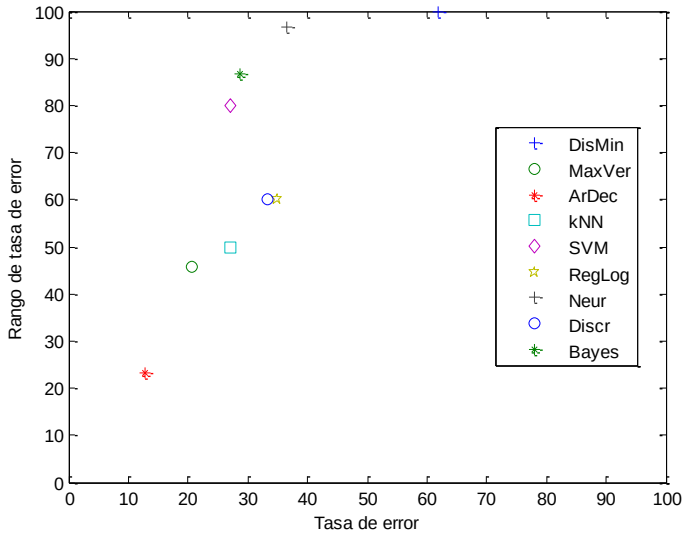


Figura 4-31. Tasa de error y su rango (unidades en %)

Un buen algoritmo de clasificación debe tener una tasa de error $\bar{E}$ baja y un rango R bajo. Cuanto más cerca al origen se encuentre un algoritmo, mejor será. Una buena medida que combina ambos valores es, pues, la distancia al origen D que se calcula como

$$D = \sqrt{\bar{E}^2 + R^2} \ . \qquad \textbf{(4-52)}$$

De la expresión anterior se puede fácilmente derivar un factor de mérito normalizado (entre 0 y 1) mediante la expresión

$$M \equiv 1 - \frac{D}{D_{max}} = 1 - \frac{D}{\sqrt{2}} \ . \qquad \textbf{(4-53)}$$

El valor de este factor de mérito para cada clasificador estudiado se recoge en la Tabla 4-14.

Algoritmos	Aciertos	Errores	Rango	Distancia origen	Mérito
Distancia mínima	38.10%	61.90%	100%	1.18%	16.84%
Máxima verosimilitud	79.37%	20.63%	43%	0.48%	66.28%
Árboles de decisión	87.30%	12.70%	23%	0.26%	81.42%
k-vecinos más próximos	73.02%	26.98%	50%	0.57%	59.83%
SVM	73.02%	26.98%	80%	0.84%	40.83%
Regresión logística	65.08%	34.92%	60%	0.69%	50.91%
Redes neuronales	63.49%	36.51%	97%	1.04%	26.71%
Función discriminante	66.67%	33.33%	60%	0.69%	51.47%
Clasificador bayesiano	71.43%	28.57%	87%	0.92%	35.25%

Tabla 4-14. Factor de mérito de clasificadores no secuenciales

Un método usual para evaluar el funcionamiento de los clasificadores es similar al empleado en la evaluación de pruebas diagnósticas en medicina. En un problema de

clasificación binaria, en la que los resultados se etiquetan como positivos o negativos, hay cuatro posibles resultados que se muestran en la matriz de confusión mostrada en la Tabla 4-15. Los casos correctos son la suma de verdaderos positivos y verdaderos negativos, mientras que los errores cometidos son la suma de falsos positivos y falsos negativos.

		Clase obtenida	
		Positivo	**Negativo**
Clase real	**Positivo**	Verdadero positivo TP (*true positive*)	Falso negativo FN (*false negative*)
	Negativo	Falso positivo FP (*false positive*)	Verdadero negativo TN (*true negative*)

Tabla 4-15. Factor de mérito de clasificadores no secuenciales

En base a estos casos, se pueden definir diversos parámetros que permiten evaluar los clasificadores. Los más usuales son:

- Sensibilidad: proporción de muestras correctamente clasificadas como positivas,

$$SNS = \frac{TP}{TP + FN} \quad . \tag{4-54}$$

- Especificidad: proporción de muestras correctamente clasificadas como negativas,

$$SPC = \frac{TN}{TN + FP} \quad . \tag{4-55}$$

- Exactitud: capacidad del clasificador de acercarse al resultado real,

$$ACC = \frac{TP + TD}{TP + TN + FP + FN} \quad . \tag{4-56}$$

- Precisión: indica la eficacia real del clasificador,

$$PRC = \frac{TP}{TP + FP} \quad . \tag{4-57}$$

- Tasa de error: indica el porcentaje de errores,

$$ERR = \frac{FP + FN}{TP + TN + FP + FN} \quad . \tag{4-58}$$

En la Tabla 4-16 se recogen los valores de los indicadores anteriores para cada uno de los clasificadores estudiados.

Algoritmo	Exactitud	Tasa de errores	Precisión	Sensib.	Especif.
Distancia mínima	58.73%	41.27%	-	53.62%	75.47%
Máxima verosimilitud	86.24%	13.76%	79.37%	74.11%	89.58%
Árboles de decisión	91.53%	8.47%	87.05%	83.43%	93.01%
k-vecinos más próximos	82.01%	17.99%	85.83%	70.00%	85.83%
SVM	82.01%	17.99%	75.21%	63.33%	86.04%
Regresión logística	76.72%	23.28%	77.42%	62.22%	81.87%
Redes neuronales	75.66%	24.34%	54.53%	49.18%	81.14%
Función discriminante	77.78%	22.22%	77.83%	63.33%	82.70%
Clasificador bayesiano	80.95%	19.05%	63.63%	57.29%	83.79%

Tabla 4-16. Indicadores para la evaluación de clasificadores

La decisión de clasificación se establece mediante un umbral sobre los resultados de las pruebas realizadas. En función de este umbral, se obtienen diferentes resultados de sensibilidad y especificidad para un determinado método. Las denominadas curvas ROC (*Receiver Operating Characteristics*), desarrolladas inicialmente en aplicaciones de radar, se han usado en distintos ámbitos como la medicina, psicología,… y más recientemente en el aprendizaje automático.

Las curvas ROC fueron introducidas en el campo de la minería de datos por (Spackman, 1989) y el análisis de la curva ROC se ha convertido en una poderosa herramienta para la evaluación y comparación de distintos algoritmos de clasificación (Fawcett, 2006). Una posible representación enfrenta los valores de sensibilidad y especificidad. Dado que los verdaderos positivos equivalen a la sensibilidad y los falsos positivos a la especificidad.

Otro modo de representar curvas ROC es mediante la razón de verdaderos positivos frente a la razón de falsos positivos. Un clasificador perfecto se situaría en la esquina superior izquierda del gráfico, correspondiéndose con un 100% de verdaderos positivos y 0% de falsos positivos.

En la Figura 4-32 se representa el análisis ROC de los algoritmos estudiados. En esta figura, la diagonal dibujada en trazos discontinuos representa una clasificación aleatoria y se le conoce con el nombre de línea de no-discriminación. Esta diagonal divide el espacio en dos regiones, donde el espacio por encima de la diagonal representa un mejor resultado de clasificación respecto al azar, y los puntos por debajo un peor resultado respecto al azar.

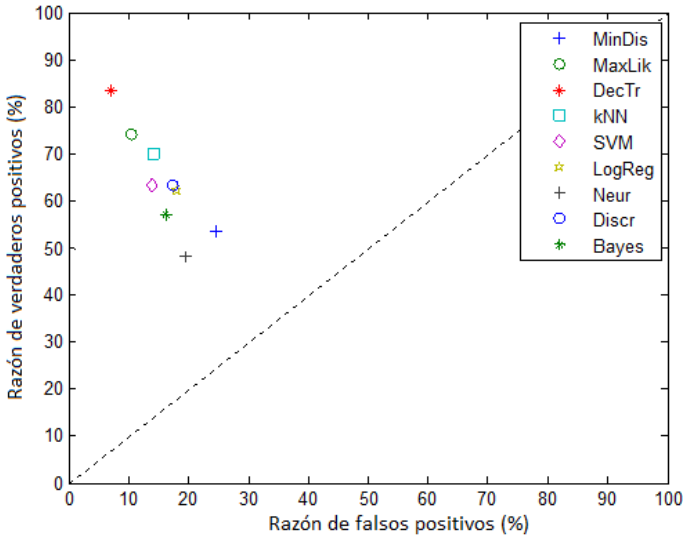


Figura 4-32 Comparación de los métodos de clasificación mediantes análisis ROC

La validez de la comparación puede verse severamente afectada si se utilizan mediciones pocos fiables.

Otra forma de compara los clasificadores es conociendo el grado de concordancia entre los distintos clasificadores, es decir, hasta qué punto dos clasificadores coinciden. Para determina el grado de concordancia entre dos clasificadores se dispone de una herramienta estadística, el coeficiente kappa de Cohen (κ) (Cohen, 1960). De forma simplificada, el coeficiente de kappa corresponde a la proporción de concordancias observadas sobre el total de observaciones, habiendo excluido las concordancias atribuibles al azar.

Desde el punto de vista probabilístico, el coeficiente kappa es la probabilidad condicional de acuerdo entre los clasificadores, dado que las clasificaciones no son independientes entre sí, estando correlacionadas. En el caso que los clasificadores sean completamente independientes el valor de los coeficientes Kappa será cero. Por otro lado, si el acuerdo entre los clasificadores es completo el valor de los coeficientes kappa será uno. Para determinar hasta qué punto la concordancia observada es superior a la que es esperable obtener por puro azar se utiliza el índice de concordancia kappa

$$\kappa = \frac{Pr(a) - Pr\,(e)}{1 - Pr\,(e)}\,, \tag{4-59}$$

donde $Pr(a)$ es el acuerdo observado relativo entre los clasificadores comparados, y $Pr(e)$ es la probabilidad hipotética de acuerdo por azar.

Una vez calculados los coeficientes kappa entre todos los clasificadores estudiados, para facilitar la comparación de los métodos, los resultados se enfrentan en la matriz de la Figura 4-33.

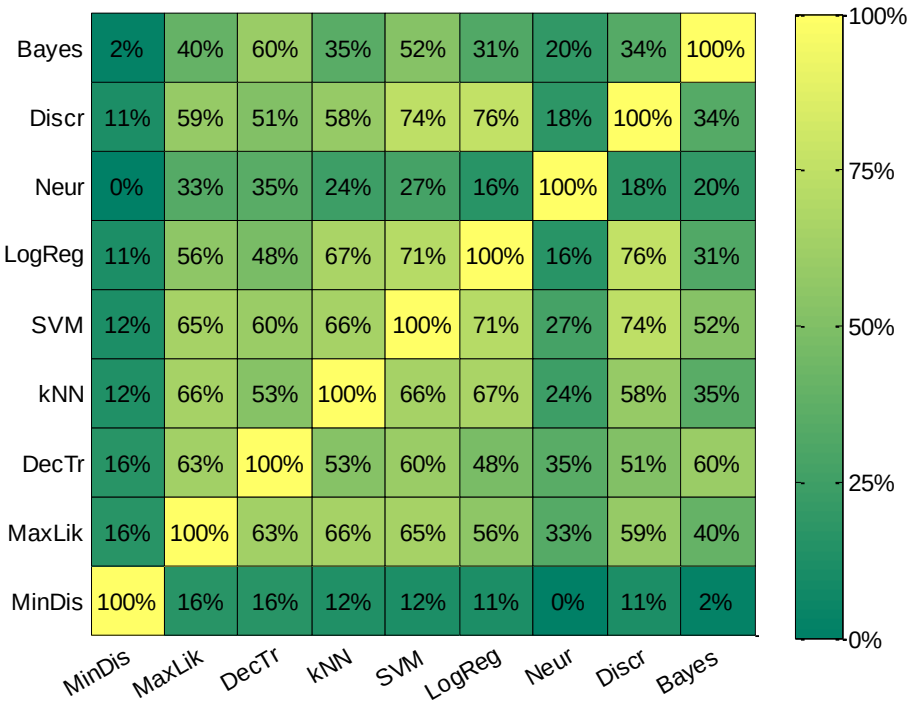


Figura 4-33 Comparación de los métodos de clasificación mediantes coeficientes kappa de Cohen

La conclusión tras analizar las distintas comparativas realizadas, es que para una clasificación automática no secuencial de los sonidos usando características basadas en el estándar MPEG-7, el árbol de decisión ofrece los mejores resultados, entre los métodos de clasificación estudiados. Este método obtiene una tasa de error del 8.47%, lo que se considera un buen resultado teniendo en cuenta la mala calidad de los archivos de sonidos procesados.

CAPÍTULO 5. CLASIFICACIÓN DE SECUENCIA TEMPORALES

5.1. Introducción a la clasificación secuencial

En el capítulo anterior se han aplicado diversas técnicas de clasificación al problema de clasificación de un sonido. Para ello se ha dividido el sonido en una secuencia de *frames* y obtenido de cada uno de ellos un conjunto de parámetros de acuerdo con los especificaciones de la norma MPEG-7 (ISO, 2001). Por tanto, el punto de partida de los algoritmos de clasificación es una nube de puntos en un espacio $\mathbb{R}^{18}$, es decir, un espacio de 18 dimensiones. Cada punto en dicho espacio se corresponde con un *frame* de sonido.

Con dicho enfoque el mejor resultado se obtiene, según se vio, con un algoritmo de clasificación basado en un árbol de decisión, que consigue una tasa de errores del 12.70%, un valor razonablemente bajo considerando la baja calidad de las grabaciones de sonido disponibles.

Sin embargo, en el método de clasificación anteriormente descrito, todos los puntos de la nube (todos los *frames*) se consideran sin ligadura unos con otros. En ningún momento se tiene en cuenta durante la clasificación el hecho de que los *frames* están ordenados en una secuencia temporal. Es decir, se clasifica cada *frame* de manera aislada, en base únicamente a sus propios parámetros, y sin tener en cuenta los *frames* anteriores o posteriores. En este capítulo se estudiaran distintos métodos de clasificación teniendo en cuenta el carácter secuencial de los *frames* para tratar de mejorar los resultados anteriores de clasificación.

5.2. Parámetros temporales

Una primera aproximación al problema se basa en la construcción de parámetros, para cada *frame*, que identifiquen de alguna forma su ligazón con los *frames* anteriores y

posteriores, es decir, la construcción de parámetros que, si bien lo son para cada *frame* considerado aisladamente, en cierta medida llevan información de la secuencia temporal. Los denominaremos “parámetros temporales”.

Observando con detalle los espectrogramas de los sonidos, se puede comprobar que en determinados sonidos, se observa una importante variación del espectrograma de un *frame* al siguiente. Así ocurre, por ejemplo, en el canto de un sapo corredor con vocalización estándar. Por el contrario en el caso del canto del sapo partero con vocalización estándar, el espectrograma permanece sensiblemente constante a lo largo de todo el canto. Este comportamiento puede observarse en la Figura 5-1 para dos sonidos de ejemplo.

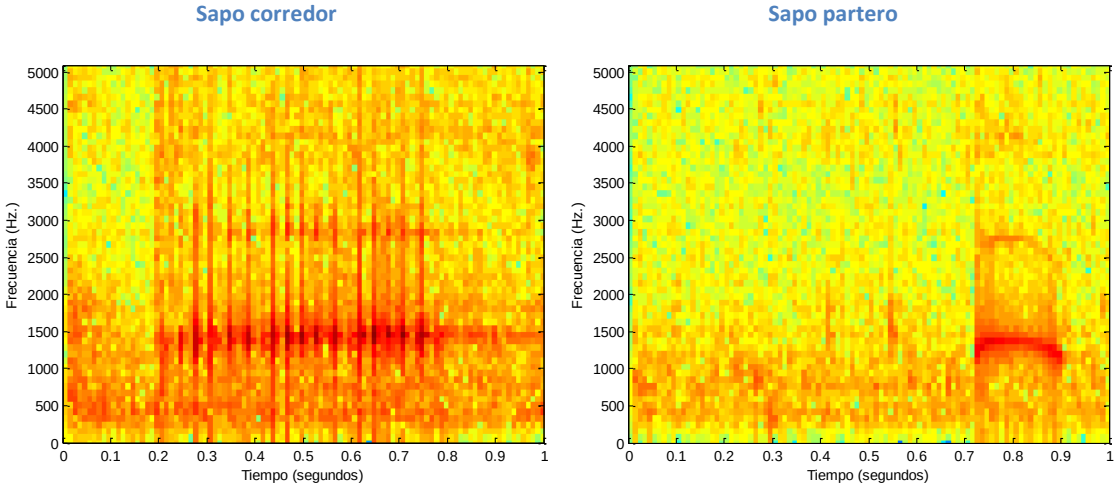


Figura 5-1 Variación del espectrograma durante el canto de sapo corredor y partero en vocalización estándar

Trazando la evolución temporal del parámetro primario “potencia total del *frame*”, a los sonidos anteriores, se observar grandes variaciones del parámetro durante el momento del canto para el sapo corredor, mientras que para el sapo partero el parámetro es más estable, como se puede ver en la Figura 5-2.

Para construir un parámetro que refleje esta característica se define un “segmento” centrado en cada *frame* y que contiene unos pocos *frames*. Para este trabajo se ha definido el “segmento” con una duración de $100\ ms$, es decir, que contiene 10 *frames*. En este segmento se calcula la dispersión de cada uno de los parámetros primarios, dando lugar a los que denominamos parámetros secundarios de *frame*.

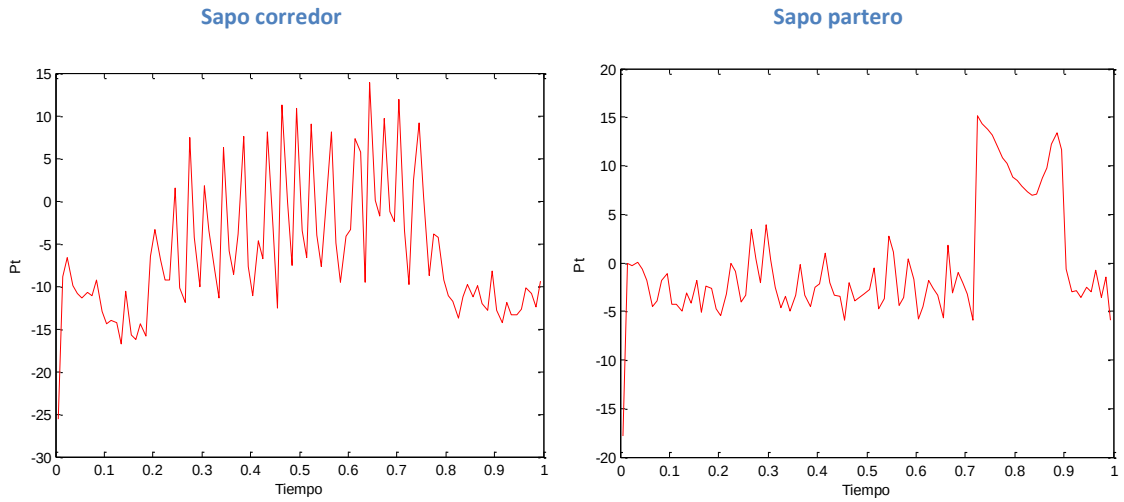


Figura 5-2 Potencia total de un *frame*

La medida de la dispersión de los parámetros puede realizarse con diversos indicadores como, por ejemplo, la desviación típica. Se ha seleccionado el rango intercuartil (*iqr: interquartile range*), es decir, el rango entre el percentil 25 y el percentil 75. Este indicador es menos sensible que la desviación típica ante valores extremos, igual que le ocurre a la mediana frente a la media. Para los sonidos anteriores, los resultados correspondientes a la dispersión de la potencia total en un segmento pueden verse en la Figura 5-3.

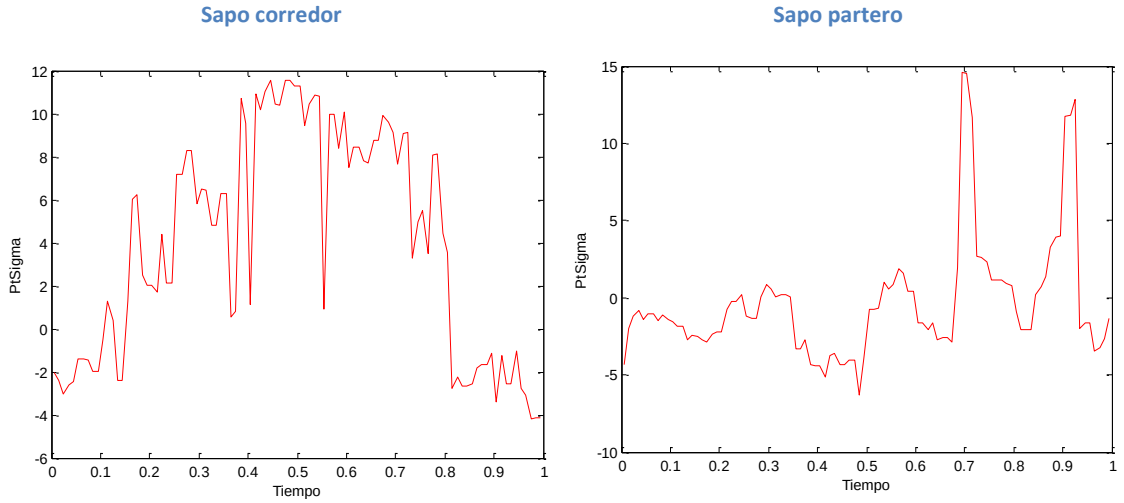
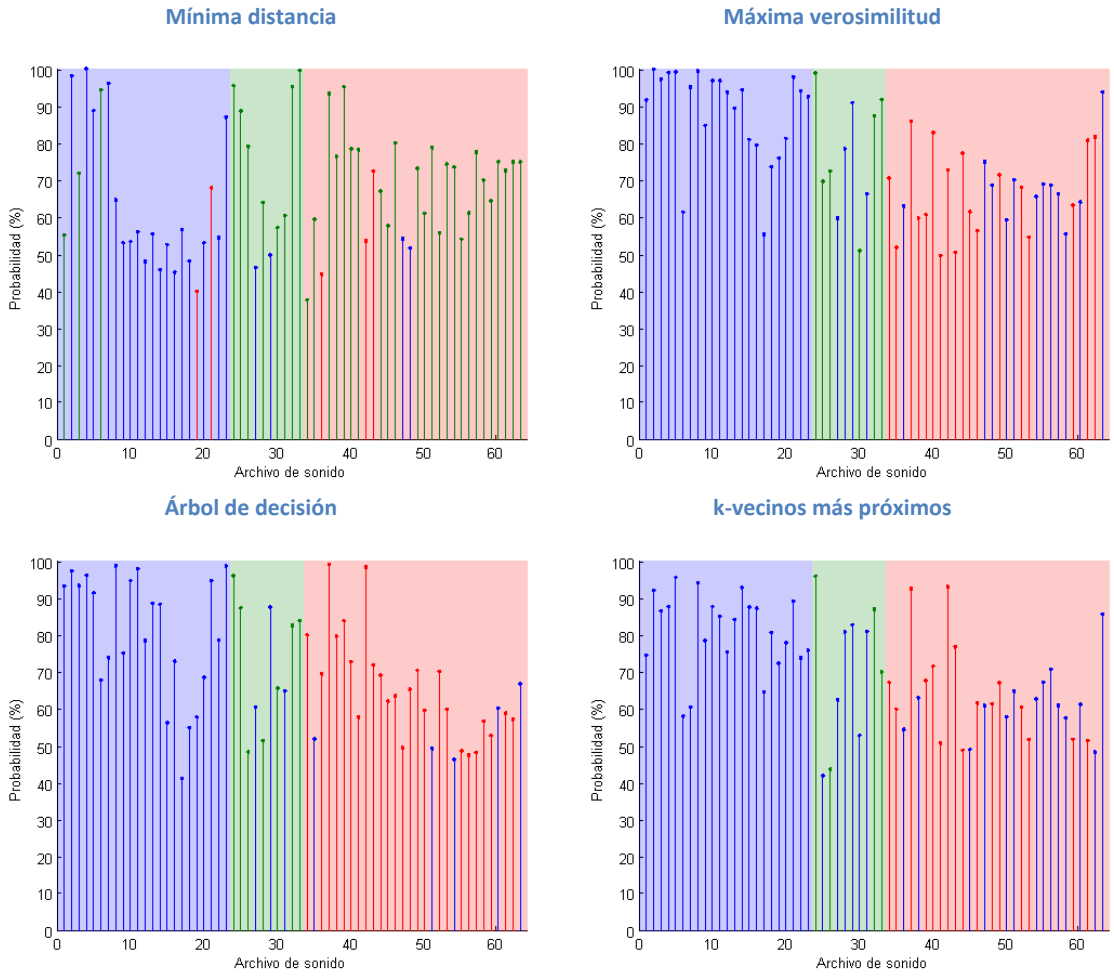


Figura 5-3 Dispersión de la potencia total de un *frame*

Como se puede observar, la dispersión de un parámetro dentro de un segmento centrado en cada *frame* puede constituir un importante parámetro secundario para la clasificación de dicho *frame*. Si se repite este análisis para cada uno de los parámetros primarios del *frame* se obtienen tantos parámetros secundarios como primarios. En definitiva, utilizando esta estrategia se define cada *frame* mediante 36 parámetros: 18 primarios y 18 secundarios; o lo que es lo mismo, mediante un punto en un espacio $\mathbb{R}^{36}$.

Sobre este nuevo espacio se aplican los procedimientos establecidos en el capítulo anterior sobre clasificación no secuencial. La Figura 5-4 muestra el resultado de la aplicación de cada una de estas técnicas al conjunto de archivos de sonido disponibles. En horizontal se representan los archivos, ordenados por tipo de sonido: sapo corredor (zona azul); sapo corredor en canto de suelta (zona verde); sapo partero (zona roja). Por cada archivo existe una línea vertical cuyo color se corresponde con la clasificación realizada por el algoritmo (con el mismo código de colores anterior). En una clasificación perfecta el código de cada línea debería corresponder con la de la zona del gráfico. Cada discrepancia supone un error de clasificación. Por último, la altura de cada línea es la probabilidad que el algoritmo asigna a la clasificación realizada.



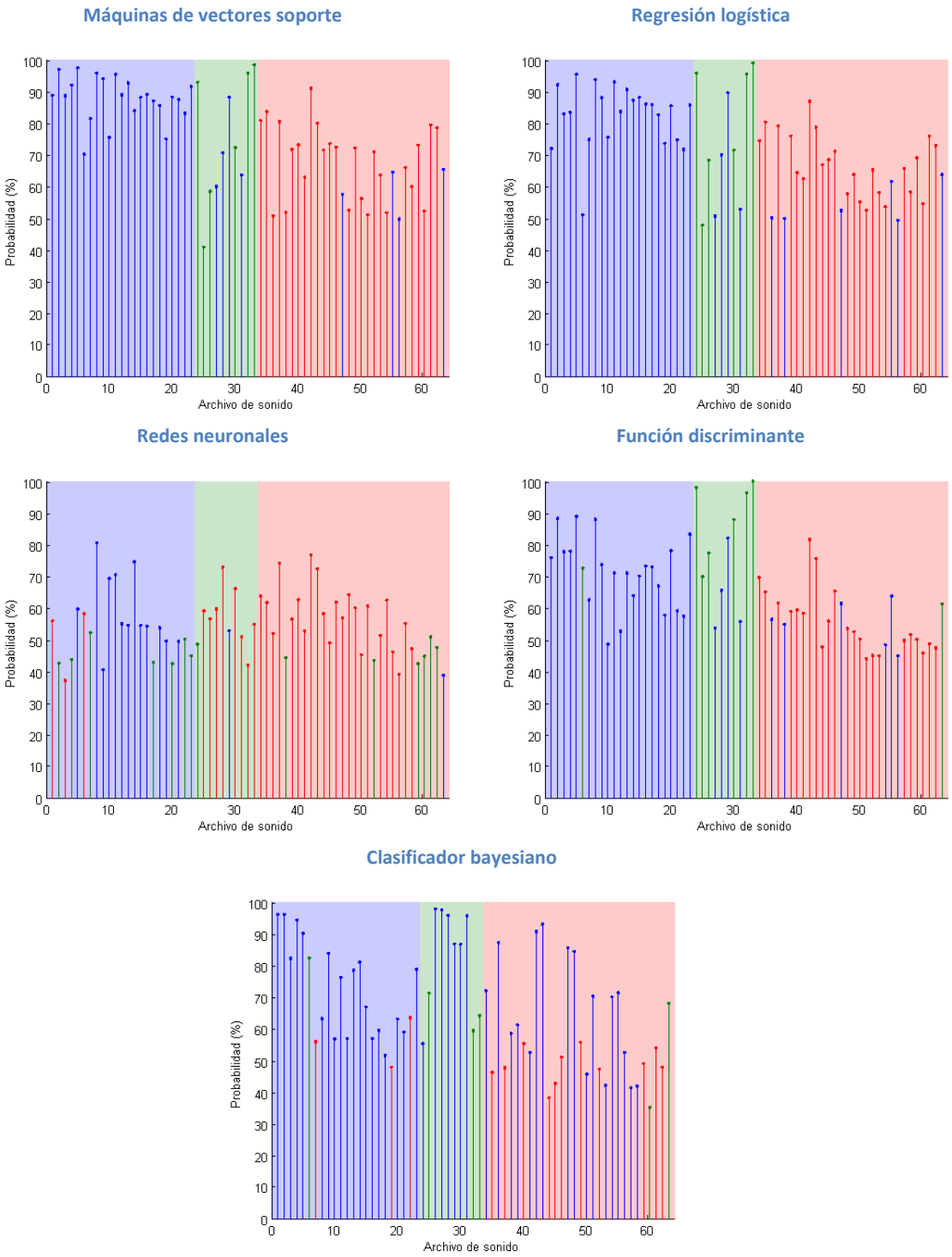


Figura 5-4 Clasificación con parámetros temporales

El resultado global de la clasificación de cada método puede resumirse en la Figura 5-5.

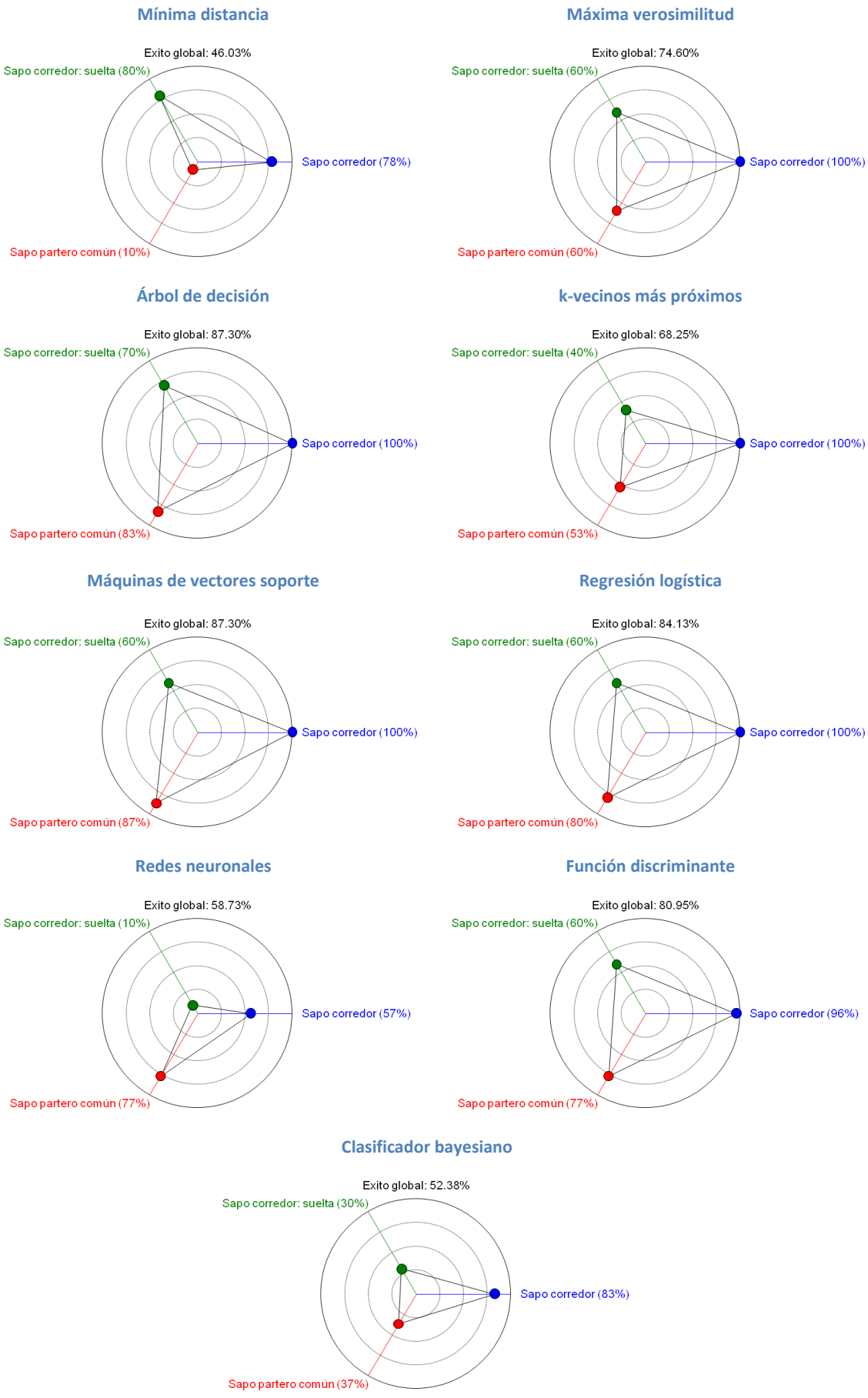


Figura 5-5 Resumen de la clasificación con parámetros temporales

De forma análoga al capítulo anterior, a continuación se realizan distintos métodos de evaluación de los clasificadores con el objetivo de determinar cuál es el algoritmo que mejores resultado obtiene.

La **¡Error! No se encuentra el origen de la referencia.** refleja de forma gráfica la omparación de los distintos algoritmos. La altura de la barra refleja la tasa global de éxito para cada clasificador. Los puntos, con el código de colores habitual, indican la tasa de éxito para cada tipo de canto.

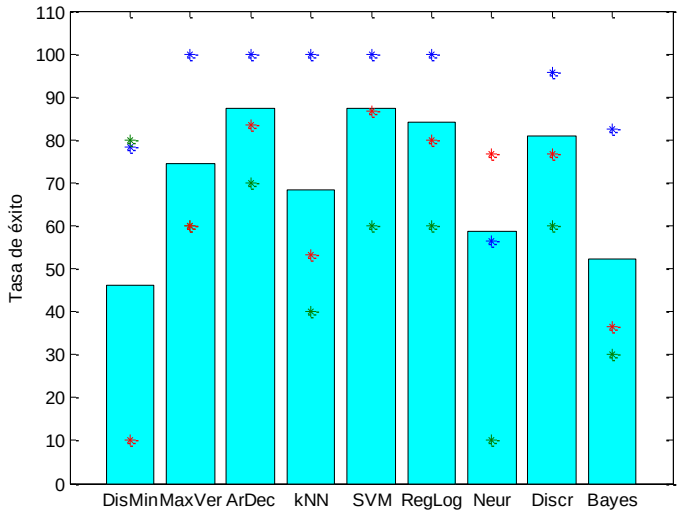


Figura 5-6 Resumen de la clasificación con parámetros temporales

La Figura 5-7 refleja el mérito de cada técnica de clasificación mediante la representación del rango de la tasa de error frente la tasa de error.

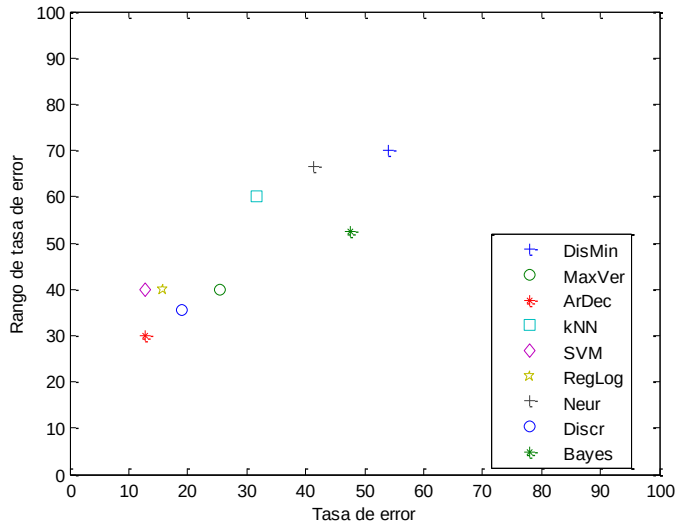


Figura 5-7 Tasa de error y su rango (unidades en %) para la clasificación con parámetros temporales

La Tabla 5-1 recoge el factor de mérito de cada uno de los clasificadores utilizados. Se puede ver que el mejor de todos ellos sigue siendo el de clasificación mediante árboles de decisión, aunque la figura de mérito es sensiblemente menor en $\mathbb{R}^{36}$.

Algoritmo	Acierto	Errores	Rango	Distancia origen	Mérito
Distancia mínima	46.03%	53.97%	70%	0.88	37.50%
Máxima verosimilitud	74.60%	25.40%	40%	0.47	66.50%
Árboles de decisión	87.30%	12.70%	30%	0.33	76.96%
k-vecinos más próximos	68.25%	31.75%	60%	0.68	52.00%
SVM	87.30%	12.70%	40%	0.42	70.32%
Regresión logística	84.13%	15.87%	40%	0.43	69.57%
Redes neuronales	58.73%	41.27%	67%	0.78	44.56%
Función discriminante	80.95%	19.05%	36%	0.40	71.42%
Clasificador bayesiano	52.38%	47.62%	53%	0.71	49.82%

Tabla 5-1. Factor de mérito de clasificadores con parámetros temporales

La Tabla 5-2 recoge los valores de los indicadores de exactitud, precisión, sensibilidad, especificidad y tasa de errores. Se puede ver que el mejor método de clasificación sigue siendo el árbol de decisión. Cabe destacar que la precisión, sensibilidad y especificidad es sensiblemente mayor en $\mathbb{R}^{36}$.

Algoritmo	Exactitud	Tasa de errores	Precisión	Sensib.	Especif.
Distancia mínima	64.02%	35.98%	54.68%	56.09%	77.04%
Máxima verosimilitud	83.07%	16.93%	86.32%	73.33%	86.67%
Árboles de decisión	91.53%	8.47%	91.40%	84.44%	93.33%
k-vecinos más próximos	78.84%	21.16%	84.50%	64.44%	83.33%
SVM	91.53%	8.47%	91.40%	82.22%	93.33%
Regresión logística	89.42%	10.58%	89.90%	80.00%	91.67%
Redes neuronales	72.49%	27.51%	53.82%	47.73%	79.05%
Función discriminante	87.30%	12.70%	81.25%	77.44%	90.41%
Clasificador bayesiano	68.25%	31.75%	57.59%	49.76%	75.08%

Tabla 5-2. Indicadores para la evaluación de clasificadores con parámetros temporales

En la Figura 5-8 se representa el análisis ROC de los distintos algoritmos estudiados donde de nuevo el mejor clasificador es el árbol de decisión.

En la Figura 5-9 se representa la concordancia de los resultados usando los coeficientes kappa.

La conclusión tras analizar las distintas comparativas realizadas, es que la clasificación por árbol de decisión ofrece los mejores resultados cuando se usan los parámetros temporales.

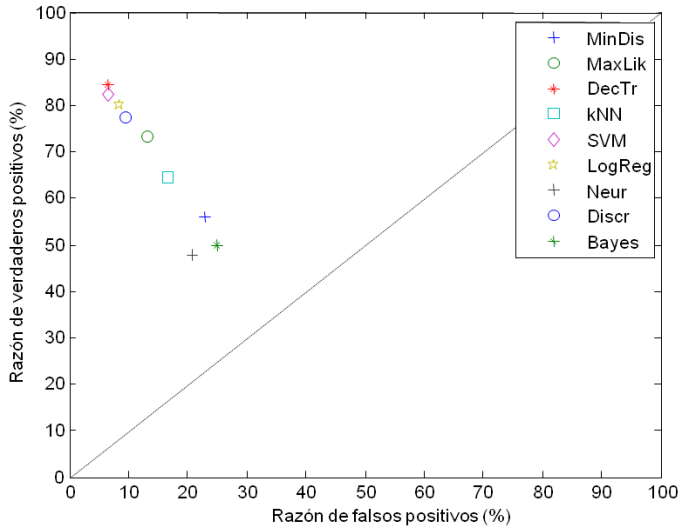


Figura 5-8 Comparación de los métodos de clasificación con parámetros temporales mediantes análisis ROC

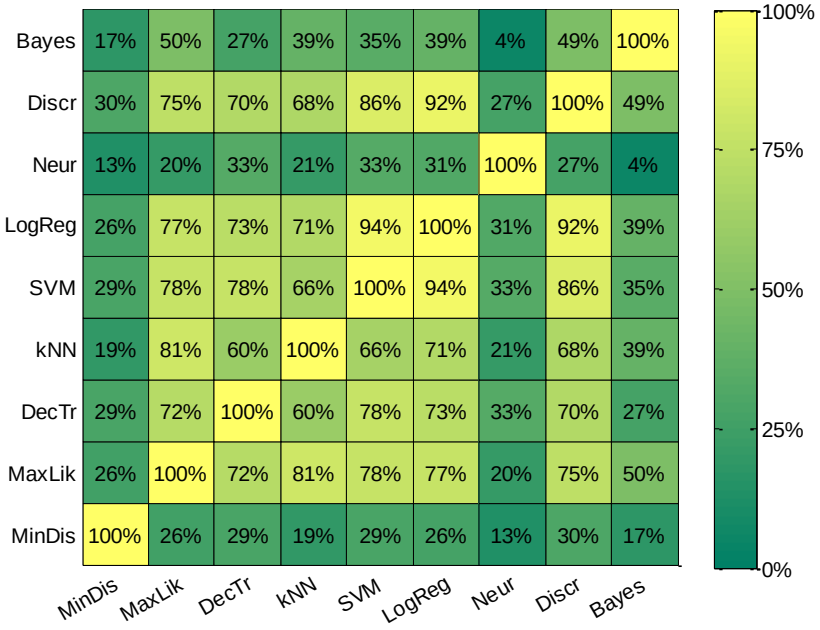


Figura 5-9 Comparación de los métodos de clasificación con parámetros temporales mediantes coeficientes kappa de Cohen

5.3. Reducción de dimensionalidad

El disponer de p parámetros de cada *frame* no obliga necesariamente a considerarlos todos la hora de la clasificación. Por razones de eficiencia computacional conviene reducir la dimensionalidad de la nube de puntos, o lo que es lo mismo, el número de parámetros que identifica cada *frame*.

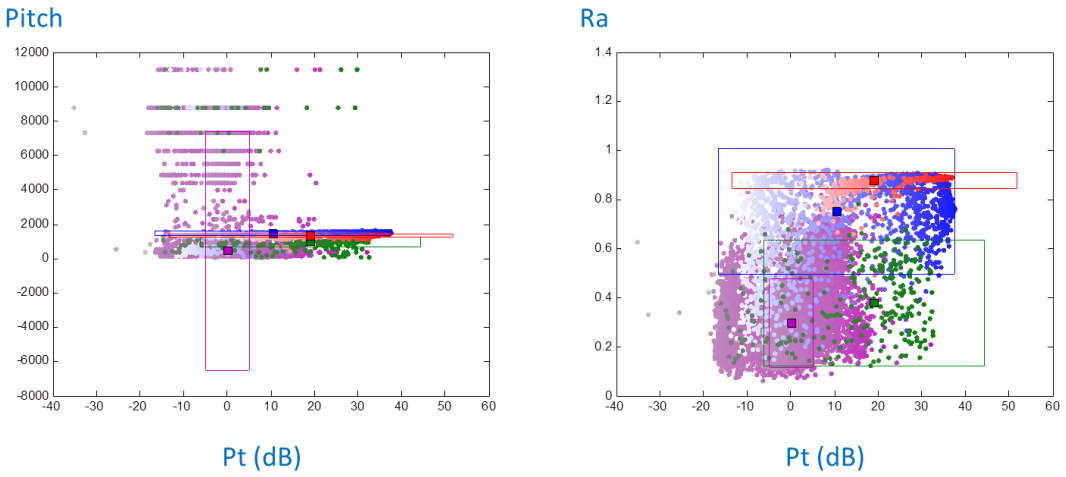
Como se ha visto en el apartado anterior, en una primera aproximación al problema secuencial se ha pasado de $\mathbb{R}^{18}$ a $\mathbb{R}^{36}$. Este aumento en la dimensionalidad se pondrá aún más de manifiesto cuando se estudien determinados algoritmos, como los de ventanas deslizantes.

Resulta interesante pues, disponer de un número reducido s de parámetros más significativos, problema conocido como reducción de dimensionaldiad. Existen distintas aproximaciones a este problema (Y. Li, Dong, & Ma, 2008; Noda, Travieso, & Sánchez-Rodríguez, 2016; Sotoca & Pla, 2010; Whitney, 1971).

La norma MPEG-7 propone como técnicas de reducción la descomposición en valores singulares (SVD) (Golub & Van Loan, 1996) o el análisis de componentes independientes (ICA) (Hyvärinen, Hoyer, & Inki, 2001). La elección de la técnica para el proceso de reducción de dimensionalidad no es un tema menor, teniendo una envergadura similar a la de este trabajo, por lo que no será objeto de estudio en esta tesis. Siguiendo los planteamientos definidos en (Luque, Carrasco, Barbancho, & Romero, 2016), se han seleccionado los siguientes cinco parámetros:

- La razón de armonicidad.
- La dispersión de la potencia.
- El ancho de banda del primer formante.
- Tono *(pitch*).
- Frecuencia límite de armonicidad.

En Figura 5-10 se presentan los gráficos de los cinco parámetros seleccionados donde se muestran los puntos correspondientes a cada *frame* con el código de colores habituales, un valor central de los parámetros mediante un punto cuadrado y una medida de la dispersión de los mismos mediante un rectángulo. Para el valor central se usa la mediana y para la dispersión el rango intercuartil.



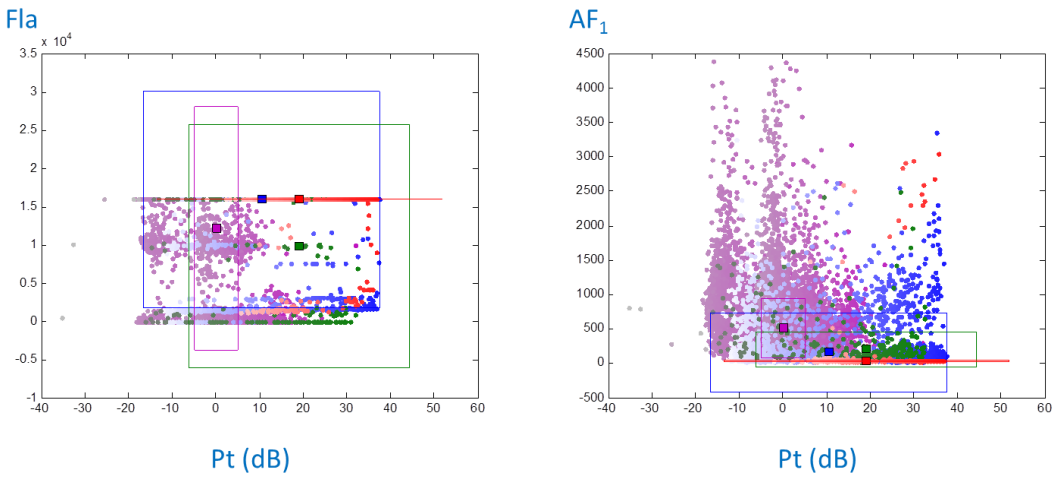


Figura 5-10 Proyecciones en $\mathbb{R}^2$ de las nubes de puntos (Pt – dispersión de potencia, Pitch – Tono, Ra – razón de amonicidad, Fla – frecuencia límite de armonicidad y AF_1- ancho de banda del primer formante)

A continuación, se compararán los resultados de la clasificación no secuencial con número reducido de parámetros $\mathbb{R}^5$, con los obtenidos inicialmente bajo $\mathbb{R}^{18}$, en el capítulo anterior.

En la Figura 5-11 se comparan los resultados de la clasificación mediante árbol de decisión sobre el mismo conjunto de sonidos en los dos espacios. El resultado global de la clasificación puede resumirse en la Figura 5-12.

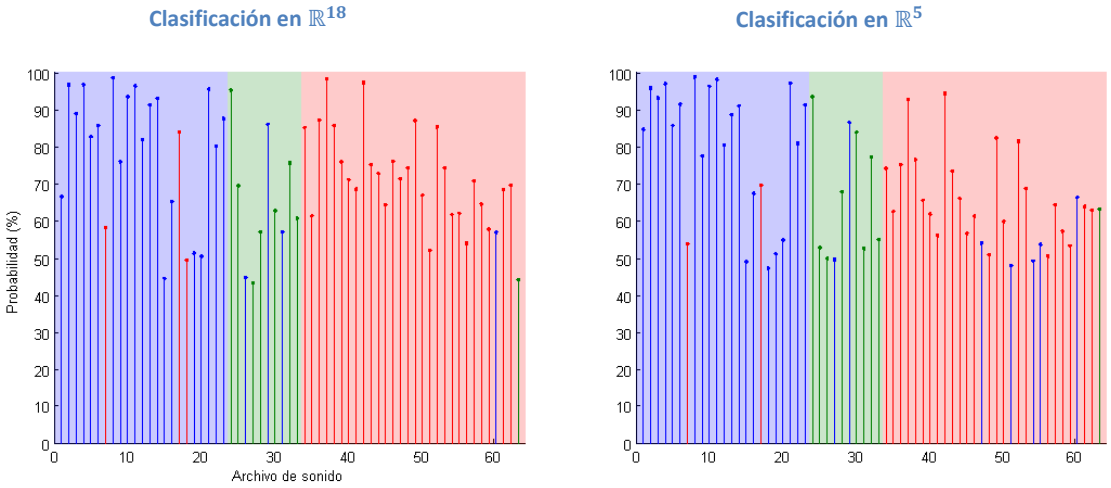


Figura 5-11 Clasificación por árbol de decisión

Como se puede observar, para la clasificación por árbol de decisión el paso de 18 a 5 parámetros incrementa la tasa de error del 12.70% hasta alcanzar un valor del 15.87%, poco más de tres puntos.

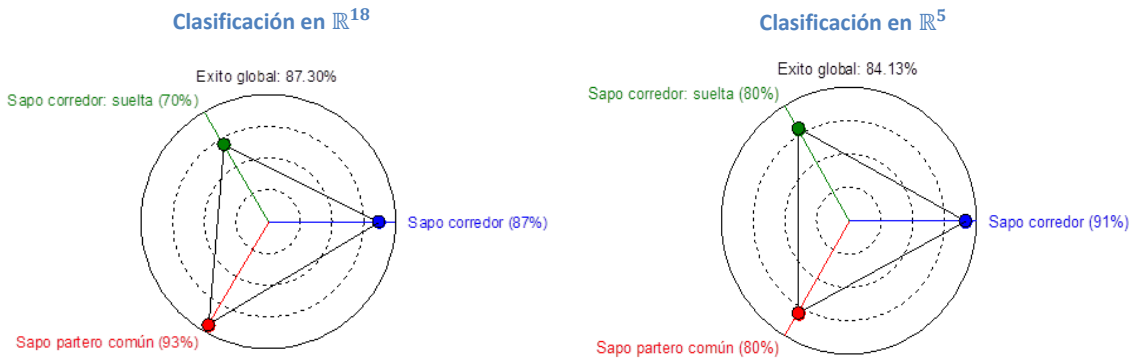


Figura 5-12 Resumen de la clasificación por árbol de decisión

En la Figura 5-13 se extiende esta comparación de resultados de clasificación entre ambos espacios para todos los métodos estudiados en la clasificación no secuencial. Como se ha venido trabajando, la altura de la barra refleja la tasa global de éxito para cada clasificador. Los puntos, con el código de colores habitual, indican la tasa de éxito para tipo de canto.

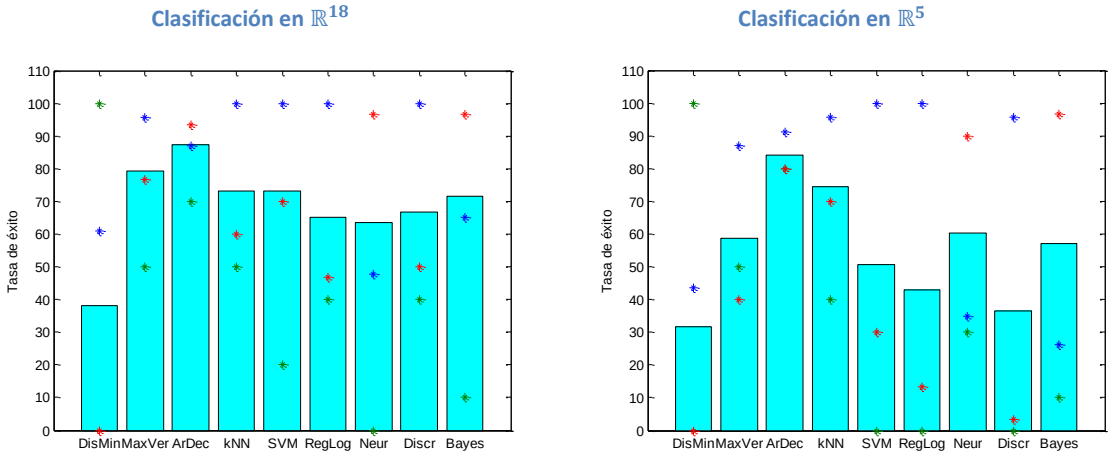


Figura 5-13 Resultados de la clasificación no secuencial

En la Figura 5-14 se comparan las figuras de mérito de cada técnica de clasificación, entre espacios, mediante un gráfico que enfrenta el rango de la tasa de error frente a la tasa de error.

La Tabla 5-3 recoge el factor de mérito de cada uno de los clasificadores utilizados. Se puede ver que el mejor método de clasificación sigue siendo el árbol de decisión.

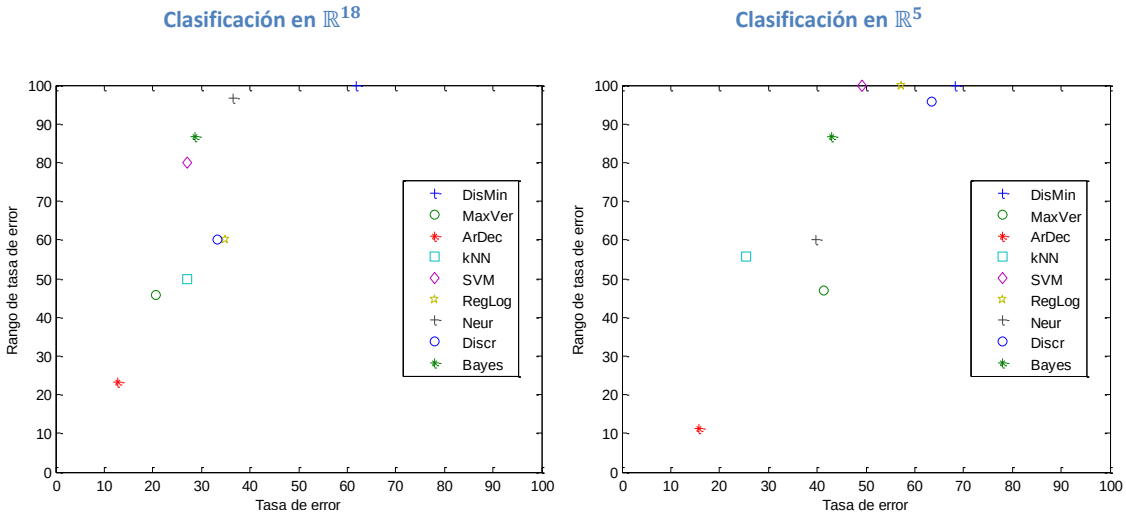


Figura 5-14 Tasa de error y su rango (unidades en %) para clasificación no secuencial

Algoritmo		Acierto	Errores	Rango	Distancia origen	Mérito
Distancia mínima	$\mathbb{R}^{18}$	38.10%	61.90%	100%	1.18	16.84%
	$\mathbb{R}^{5}$	31.75%	68.25%	100%	1.21	14.39%
Máxima verosimilitud	$\mathbb{R}^{18}$	79.37%	20.63%	43%	0.48	66.28%
	$\mathbb{R}^{5}$	58.73%	41.27%	47%	0.63	55.80%
Árboles de decisión	$\mathbb{R}^{18}$	87.30%	12.70%	23%	0.26	81.42%
	$\mathbb{R}^{5}$	84.13%	15.87%	11%	0.19	86.22%
k-vecinos más próximos	$\mathbb{R}^{18}$	73.02%	26.98%	50%	0.57	59.83%
	$\mathbb{R}^{5}$	74.60%	25.40%	56%	0.61	56.74%
SVM	$\mathbb{R}^{18}$	73.02%	26.98%	80%	0.84	40.30%
	$\mathbb{R}^{5}$	50.79%	49.21%	100%	1.11	21.19%
Regresión logística	$\mathbb{R}^{18}$	65.08%	34.92%	60%	0.69	50.91%
	$\mathbb{R}^{5}$	42.86%	57.14%	100%	1.15	18.56%
Redes neuronales	$\mathbb{R}^{18}$	63.49%	36.51%	97%	1.04	26.71%
	$\mathbb{R}^{5}$	60.32%	39.68%	60%	0.72	49.13%
Función discriminante	$\mathbb{R}^{18}$	66.67%	33.33%	60%	0.69	51.47%
	$\mathbb{R}^{5}$	36.51%	63.49%	96%	1.15	18.82%
Clasificador bayesiano	$\mathbb{R}^{18}$	71.43%	28.57%	87%	0.92	35.25%
	$\mathbb{R}^{5}$	57.14%	42.86%	87%	0.97	31.63%

Tabla 5-3. Factor de mérito de clasificadores no secuencial

La Tabla 5-4 compara los valores de los indicadores de exactitud, precisión, sensibilidad, especificidad y tasa de errores.

La Figura 5-15 se representa el análisis ROC de los distintos algoritmos estudiados donde de nuevo, en $\mathbb{R}^5$, el mejor clasificador es el basado en árbol de decisión.

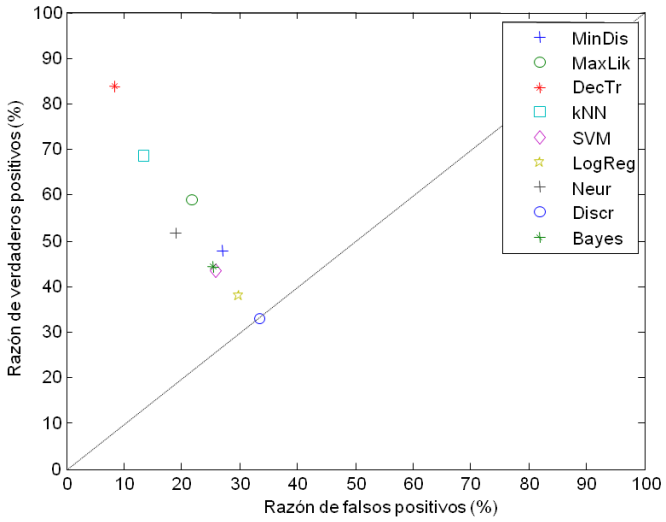


Figura 5-15 Comparación de los métodos de clasificación en $\mathbb{R}^5$ mediantes análisis ROC

Algoritmo		Exactitud	Tasa de errores	Precisión	Sensib.	Especif.
Distancia mínima	$\mathbb{R}^{18}$	58.73%	41.27%	-	53.62%	75.47%
	$\mathbb{R}^5$	54.50%	45.50%	-	47.83%	72.96%
Máxima verosimilitud	$\mathbb{R}^{18}$	86.24%	13.76%	79.37%	74.11%	89.58%
	$\mathbb{R}^5$	72.49%	27.51%	66.74%	58.99%	78.21%
Árboles de decisión	$\mathbb{R}^{18}$	91.53%	8.47%	87.05%	83.43%	93.01%
	$\mathbb{R}^5$	89.42%	10.58%	85.40%	83.77%	91.52%
k-vecinos más próximos	$\mathbb{R}^{18}$	82.01%	17.99%	85.83%	70.00%	85.83%
	$\mathbb{R}^5$	83.07%	16.93%	84.97%	68.55%	86.49%
SVM	$\mathbb{R}^{18}$	82.01%	17.99%	75.21%	63.33%	86.04%
	$\mathbb{R}^5$	67.20%	32.80%	-	43.33%	74.17%
Regresión logística	$\mathbb{R}^{18}$	76.72%	23.28%	77.42%	62.22%	81.87%
	$\mathbb{R}^5$	61.90%	38.10%	46.55%	37.78%	70.20%
Redes neuronales	$\mathbb{R}^{18}$	75.66%	24.34%	54.53%	49.18%	81.14%
	$\mathbb{R}^5$	73.54%	26.46%	60.61%	51.59%	81.02%
Función discriminante	$\mathbb{R}^{18}$	77.78%	22.22%	77.83%	63.33%	82.70%
	$\mathbb{R}^5$	57.67%	42.33%	-	33.00%	66.49%
Clasificador bayesiano	$\mathbb{R}^{18}$	80.95%	19.05%	63.63%	57.29%	83.79%
	$\mathbb{R}^5$	71.43%	28.57%	66.85%	44.25%	74.49%

Tabla 5-4. Indicadores para la evaluación de clasificadores en $\mathbb{R}^5$

En la Figura 5-16 se representa la concordancia de los resultados usando los coeficientes kappa.

La conclusión tras analizar las distintas comparativas realizadas, es que la clasificación no secuencial en $\mathbb{R}^5$ por árbol de decisión ofrece los mejores resultados, no alejándose de los resultados obtenidos en $\mathbb{R}^{18}$.

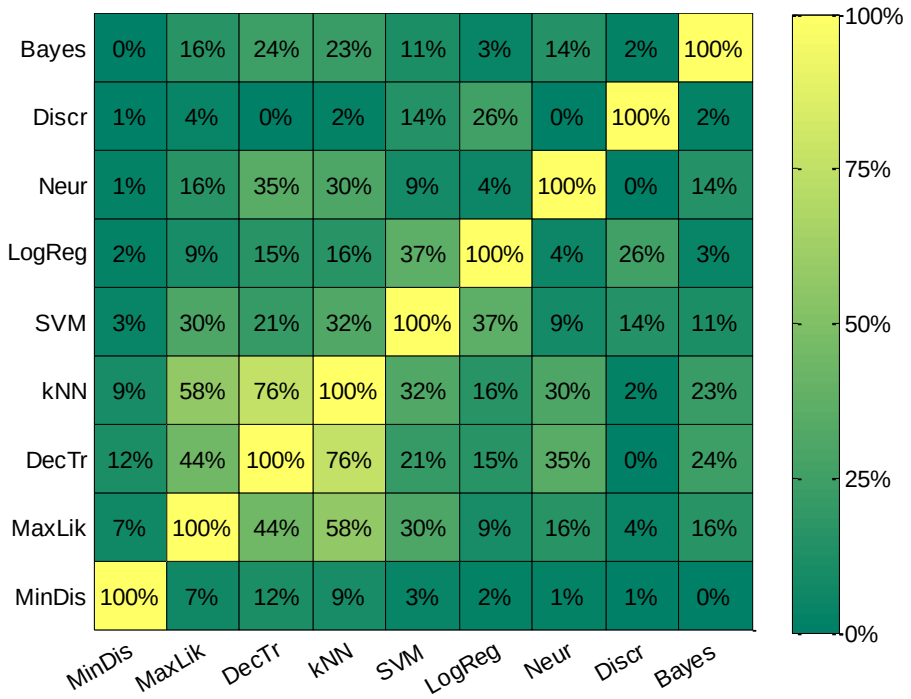


Figura 5-16 Comparación de los métodos de clasificación en $\mathbb{R}^5$ mediantes coeficientes kappa de Cohen

5.4. Ventanas deslizantes

Una de las técnicas básicas de clasificación de datos secuenciales es la basada en ventanas deslizantes (Aggarwal, 2007; Dietterich, 2002). De forma genérica en un espacio $\mathbb{R}^n$ sean: $\boldsymbol{x}_i$ es el vector de n parámetros con el que se identifica el *frame* i-ésimo, C el algoritmo de clasificación utilizado y $\hat{y}_i$ la clase estimada a la que pertenece dicho *frame*. En una clasificación no secuencial se tiene que

$$\hat{y}_i = C(\boldsymbol{x}_i) \ . \tag{5.1}$$

La técnica de clasificación secuencial mediante ventanas deslizantes se basa en utilizar los mismos clasificadores que en el caso no secuencial, pero aplicándolos sobre otro conjunto de datos. Para esto se define una "ventana" de tamaño w que comience en el *frame* i-ésimo y comprenda w *frames* hacia atrás. En este caso, la clasificación estimada se obtiene mediante la expresión

$$\hat{y}_i = C(\boldsymbol{x_i}, \boldsymbol{x_{i-1}}, \cdots, \boldsymbol{x_{i-w+1}}) = C(\boldsymbol{X_i}) \, , \tag{5.2}$$

siendo $\boldsymbol{X_i} \equiv (\boldsymbol{x_i}, \boldsymbol{x_{i-1}}, \cdots, \boldsymbol{x_{i-w+1}})$. La denominación de "ventana deslizante" deriva del hecho de que la "ventana" se va "deslizando" hacia adelante a medida que se van considerando *frames* consecutivos.

Si la dimensión de $\boldsymbol{x_i}$ es n ($\boldsymbol{x_i} \in \mathbb{R}^n$), entonces la dimensión de $\boldsymbol{X_i}$ es $n \cdot w$ ($\boldsymbol{X_i} \in \mathbb{R}^{n \cdot w}$), poniendo de manifiesto, de forma aún más clara que en el apartado 5.2, la importancia de reducir la dimensionalidad de cada *frame*.

Utilizando los mismos algoritmos que en la clasificación no secuencial del capítulo anterior, y definiendo cada *frame* mediante 5 parámetros ($\boldsymbol{x_i} \in \mathbb{R}^5$), la Figura 5-17 refleja el factor de mérito obtenido por cada clasificador para distintos tamaños de la ventana.

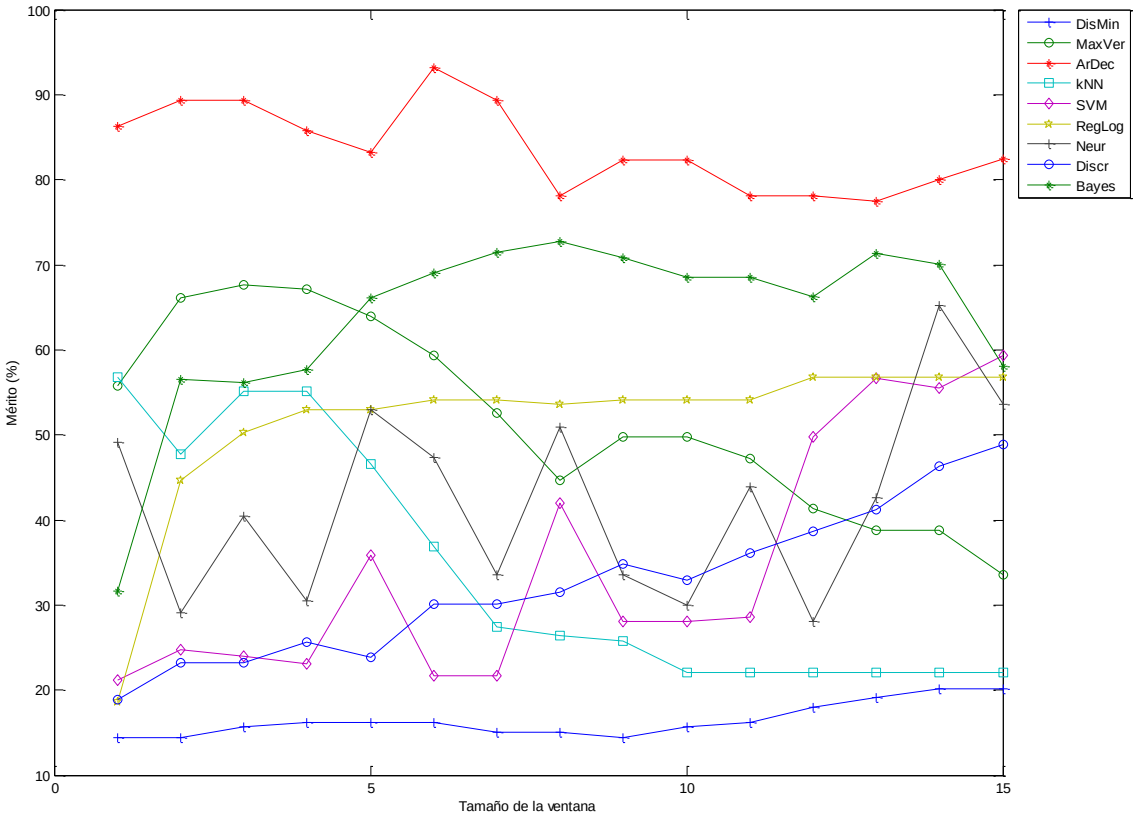


Figura 5-17 Factor de mérito en función del tamaño de la ventana

Como se puede ver el comportamiento es diverso para cada uno de los algoritmos ensayados. La Figura 5-18 refleja el comportamiento del factor de mérito promedio, así como el valor máximo y el mínimo.

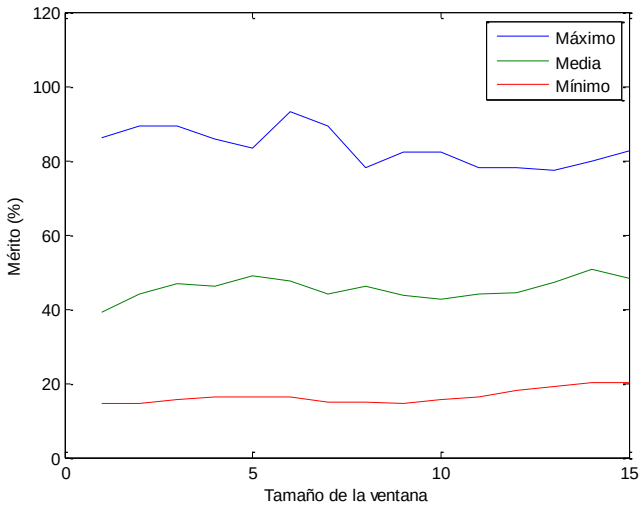


Figura 5-18 Factor de mérito en función del tamaño de la ventana (media)

Se observa que el valor medio (en verde) tiene un máximo relativo para un tamaño de la ventana de 5 *frames*, mientras que el valor máximo (en azul) alcanza el máximo para 6 *frames* (casi el mismo valor). A efectos de comparación de los distintos algoritmos, en lo que sigue se utilizará un tamaño de ventana de 5 *frames*.

Una vez fijado el tamaño de la ventana y de forma análoga al capítulo anterior, a continuación se realizan distintos métodos de evaluación de los clasificadores con el objetivo de determinar cuál es el algoritmo que mejores resultado obtiene.

La Figura 5-19 refleja de forma gráfica la comparación de los distintos algoritmos usando ventanas deslizantes de tamaño 5 *frames* en $\mathbb{R}^5$. Los distintos algoritmos, usados en el estudio no secuencial, son aplicados sobre cada ventana. Finalmente por conteo se elige la clase asignada. La altura de la barra refleja la tasa global de éxito para cada clasificador. Los puntos, con el código de colores habitual, indican la tasa de éxito para cada tipo de canto.

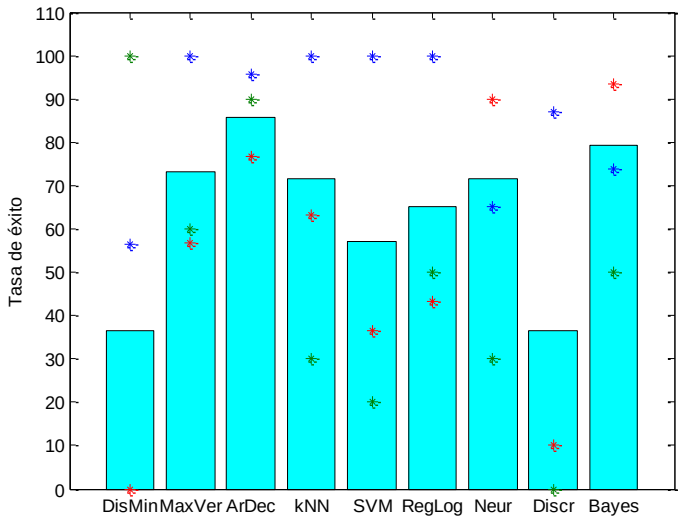


Figura 5-19 Resultados de la clasificación con ventana deslizante (tamaño ventana: 5)

La Figura 5-20 refleja el mérito de cada técnica de clasificación mediante un gráfico que enfrenta el rango de la tasa de error y la tasa de error.

La Tabla 5-5 recoge el factor de mérito de cada uno de los clasificadores utilizados. La Tabla 5-6 recoge los valores de los indicadores de exactitud, precisión, sensibilidad, especificidad y tasa de errores. Se puede ver que el mejor método de clasificación sigue siendo el árbol de decisión.

En la Figura 5-21 se representa el análisis ROC de los distintos algoritmos estudiados donde de nuevo el mejor clasificador es el árbol de decisión.

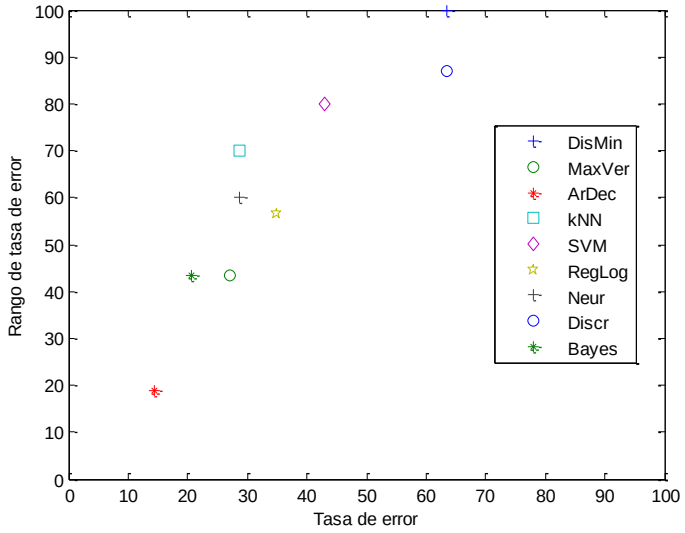


Figura 5-20 Tasa de error (%) y su rango (%) para clasificación con ventana deslizante (tamaño ventana: 5)

Algoritmo	Acierto	Errores	Rango	Distancia origen	Mérito
Distancia mínima	36.51%	63.49%	100%	1.18	16.24%
Máxima verosimilitud	73.02%	26.98%	43%	0.51	63.90%
Árboles de decisión	85.71%	14.29%	19%	0.24	83.20%
k-vecinos más próximos	71.43%	28.57%	70%	0.76	46.54%
SVM	57.14%	42.86%	80%	0.91	35.83%
Regresión logística	65.08%	34.92%	57%	0.67	52.93%
Redes neuronales	71.43%	28.57%	0%	0.66	53.01%
Función discriminante	36.51%	63.49%	87%	1.08	23.87%
Clasificador bayesiano	79.37%	20.63%	43%	0.48	66.06%

Tabla 5-5. Factor de mérito para clasificación con ventana deslizante (tamaño ventana: 5)

Algoritmo	Exactitud	Tasa de errores	Precisión	Sensib.	Especif.
Distancia mínima	57.67%	42.33%	-	52.17%	74.84%
Máxima verosimilitud	82.01%	17.99%	85.83%	72.22%	85.83%
Árboles de decisión	90.48%	9.52%	87.23%	87.44%	92.53%
k-vecinos más próximos	80.95%	19.05%	85.37%	64.44%	85.00%
SVM	71.43%	28.57%	71.20%	52.22%	77.70%
Regresión logística	76.72%	23.28%	78.54%	64.44%	81.87%
Redes neuronales	80.95%	19.05%	65.60%	61.74%	84.08%
Función discriminante	57.67%	42.33%	-	32.32%	66.14%
Clasificador bayesiano	86.24%	13.76%	79.67%	72.42%	88.49%

Tabla 5-6. Indicadores para la evaluación de clasificadores con ventana deslizante (tamaño ventana: 5)

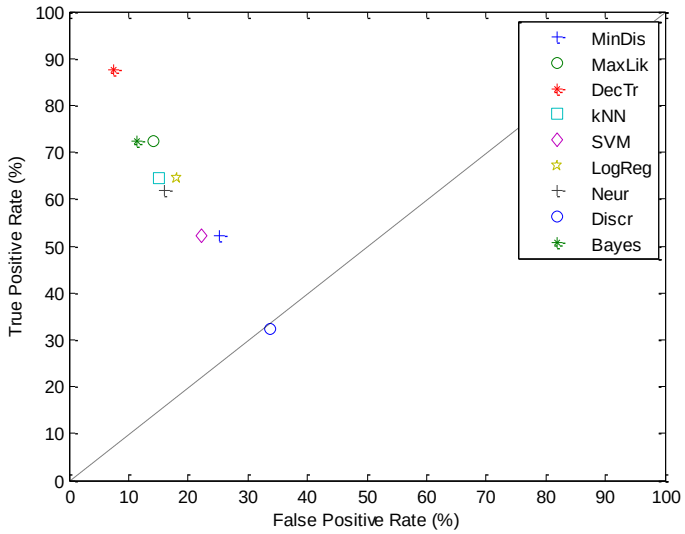


Figura 5-21 Comparación de los métodos de clasificación con ventana deslizante (tamaño ventana: 5) mediantes análisis ROC

En la Figura 5-22 se representa la concordancia de los resultados usando los coeficientes kappa.

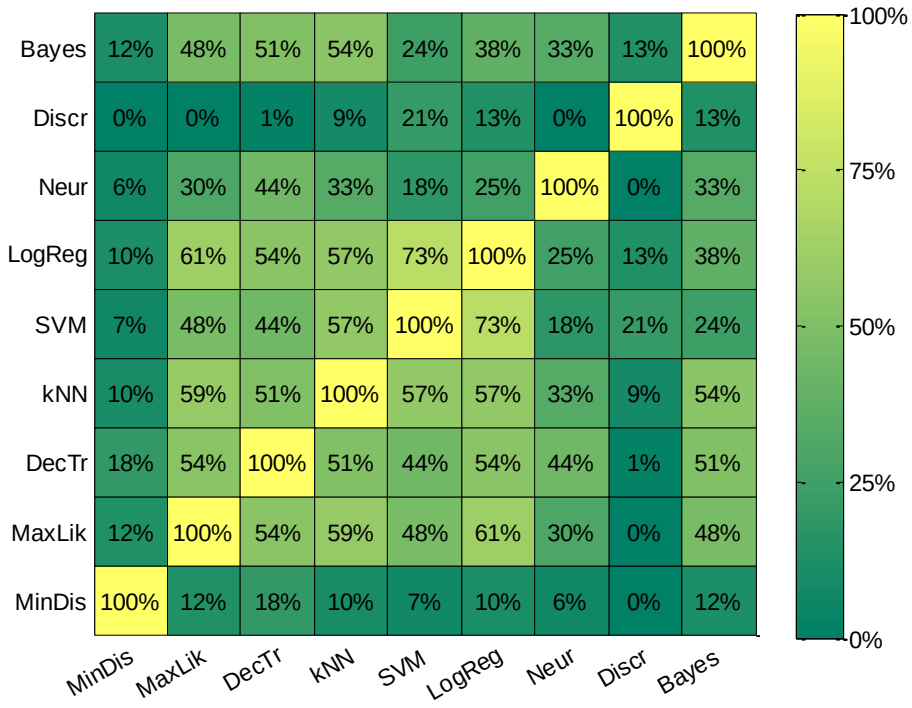


Figura 5-22 Comparación de los métodos de clasificación con ventana deslizante (tamaño ventana: 5) mediantes coeficientes kappa de Cohen

La conclusión tras analizar las distintas comparativas realizadas, es que la clasificación por árbol de decisión ofrece los mejores resultados cuando se usan ventanas deslizantes de tamaño 5.

5.5. Ventanas deslizantes recursivas

Una variante de la técnica anterior es la de ventanas deslizantes recursivas (Dietterich, 2002). En ella la clasificación de un *frame* tiene en cuenta, no sólo los parámetros en *frames* anteriores, sino también la clasificación estimada de dichos *frames*. De forma genérica en un espacio $\mathbb{R}^n$ sean: $\boldsymbol{x_i}$ el vector de parámetros (en $\mathbb{R}^n$) con el que se identifica el *frame* i-ésimo, C el algoritmo de clasificación utilizado, $\hat{y}_i$ la clase estimada a la que pertenece dicho *frame* y w el tamaño de la ventana que comienza en el *frame* i-ésimo y comprende w *frames* hacia atrás. En este algoritmo, la clasificación estimada se obtiene mediante la expresión

$$\hat{y}_i = C(\boldsymbol{x_i}, \boldsymbol{x_{i-1}}, \cdots, \boldsymbol{x_{i-w+1}}, \hat{y}_{i-1}, \hat{y}_{i-2}, \cdots, \hat{y}_{i-w}) = C(\boldsymbol{X_i}, \boldsymbol{\hat{y}_{i-1}}) \ , \qquad \textbf{(5.3)}$$

siendo

$$\boldsymbol{X_i} \equiv (\boldsymbol{x_i}, \boldsymbol{x_{i-1}}, \cdots, \boldsymbol{x_{i-w+1}}) \text{ e } \boldsymbol{\hat{y}_{i-1}} \equiv (\hat{y}_{i-1}, \hat{y}_{i-2}, \cdots, \hat{y}_{i-w}). \qquad \textbf{(5.4)}$$

Si la dimensión de $\boldsymbol{x_i}$ es n ($\boldsymbol{x_i} \in \mathbb{R}^n$), entonces la dimensión de $\boldsymbol{X_i}$ es $n \cdot w$ ($\boldsymbol{X_i} \in \mathbb{R}^{n \cdot w}$) e $\boldsymbol{\hat{y}_{i-1}}$ es de dimensión w ($\boldsymbol{\hat{y}_{i-1}} \in \mathbb{R}^w$). Por tanto, el predictor $(\boldsymbol{X_i}, \boldsymbol{\hat{y}_{i-1}})$ es de dimensión $(n+1) \cdot w$.

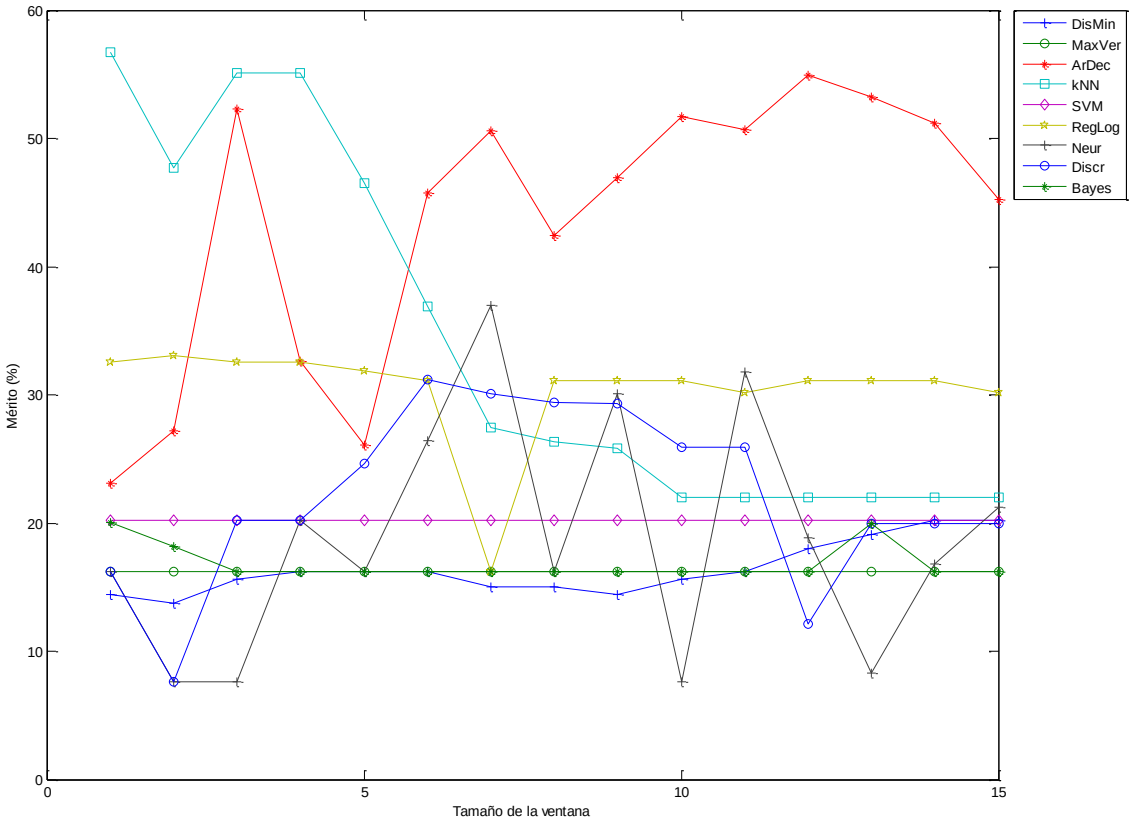


Figura 5-23 Factor de mérito en función del tamaño de la ventana

Utilizando los mismos algoritmos que en la clasificación no secuencial del capítulo anterior, y definiendo cada *frame* mediante 5 parámetros ($\boldsymbol{x_i} \in \mathbb{R}^5$), la Figura 5-23

refleja el factor de mérito obtenido por cada clasificador para distintos tamaños de la ventana.

Como se puede ver el comportamiento es diverso para cada uno de los algoritmos estudiados. La Figura 5-24 refleja el comportamiento del factor de mérito promedio, así como el valor máximo y mínimo.

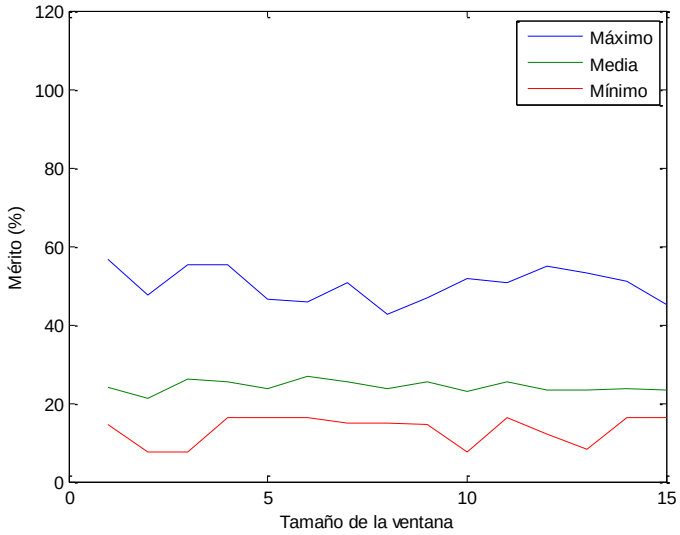


Figura 5-24 Factor de mérito en función del tamaño de la ventana (media)

En este caso, no es fácil realizar una elección del tamaño de ventana. Por ello, y para facilitar la comparación con el caso anterior, se utilizará el mismo tamaño de ventana que el apartado anterior, 5 *frames*.

La Figura 5-25 refleja de forma gráfica la comparación de los distintos algoritmos. La altura de la barra refleja la tasa global de éxito para cada clasificador. Los puntos, con el código de colores habitual, indican la tasa de éxito para cada tipo de canto.

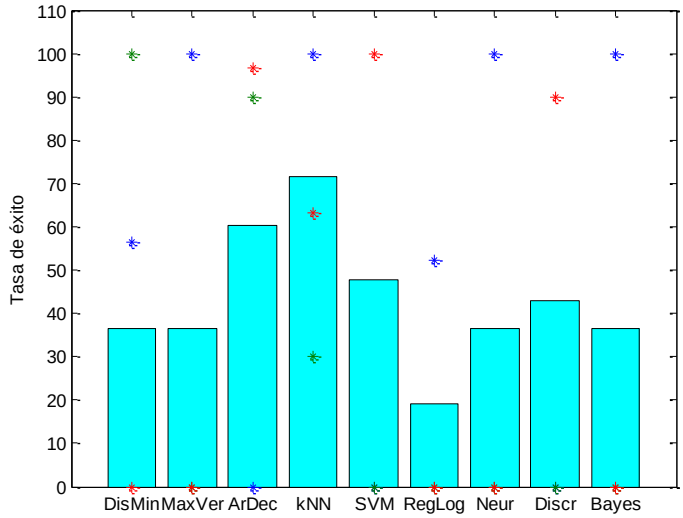


Figura 5-25 Resultados de la clasificación con ventana deslizante recursiva (tamaño ventana: 5)

La Figura 5-26 refleja el mérito de cada técnica de clasificación enfrentando el rango de la tasa de error frente la tasa de error.

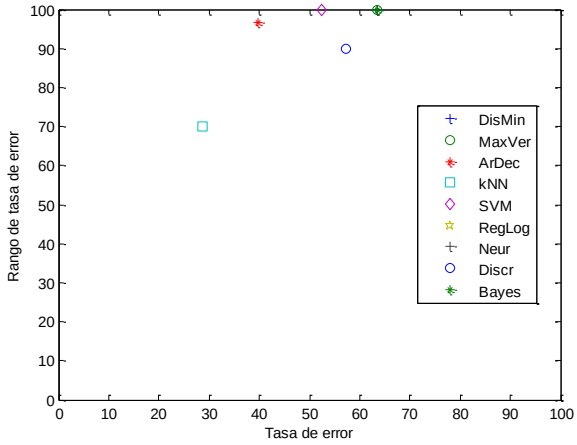


Figura 5-26 Tasa de error y su rango para clasificación con ventana deslizante recursiva (tamaño ventana: 5)

La Tabla 5-7 recoge el factor de mérito de cada uno de los clasificadores utilizados. La Tabla 5-8 recoge los valores de los indicadores de exactitud, precisión, sensibilidad, especificidad y tasa de errores.

Algoritmo	Acierto (%)	Errores (%)	Rango (%)	Distancia origen	Mérito (%)
Distancia mínima	36.51%	63.49%	100%	1.18	16.24%
Máxima verosimilitud	36.51%	63.49%	100%	1.18	16.24%
Árboles de decisión	60.32%	39.68%	97%	1.04	26.11%
k-vecinos más próximos	71.43%	28.57%	70%	0.76	46.54%
SVM	47.62%	52.38%	100%	1.13	20.18%
Regresión logística	19.05%	80.95%	52%	0.96	31.90%
Redes neuronales	36.51%	63.49%	100%	1.18	16.24%
Función discriminante	42.86%	57.14%	90%	1.07	24.62%
Clasificador bayesiano	36.51%	63.49%	100%	1.18	16.24%

Tabla 5-7. Factor de mérito para clasificación con ventana deslizante recursiva (tamaño ventana: 5)

Algoritmo	Exactitud (%)	Tasa de errores (%)	Precisión (%)	Sensib. (%)	Especif. (%)
Distancia mínima	57.67%	42.33%	-	52.17%	74.84%
Máxima verosimilitud	57.67%	42.33%	-	33.33%	66.67%
Árboles de decisión	73.54%	26.46v	51.57%	62.22%	74.92%
k-vecinos más próximos	80.95%	19.05%	85.37%	64.44%	85.00%
SVM	65.08%	34.92%	-	33.33%	66.67%
Regresión logística	46.03%	53.97%	-	17.39%	55.56%
Redes neuronales	57.67%	42.33%	-	33.33%	66.67%
Función discriminante	61.90%	38.10%	15.25%	30.00%	64.96%
Clasificador bayesiano	57.67%	42.33%	-	33.33%	66.67%

Tabla 5-8. Indicadores para la evaluación de clasificadores con ventana deslizante recursiva (tamaño ventana: 5)

En la Figura 5-27 se representa el análisis ROC de los distintos algoritmos estudiados. Y en la Figura 5-28 se representa la concordancia de los resultados usando los coeficientes kappa.

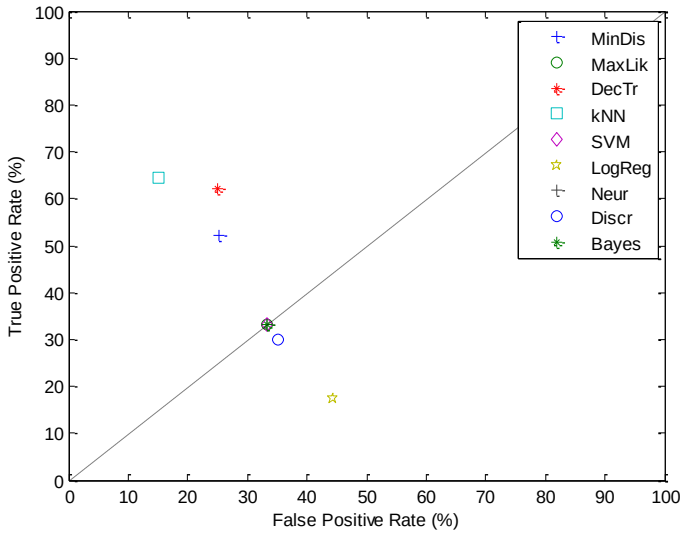


Figura 5-27 Comparación de los métodos de clasificación con ventana deslizante recursiva (tamaño ventana: 5) mediantes análisis ROC

Figura 5-28 Comparación de los métodos de clasificación con ventana deslizante recursiva (tamaño ventana: 5) mediantes coeficientes kappa de Cohen

La conclusión tras analizar las distintas comparativas realizadas, es que la clasificación usando los k-vecinos más próximos ofrece los mejores resultados cuando se usan ventanas recursivas de tamaño 5.

5.6. Modelos ocultos de Markov

5.6.1. Descripción general

Una de las técnicas más utilizadas en el reconocimiento de sonidos (L. R. Rabiner, 1989; L. R. Rabiner & Juang, 1986; Stamp, 2012) es la basada en los modelos ocultos de Markov (HMM: *Hidden Markov Models*), que tienen un enfoque estadístico. Este es el método de clasificación de sonidos sugerido en la norma MPEG-7 (apartado 6.4 de ISO, 2001) y muy utilizado en el reconocimiento de sonidos junto con los MFCC.

Según los modelos ocultos de Markov, el sonido "observado" O_i en el instante (*frame*) i-ésimo es consecuencia (es emitido) por un proceso subyacente "oculto" que en ese instante se encuentra en el estado Q_i, como se representa en la Figura 5-29.

El conjunto de posibles estados S está formado por un número finito de estados (n), es decir, $S = \{S_1, S_2, \cdots, S_n\}$. Naturalmente el estado Q_t en el instante (*frame*) t-ésimo es uno de los posibles estados, por tanto, $Q_t \in S$. Por otro lado, el conjunto de posibles observaciones C está formado por un número finito de códigos (m), es decir, $C = \{C_1, C_2, \cdots, C_m\}$. De forma análoga, la observación O_t en el instante (*frame*) t-ésimo es uno de los posibles códigos, por tanto, $O_t \in C$.

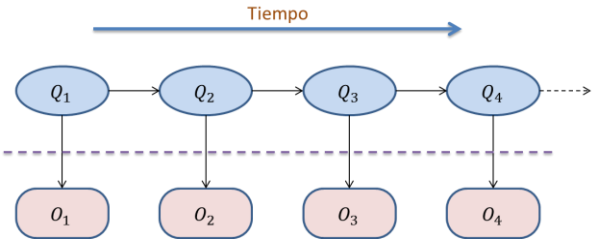


Figura 5-29 Estados y observaciones en un modelo oculto de Markov

Los elementos que componen los modelos ocultos de Markov son:

- Un conjunto S de n estados. Estos estados se encuentran ocultos.
- Una matriz de transiciones de probabilidades $\boldsymbol{T}$, en la que cada elemento t_{ij} representa la probabilidad de que se produzca una transición del estado S_i al S_j.
- Una matriz de emisiones $\boldsymbol{E}$, formado por n vectores E_i, cada uno de los cuales está constituido por m elementos e_{ik} cada uno de los cuales representa la probabilidad de que estando en el estado S_i se emita un código C_k.
- Una matriz $\boldsymbol{P}$ en la que cada elemento genérico p_i representa la probabilidad de que estado i-ésimo sea el estado inicial.

Establecidos los valores apropiados para estos componentes se puede usar el HMM para generar una secuencia de observaciones $O \equiv O_1, O_2, \cdots, O_T$. El procedimiento de reconocimiento puede ser descrito como el cálculo de la probabilidad de que una

observación O sea producida por alguno de los modelos de referencia creados en la fase de entrenamiento, a fin de encontrar la que proporciona el valor máximo.

La Figura 5-30 muestra un ejemplo de un modelo oculto de Markov con 4 estados.

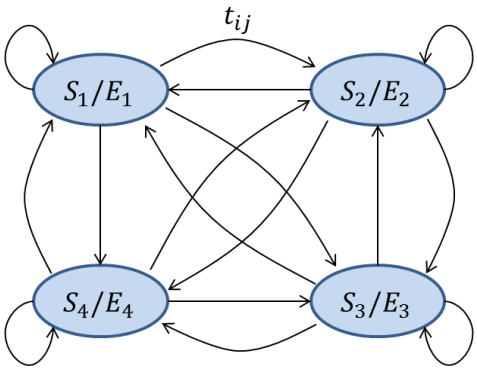


Figura 5-30 Transiciones y emisiones en un modelo oculto de Markov

Como se ha visto, en un modelo oculto de Markov las observaciones O_t pertenecen al conjunto C de posibles observaciones que está formado por un número finito (m) de códigos. Para el caso de estudio, en el instante (*frame*) t-ésimo se dispone de un valor $\boldsymbol{X_t}$ con el valor de los parámetros para ese *frame*, es decir, $\boldsymbol{X_t} \in \mathbb{R}^5$ (o $\boldsymbol{X_t} \in \mathbb{R}^{18}$ si no se ha efectuado ningún proceso de reducción de dimensionalidad). Para convertir este valor de $\boldsymbol{X_t}$ en una observación O_t perteneciente a un conjunto finito de códigos se pueden usar 2 estrategias.

La primera de las estrategias consiste en construir un modelo oculto de Markov para cada uno de los parámetros del *frame*. Estos modelos no son independiente sino que están acoplados (Brand, 1997; Zhong & Ghosh, 2002). Si p es el número de parámetros de un *frame*, se tiene que $\boldsymbol{X_t} = \{x_{t1}, x_{t2}, \cdots, x_{tp}\}$ y, por tanto, el número de modelos ocultos de Markov que se tendrá que construir es p. Para cada uno de esos modelos, por ejemplo el u-ésimo, el problema ahora es pasar el valor del parámetro x_{tu} a la observación O_{tu}, perteneciente al conjunto finito de código. Esto puedo hacerse fácilmente con una simple cuantización de x_{tu} en tantos niveles como códigos se deseen.

La segunda estrategia para convertir $\boldsymbol{X_t}$ en una observación O_t perteneciente a un conjunto finito de códigos es la cuantización multidimensional, utilizando para ellos algunos de los algoritmos disponibles (Linde, Buzo, & Gray, 1980). Con este enfoque sólo es necesario construir un modelo oculto de Markov. Por su sencillez está será la estrategia que se adaptara, utilizando para ello la función, en código MATLAB, *kmeanlbg* (Brookes, 1998b).

La selección del tipo de modelo, el número de estado y código no es simple. Estas elecciones deben realizarse en función de la señal que se está modelando. En (L. R.

Rabiner, 1989) se sugiere un número de estados entre 2 y 10 para el procesado de sonidos y 256 códigos. Para el caso de estudio se ha seleccionado 5 estados. Entrenando el modelo con los patrones de cada clase de sonido, se obtienen tantos modelos como clases, que resultan tener la misma estructura reflejada en la Figura 5-31, que corresponde con una topología conocida como de "izquierda a derecha" (Modelo HMM de Bakis). Esta topología hace que la matriz de transiciones sea triangular superior (5.5). El entrenamiento se realiza con la función *hmmtrain* de MATLAB.

$$a_{ij} = 0 \quad \forall i > j \tag{5.5}$$

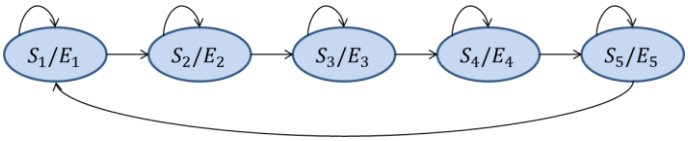


Figura 5-31 Modelo oculto de Markov para el canto de un anuro

La etapa final consiste en considerar una secuencia de *frames* que se desea clasificar, para cada uno de los cuales se conocer el vector de parámetros $\boldsymbol{X_t}$, y transformarlo en una secuencia de observaciones O_t mediante la función disteusq (Brookes, 1998a). A continuación, se calcula mediante un procedimiento de avance y retroceso (Baum & Eagon, 1967), la posibilidad de que dicha secuencia de observaciones haya sido generada por los modelos ocultos correspondientes a cada una de las clases. Esta operación se realiza mediante la función *hmmdecode* de MATLAB, clasificando la secuencia como la de mayor probabilidad.

En este punto, se han utilizado tres enfoques sobre la forma de obtener una secuencia de *frames*:

a. Usando una ventana deslizante de tamaño w.
b. Usando una única secuencia formada por todos los *frames* del sonido.
c. Usando secuencias consecutivas (sin deslizamiento) cuyo tamaño coincide con la duración media de las regiones de interés de los sonidos patrón.

Estos tres enfoques se desarrollan en los siguientes apartados, obteniendo resultados distintos para cada uno de ellos.

5.6.2. Modelos ocultos de Markov sobre ventanas deslizantes

El primer enfoque para obtener una secuencia de *frames* es usar una ventana deslizante de tamaño w (Tiberiu, 2013). La Figura 5-32 muestra el resultado de la clasificación mediante modelos ocultos de Markov aplicados a secuencias de sonidos con técnica de ventana deslizante. Se han utilizado 5 parámetros para definir cada *frame*, y una ventana deslizante de tamaño 5. El resultado global de la clasificación puede resumirse en la Figura 5-33.

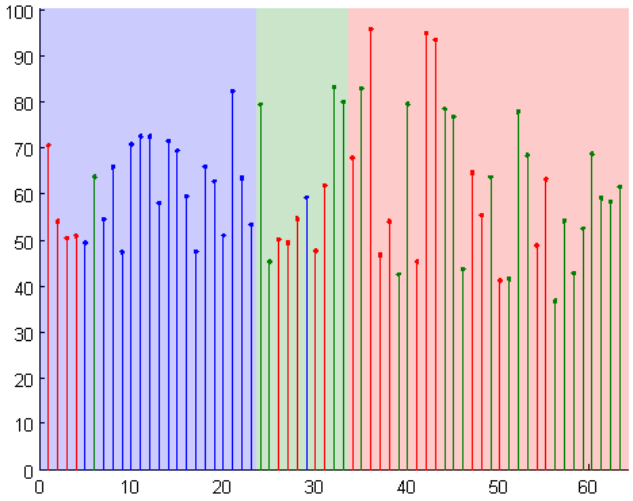


Figura 5-32 Clasificación por modelo oculto de Markov sobre ventanas deslizantes

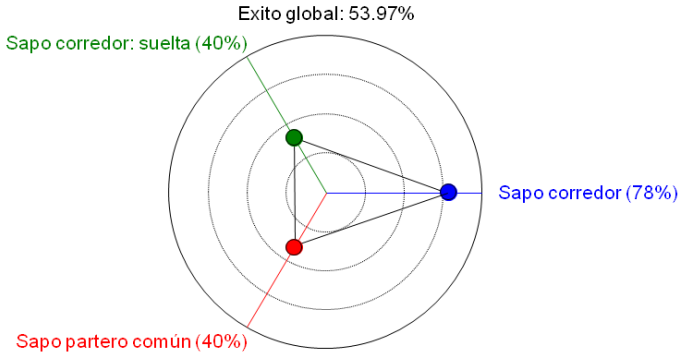


Figura 5-33 Resumen de la clasificación por modelo oculto de Markov sobre ventanas deslizantes

La Figura 5-34 refleja de forma gráfica la comparación de los distintos algoritmos de ventana deslizante incluyendo el de modelos ocultos de Markov. La altura de la barra refleja la tasa global de éxito para cada clasificador. Los puntos, con el código de colores habitual, indican la tasa de éxito para cada tipo de canto.

La Figura 5-35 refleja el mérito de cada técnica de clasificación enfrentando el rango de la tasa de error y la tasa de error.

La Tabla 5-9 recoge el factor de mérito de cada uno de los clasificadores utilizados con el método de ventana deslizante.

La Tabla 5-10 recoge los valores de los indicadores de exactitud, precisión, sensibilidad, especificidad y tasa de errores. Se puede ver que el mejor método de clasificación sigue siendo el árbol de decisión.

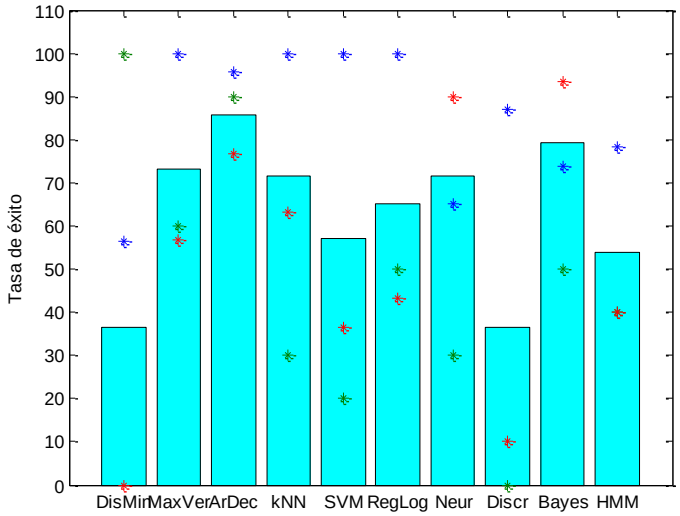


Figura 5-34 Resultados de la clasificación con ventanas deslizantes y HMM

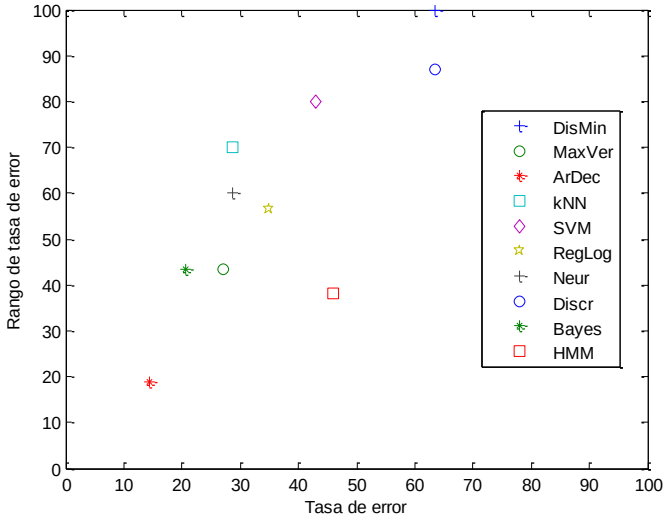


Figura 5-35 Tasa de error y su rango (unidades en %) para clasificación con ventanas deslizantes y HMM

Algoritmo	Acierto	Errores	Rango	Distancia origen	Mérito
Distnacia mínima	36.51%	63.49%	100%	1.18	16.24%
Máxima verosimilitud	73.02%	26.98%	43%	0.51	63.90%
Árboles de decisión	85.71%	14.29%	19%	0.24	83.20%
k-vecinos más próximos	71.43%	28.57%	70%	0.76	46.54%
SVM	57.14%	42.86%	80%	0.91	35.83%
Regresión logística	65.08%	34.92%	57%	0.67	52.93%
Redes neuronales	71.43%	28.57%	60%	0.66	53.01%
Función discriminante	36.51%	63.49%	87%	1.08	23.87%
Clasificador bayesiano	79.37%	20.63%	43%	0.48	66.06%
Modelo oculto de Markov	53.97%	46.03%	38%	0.60	57.67%

Tabla 5-9. Factor de mérito para clasificación con ventanas deslizantes y HMM

Algoritmo	Exactitud	Tasa de errores	Precisión	Sensib.	Especif.
Distancia mínima	57.67%	42.33%	-	52.17%	74.84%
Máxima verosimilitud	82.01%	17.99%	85.83%	72.22%	85.83%
Árboles de decisión	90.48%	9.52%	87.23%	87.44%	92.53%
k-vecinos más próximos	80.95%	19.05%	85.37%	64.44%	85.00%
SVM	71.43%	28.57%	71.20%	52.22%	77.70%
Regresión logística	76.72%	23.28%	78.54%	64.44%	81.87%
Redes neuronales	80.95%	19.05%	65.60%	61.74%	84.08%
Función discriminante	57.67%	42.33%	-	32.32%	66.14%
Clasificador bayesiano	86.24%	13.76%	79.67%	72.42%	88.49%
Modelo oculto de Markov	69.31%	30.69%	56.42%	52.75%	78.13%

Tabla 5-10. Indicadores para la evaluación de clasificadores con ventana deslizante (tamaño ventana: 5)

En la Figura 5-36 se representa el análisis ROC de los distintos algoritmos estudiados donde de nuevo el mejor clasificador es el árbol de decisión.

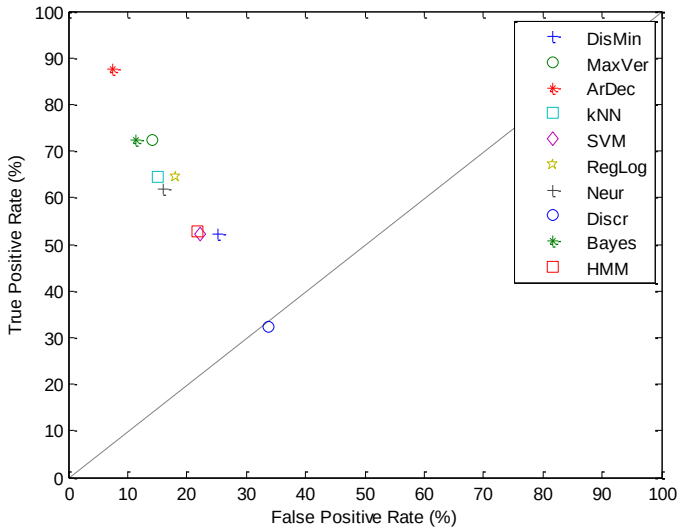


Figura 5-36 Comparación de los métodos de clasificación con ventana deslizante (tamaño ventana: 5) mediantes análisis ROC

En la Figura 5-37 se representa la concordancia de los resultados usando los coeficientes kappa.

La conclusión, en $\mathbb{R}^5$ tras analizar las distintas comparativas realizadas, es que la clasificación por HMM genera peores resultados de forma generalizadas respecto a la mayoría de métodos sobre ventana deslizante.

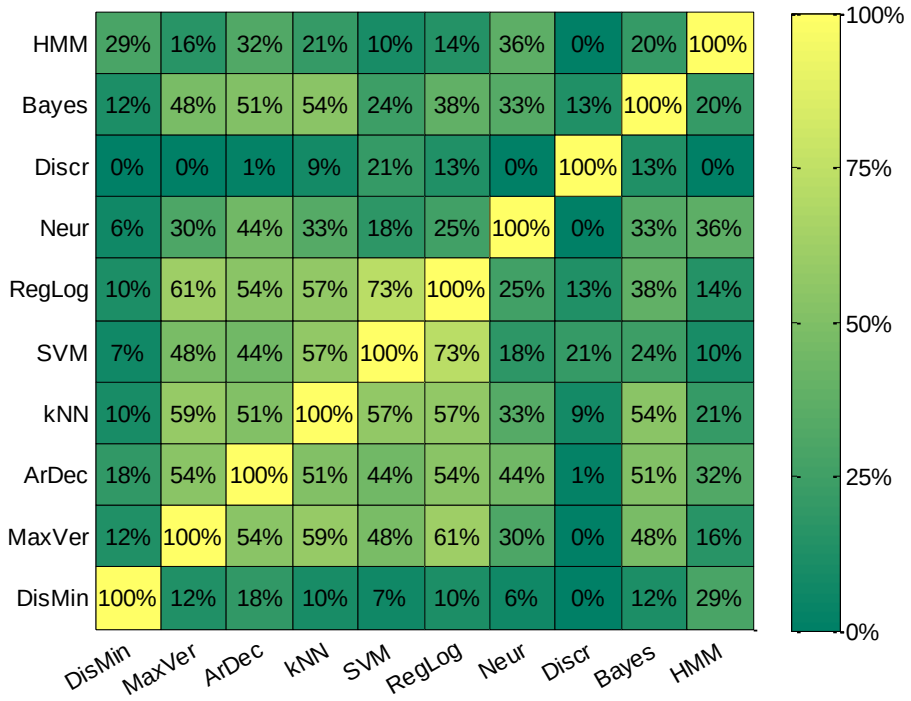


Figura 5-37 Comparación de los métodos de clasificación con ventana deslizante (tamaño ventana: 5) mediantes coeficientes kappa de Cohen

5.6.3. Modelos ocultos de Markov sobre el sonido completo

La Figura 5-38 muestra el resultado de la clasificación mediante modelos ocultos de Markov aplicados a secuencias de sonidos constituidas por el sonido completo. En ella se han utilizado 5 parámetros para definir cada *frame*. El resultado global de la clasificación puede resumirse en la Figura 5-39.

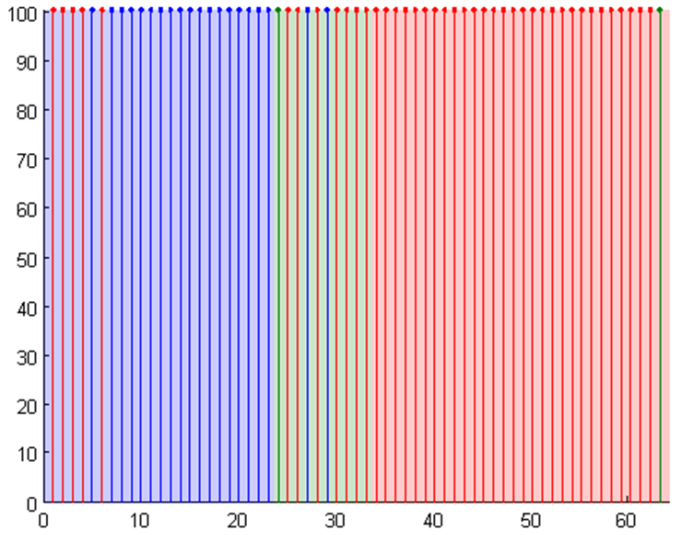


Figura 5-38 Clasificación por modelo oculto de Markov sobre el sonido completo

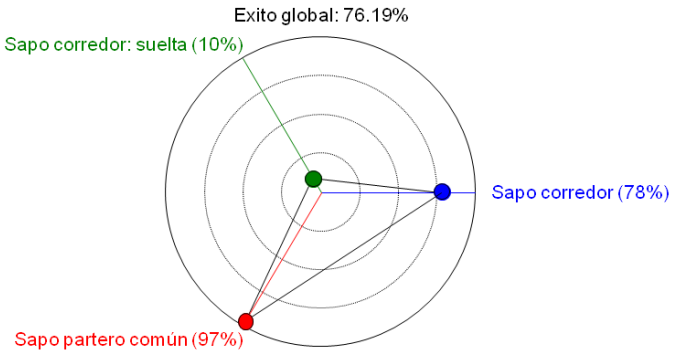


Figura 5-39 Resumen de la clasificación por modelo oculto de Markov sobre el sonido completo

Los resultados obtenidos de la clasificación por HMM sobre el sonido completo serán comparados con los resultados de los distintos método de clasificación no secuencial en $\mathbb{R}^5$. Para empezar, la Figura 5-40 refleja de forma gráfica esta comparación. La altura de la barra refleja la tasa global de éxito para cada clasificador. Los puntos, con el código de colores habitual, indican la tasa de éxito para cada tipo de canto.

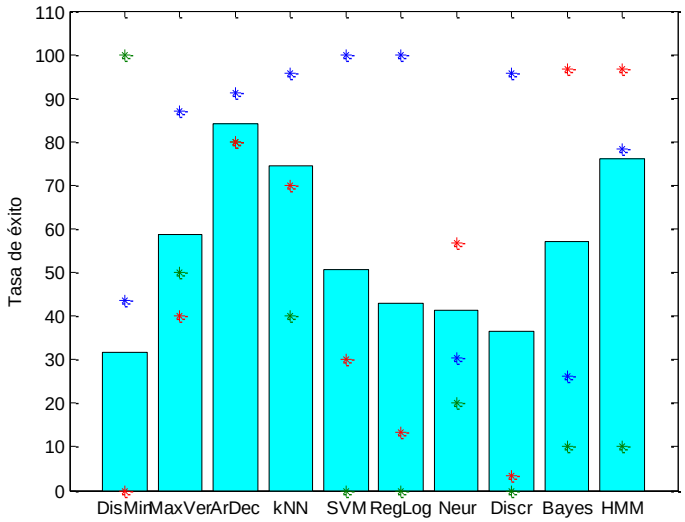


Figura 5-40 Resultados de la clasificación no secuencial en $\mathbb{R}^5$ y HMM

La Figura 5-41 refleja el mérito de cada técnica de clasificación mediante la representación del rango de la tasa de error frente la tasa de error.

La Tabla 5-11 recoge el factor de mérito de cada uno de los clasificadores utilizados.

La Tabla 5-12 recoge los valores de los indicadores de exactitud, precisión, sensibilidad, especificidad y tasa de errores.

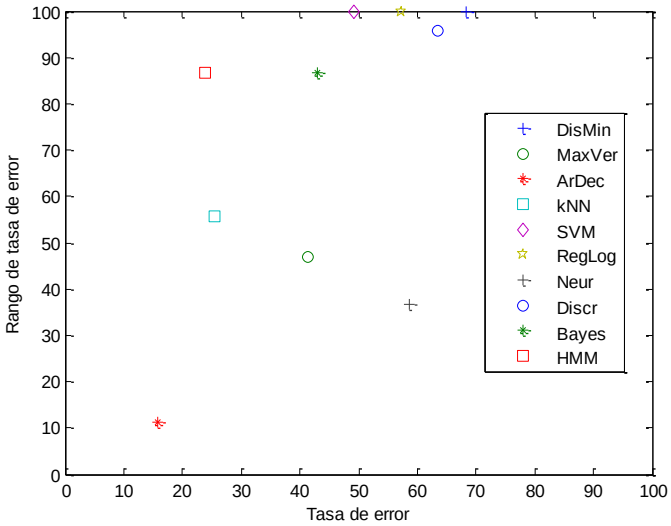


Figura 5-41 Tasa de error y su rango (unidades en %) para clasificación no secuencial en $\mathbb{R}^5$ y HMM

Algoritmo	Acierto	Errores	Rango	Distancia origen	Mérito
Distnacia mínima	31.75%	68.25%	100%	1.21	14.39%
Máxima verosimilitud	58.73%	41.27%	47%	0.63	55.80%
Árboles de decisión	84.13%	15.87%	11%	0.19	86.22%
k-vecinos más próximos	74.60%	25.40%	56%	0.61	56.74%
SVM	50.79%	49.21%	100%	1.11	21.19%
Regresión logística	42.86%	57.14%	100%	1.15	18.56%
Redes neuronales	41.27%	58.73%	37%	0.69	51.04%
Función discriminante	36.51%	63.49%	96%	1.15	18.82%
Clasificador bayesiano	57.14%	42.86%	87%	0.97	31.63%
Modelo oculto de Markov	76.19%	23.81%	87%	0.90	36.45%

Tabla 5-11. Factor de mérito para clasificación no secuencial en $\mathbb{R}^5$ y HMM

Algoritmo	Exactitud	Tasa de errores	Precisión	Sensib.	Especif.
Distancia mínima	54.50%	45.50%	-	47.83%	72.96%
Máxima verosimilitud	72.49%	27.51%	66.74%	58.99%	78.21%
Árboles de decisión	89.42%	10.58%	85.40%	83.77%	91.52%
k-vecinos más próximos	83.07%	16.93%	84.97%	68.55%	86.49%
SVM	67.20%	32.80%	-	43.33%	74.17%
Regresión logística	61.90%	38.10%	46.55%	37.78%	70.20%
Redes neuronales	73.54%	26.46%	60.61%	51.59%	81.02%
Función discriminante	57.67%	42.33%	-	33.00%	66.49%
Clasificador bayesiano	71.43%	28.57%	66.85%	44.25%	74.49%
Modelo oculto de Markov	84.13%	15.87%	70.24%	61.64%	85.58%

Tabla 5-12. Indicadores para la evaluación de clasificadores no secuencial en $\mathbb{R}^5$ y HMM

En la Figura 5-42 se representa el análisis ROC de los distintos algoritmos estudiados donde de nuevo el mejor clasificador es el árbol de decisión.

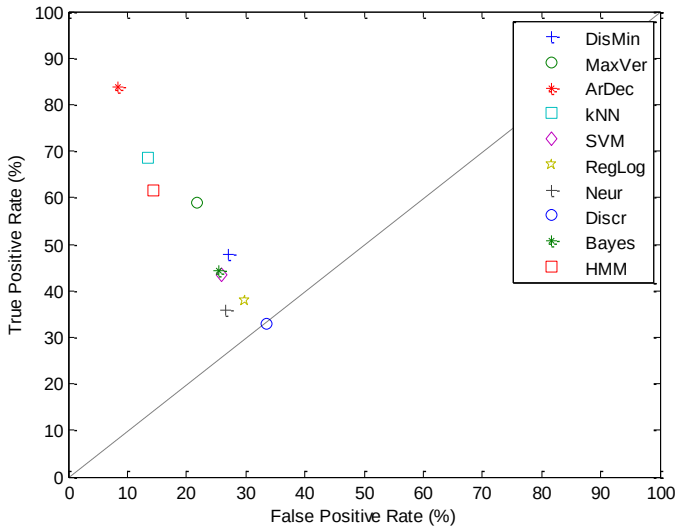


Figura 5-42 Comparación de los métodos de clasificación no secuencial en $\mathbb{R}^5$ y HMM mediantes análisis ROC

En la Figura 5-43 se representa la concordancia de los resultados usando los coeficientes kappa.

La conclusión en $\mathbb{R}^5$ tras analizar las distintas comparativas realizadas, es que la clasificación por HMM sobre el sonido completo aunque mejora los resultados respecto a su uso sobre ventanas deslizantes, sigue teniendo peores resultados respecto al árbol de decisión en $\mathbb{R}^5$, ya sea sobre ventana deslizante o no secuencial.

Figura 5-43 Comparación de los métodos de clasificación no secuencial en $\mathbb{R}^5$ y HMM mediantes coeficientes kappa de Cohen

5.6.4. Modelos ocultos de Markov sobre secuencias tamaño ROI

La Figura 5-44 muestra el resultado de la clasificación mediante modelos ocultos de Markov aplicados a secuencias de sonidos constituidas por secuencias consecutivas (sin ventanas deslizantes) cuyo tamaño es el de la duración de una ROI (*Region Of Interest*) media, es decir, la duración promedio de las zonas de interés de los sonidos patrón. En estos modelos se han utilizado 5 parámetros para definir cada *frame*. El resultado global de la clasificación puede resumirse en la Figura 5-45.

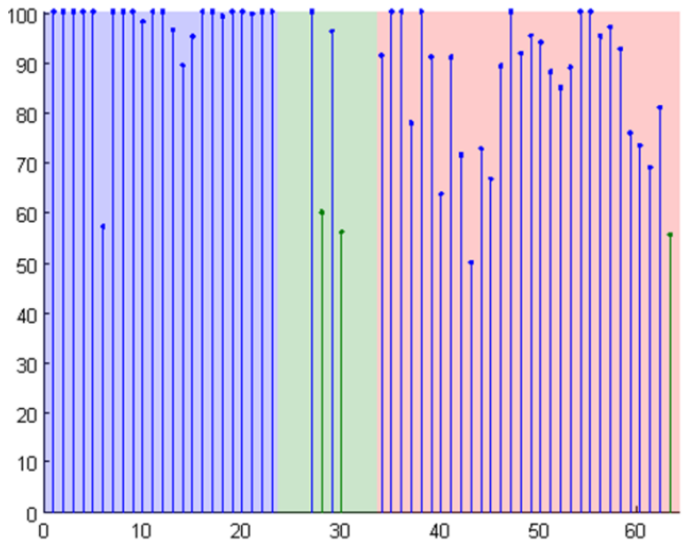


Figura 5-44 Clasificación por modelo oculto de Markov sobre secuencias tamaño ROI

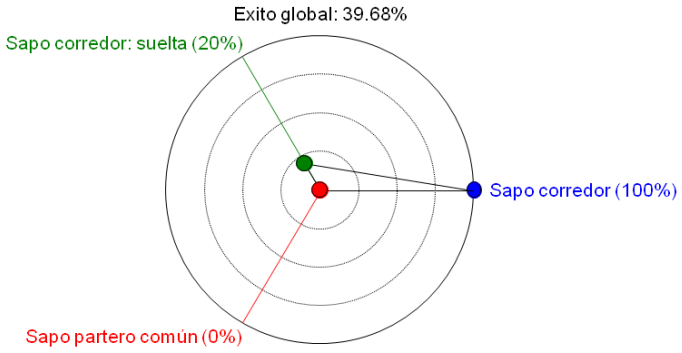


Figura 5-45 Resumen de la clasificación por modelo oculto de Markov sobre secuencias tamaño ROI

La Figura 5-46 refleja de forma gráfica la comparación de esta clasificación secuencial (modelo oculto de Markov sobre el sonido completo) con la clasificación no secuencial $\mathbb{R}^5$. La altura de la barra refleja la tasa global de éxito para cada clasificador. Los puntos, con el código de colores habitual, indican la tasa de éxito para cada tipo de canto.

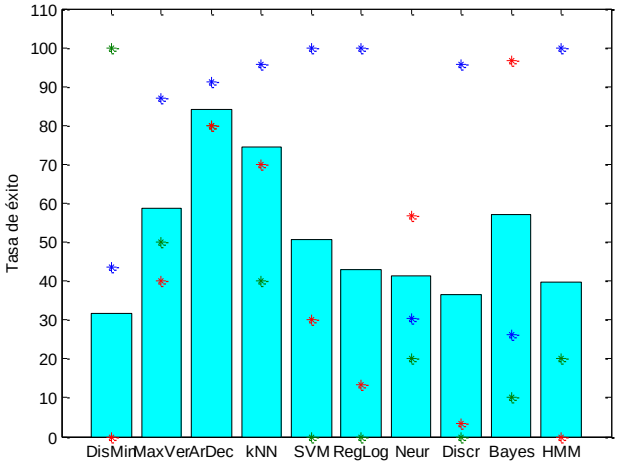


Figura 5-46 Resultados de la clasificación no secuencial en $\mathbb{R}^5$ y HMM sobre secuencias tamaño ROI

La Figura 5-47 refleja el mérito de cada técnica de clasificación mediante la representación del rango de la tasa de error frente la tasa de error.

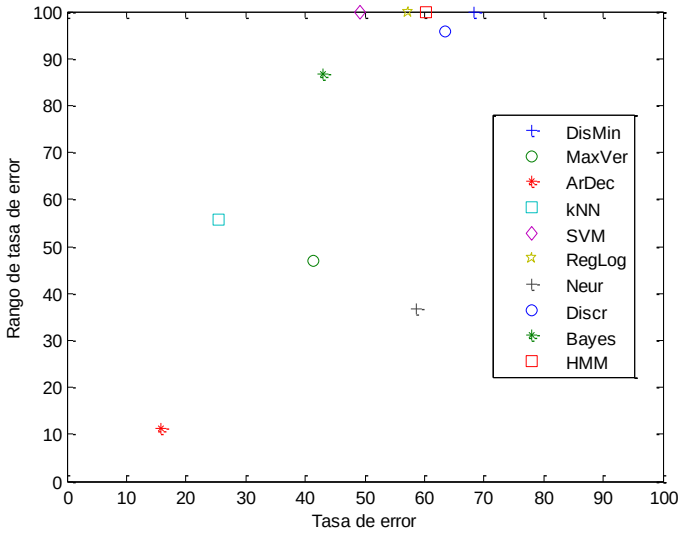


Figura 5-47 Tasa de error y su rango (unidades en %) para clasificación no secuencial en $\mathbb{R}^5$ y HMM sobre secuencias tamaño ROI

Las Tabla 5-13 y Tabla 5-14 recogen el factor de mérito y los valores de los indicadores de exactitud, precisión, sensibilidad, especificidad y tasa de errores, respectivamente.

En la Figura 5-48 se representa el análisis ROC de los distintos algoritmos estudiados donde de nuevo el mejor clasificador es el árbol de decisión.

Algoritmo	Acierto	Errores	Rango	Distancia origen	Mérito
Distnacia mínima	31.75%	68.25%	100%	1.21	14.39%
Máxima verosimilitud	58.73%	41.27%	47%	0.63	55.80%
Árboles de decisión	84.13%	15.87%	11%	0.19	86.22%
k-vecinos más próximos	74.60%	25.40%	56%	0.61	56.74%
SVM	50.79%	49.21%	100%	1.11	21.19%
Regresión logística	42.86%	57.14%	100%	1.15	18.56%
Redes neuronales	41.27%	58.73%	37%	0.69	51.04%
Función discriminante	36.51%	63.49%	96%	1.15	18.82%
Clasificador bayesiano	57.14%	42.86%	87%	0.97	31.63%
Modelo oculto de Markov	39.68%	60.32%	100%	1.17	17.42%

Tabla 5-13. Factor de mérito para clasificación no secuencial en $\mathbb{R}^5$ y HMM sobre secuencias tamaño ROI

Algoritmo	Exactitud (%)	Tasa de errores (%)	Precisión (%)	Sensib. (%)	Especif. (%)
Distancia mínima	54.50%	45.50%	-	47.83%	72.96%
Máxima verosimilitud	72.49%	27.51%	66.74%	58.99%	78.21%
Árboles de decisión	89.42%	10.58%	85.40%	83.77%	91.52%
k-vecinos más próximos	83.07%	16.93%	84.97%	68.55%	86.49%
SVM	67.20%	32.80%	-	43.33%	74.17%
Regresión logística	61.90%	38.10%	46.55%	37.78%	70.20%
Redes neuronales	73.54%	26.46%	60.61%	51.59%	81.02%
Función discriminante	57.67%	42.33v	-	33.00%	66.49%
Clasificador bayesiano	71.43%	28.57%	66.85%	44.25%	74.49%
Modelo oculto de Markov	59.79%	40.21%	-	40.00%	68.54%

Tabla 5-14. Indicadores para la evaluación de clasificadores no secuencial en $\mathbb{R}^5$ y HMM sobre secuencias tamaño ROI

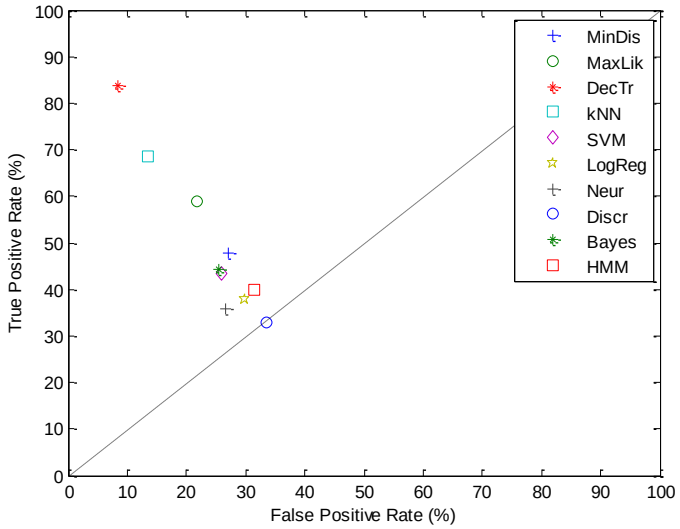


Figura 5-48 Comparación de los métodos de clasificación no secuencial en $\mathbb{R}^5$ y HMM sobre secuencias tamaño ROI mediantes análisis ROC

En la Figura 5-49 se representa la concordancia de los resultados usando los coeficientes kappa.

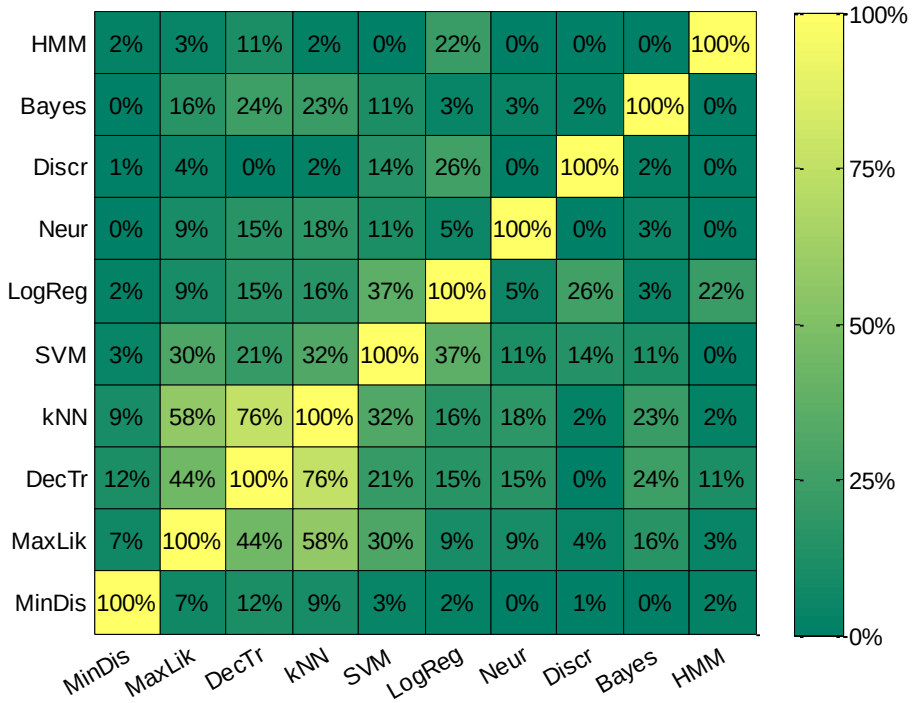


Figura 5-49 Comparación de los métodos de clasificación no secuencial en $\mathbb{R}^5$ y HMM sobre secuencias tamaño ROI mediantes coeficientes kappa de Cohen

La conclusión, en $\mathbb{R}^5$ tras analizar las distintas comparativas realizadas, es que la clasificación por HMM sobre el sonido completo aunque mejora los resultados respecto a su uso sobre ventanas deslizantes, sigue teniendo peores resultados respecto al árbol de decisión en $\mathbb{R}^5$, ya sea sobre ventana deslizante o no secuencial.

5.7. Clasificación de modelos ARIMA

5.7.1. Modelado de series temporales

Una de las técnicas más utilizadas en la descripción de secuencias temporales de datos son las series temporales (Box, Jenkins, & Reinsel, 2011) y, dentro de ellas, los denominados modelos ARIMA (*AutoRegressive Integrated Moving Average*).

Consideremos la secuencia temporal (serie temporal) de valores x_t que toma uno de los parámetros que caracterizan un *frame* de sonido. Se dice que dicha serie temporal sigue un modelo autoregresivo de orden p, $AR(p)$, si

$$x_t = a_0 + \sum_{i=1}^{p} a_i x_{t-i} + \varepsilon_t \ , \qquad \textbf{(5.6)}$$

expresión en la que los a_i son los coeficientes del modelo y ε_t es una variable aleatoria. De forma análoga, se dice que la serie temporal x_t sigue un modelo de media móvil de orden q, $MA(q)$, si

$$x_t = b_0 + \sum_{i=1}^{q} b_i \varepsilon_{t-i} + \varepsilon_t \ , \tag{5.7}$$

expresión en la que los b_i son los coeficientes del modelo y ε_j es una variable aleatoria. Combinando las anteriores expresiones, se dice que la serie temporal x_t sigue un modelo de autoregresivo con media móvil de orden p, q, $ARMA(p, q)$, si

$$x_t = c_0 + \sum_{i=1}^{p} a_i x_{t-i} + \sum_{i=1}^{q} b_i \varepsilon_{t-i} + \varepsilon_t \ . \tag{5.8}$$

Para que el análisis de series temporales sea efectivo se requiere que la serie sea estacionaria, es decir, que sus características permanezcan constantes a lo largo del tiempo. En ocasiones esta condición no lo cumple la serie original pero si la serie de diferencias, también denominada serie integrada de orden 1

$$x_t^{(1)} \equiv \Delta x_t = x_t - x_{t-1} = (1 - L) x_t \ , \tag{5.9}$$

expresión en la que L es el operador "retraso de una unidad de tiempo". Si tampoco la serie $x_t^{(1)}$ es estacionaria, se puede calcular de nuevo la diferencia sobre ella, de forma que

$$x_t^{(2)} \equiv \Delta x_t^{(1)} = x_t^{(1)} - x_{t-1}^{(1)} = (1 - L) x_t^{(1)} = (1 - L)(1 - L) x_t = (1 - L)^2 x_t \ . \tag{5.10}$$

El proceso se puede repetir varias veces, de forma que la serie integrada de orden d es

$$x_t^{(d)} \equiv (1 - L)^d x_t \ . \tag{5.11}$$

Combinando todo lo expuesto hasta ahora, se dice que la serie temporal x_t sigue un modelo de autoregresivo integrado con media móvil de orden p, d, q, $ARIMA(p, d, q)$, si

$$x_t^{(d)} = c_0 + \sum_{i=1}^{p} a_i x_{t-i}^{(d)} + \sum_{i=1}^{q} b_i \varepsilon_{t-i} + \varepsilon_t \ . \tag{5.12}$$

En el caso de la secuencia de *frames* de sonido que nos ocupa, cada uno de ellos no viene identificado por un único parámetro x_t, sino por un vector de n parámetros $\boldsymbol{X_t} \in \mathbb{R}^n$. Para acomodar esta nueva realidad, se extienden los conceptos anteriores al de series temporales vectoriales que pueden ser descritas mediante modelos VARIMA (ARIMA vectorial). Un modelo $VARIMA(p, d, q)$ se describe de la siguiente forma

$$\boldsymbol{X_t^{(d)} = C_0 + \sum_{i=1}^{p} A_i X_{t-i}^{(d)} + \sum_{i=1}^{q} B_i \varepsilon_{t-i} + \varepsilon_t} \ , \qquad \textbf{(5.13)}$$

expresión en la que los términos $\boldsymbol{X_j^{(d)}}$, $\boldsymbol{\varepsilon_j}$ y $\boldsymbol{C_0}$ son vectores de dimensión n, y los términos $\boldsymbol{A_i}$ y $\boldsymbol{B_i}$ son matrices de coeficientes de dimensión $n \times n$.

Dada una secuencia $\boldsymbol{X_t}$ se pueden obtener las matrices de coeficientes del modelo $VARIMA(p, d, q)$ que mejor se le ajusta mediante diversas técnicas entre las que destaca la de máxima verosimilitud (Hevia, 2008).

El primer paso para la construcción de un modelo VARIMA es la elección de la estructura del modelo, es decir, la determinación de los valores de p, d y q. El problema general es bastante complejo por lo que se suele abordar con algunas simplificaciones. En primer lugar, dada la naturaleza de las series temporales, parámetros derivados de señales de voz, es razonable asumir que las series son estacionarias, es decir, que se puede considerar $d = 0$.

Por otra parte, dado un modelo VARMA, se puede construir a partir de él otro modelo VAR que con bastante aproximación, se comporte de forma equivalente (Box et al., 2011). Por tanto, en este trabajo y por simplicidad, se utilizarán modelos VAR ($q = 0$) quedando pendiente de determinar el orden del mismo, el valor de p. Para ello se recurre al Criterio de Información Akaike (Akaike, 1974), conocido como AIC por sus siglas en inglés.

El procedimiento que se ha seguido es:

1. Considerar cada una de las Regiones de Interés (ROIs) de los patrones.
2. Obtener las secuencias de parámetros en $\mathbb{R}^5$ correspondientes a cada ROI. Por las razones expuestas en los apartados anteriores se utilizará un número de parámetros $n = 5$.
3. Estimar los parámetros de modelos VAR de distinto orden.
4. Obtener el AIC correspondiente a cada modelo de cada ROI.

Para una mejor comparación, el valor de los AIC obtenidos se normaliza en el intervalo $[0,1]$ correspondiendo el 1 al mejor de los modelos. El resultado obtenido se muestra en la Figura 5-50 en la que los puntos blancos representan la media ponderada del orden del modelo para cada ROI. La línea blanca, análogamente, representa la media ponderada del orden del modelo para todos los ROIs.

En la Figura 5-51 se recoge en forma de barras la información sobre la media ponderada del orden del modelo para cada ROI y para todos los ROIs, la cual tiene el valor $\mu = 3.42$. Como el orden del modelo elegido tiene que ser un número entero, finalmente se determina que el modelo óptimo será un VAR de orden $p = 3$. Y una vez

elegido la estructura del modelo VAR(3), sus parámetros se estiman por el método de máxima verosimilitud (Box et al., 2011).

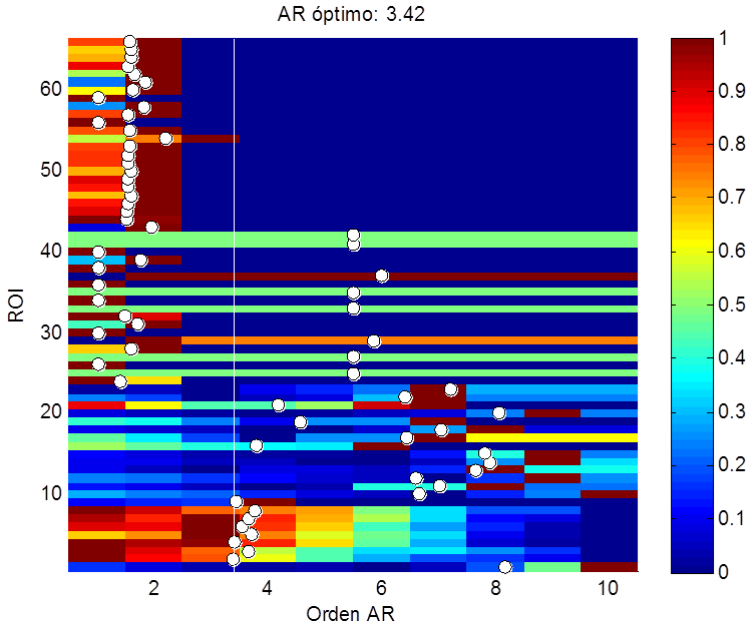


Figura 5-50 Valores del AIC normalizado

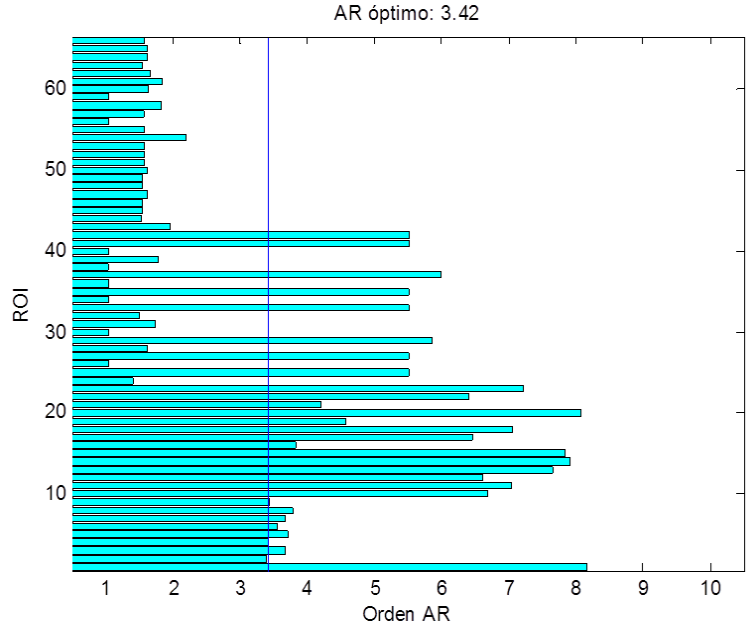


Figura 5-51 Media ponderada del orden del modelo para cada ROI

El modelo VAR(3) tendrá la forma

$$\boldsymbol{X_t} = \sum_{i=1}^{3} \boldsymbol{A_i X_{t-i}} + \boldsymbol{\varepsilon_t} \ , \qquad \textbf{(5.14)}$$

expresión en la que el término $\boldsymbol{A_i}$ son matrices de coeficientes de dimensión $n \times n$, 5×5 en el caso de estudio.

El número total de parámetros del modelo VAR(3) es

$$N = p \times (n \times n) \tag{5.15}$$

que, en este caso, vale $N = 75$.

5.7.2. Clasificación de series temporales

En el apartado anterior se ha establecido cómo modelar una serie temporal vectorial, en el caso de estudio en $\mathbb{R}^5$, de tal forma que cada serie viene definida por un punto en $\mathbb{R}^{75}$. A partir de este resultado es posible aplicar técnicas de minería de datos a la clasificación de las series temporales (Deng, Moore, & Nechyba, 1997). Para ello se procede mediante los siguientes pasos:

1. Se obtienen los distintos segmentos de sonido que constituyen los archivos patrón, ROIs y no ROIs.
2. Se estiman los parámetros VAR(3) de cada uno de esos segmentos patrón y se representan mediante un punto en $\mathbb{R}^{75}$. En la Figura 5-52 se refleja esta nueve de puntos para dos parámetros cualesquiera de los segmentos patrón. Los puntos se han codificado con el código de colores habitual para cada tipo de sonido.
3. Se considera un archivo de sonido desea clasificar y se divide en segmentos cuya duración sea la duración media de los segmentos patrón.
4. De cada uno de los segmentos anteriores se estiman los parámetros VAR(3) y se representan mediante un punto en $\mathbb{R}^{75}$.
5. Cada uno de los puntos anteriores se somete a un algoritmo de clasificación supervisada, por comparación con los puntos en $\mathbb{R}^{75}$ correspondientes a los segmentos patrón.
6. Para clasificar modelos VARIMA se emplearán los mismos algoritmos descritos anteriormente en el capítulo de clasificación no secuencial.

La Figura 5-53 refleja de forma gráfica la comparación de los distintos algoritmos de clasificación supervisada aplicada sobre los modelos ARIMA. La altura de la barra refleja la tasa global de éxito para cada clasificador. Los puntos, con el código de colores habitual, indican la tasa de éxito para cada tipo de canto.

La Figura 5-54 refleja el mérito de cada técnica de clasificación mediante la representación del rango de la tasa de error frente la tasa de error.

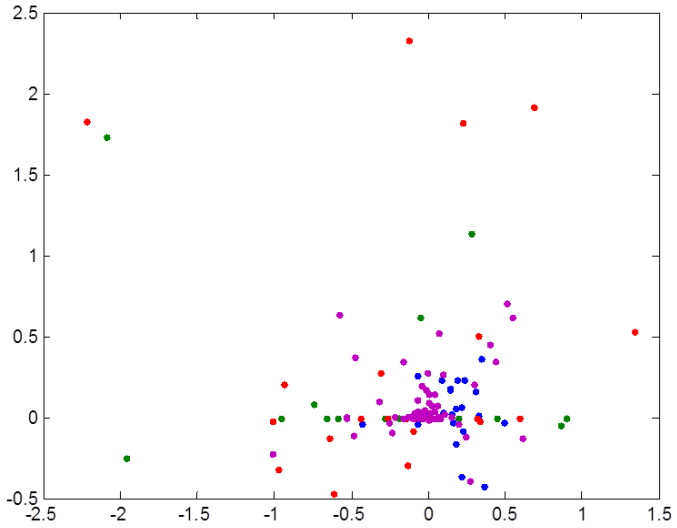


Figura 5-52 Proyección en $\mathbb{R}^2$ de la nube de puntos $\mathbb{R}^{75}$ de los segmentos patrón

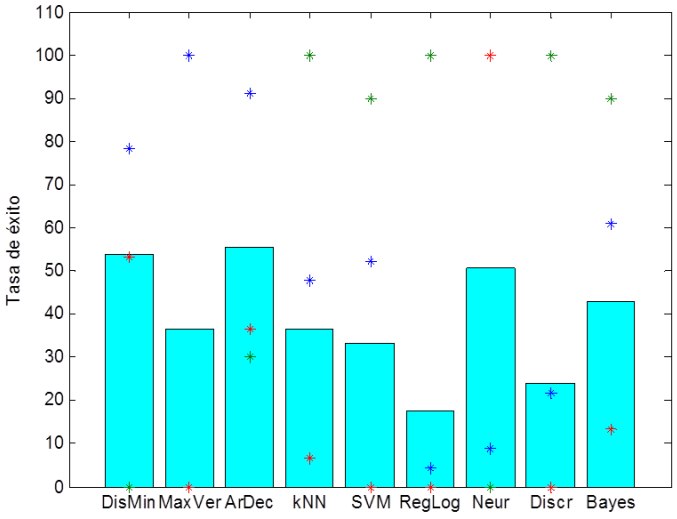


Figura 5-53 Resultados de la clasificación de modelos ARIMA

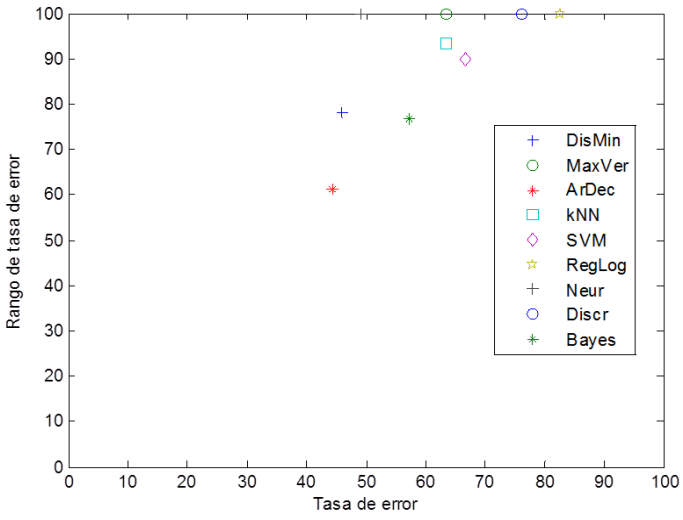


Figura 5-54 Tasa de error y su rango (unidades en %) para clasificación de modelos ARIMA

La Tabla 5-15 recoge el factor de mérito de cada uno de los clasificadores utilizados. La Tabla 5-16 recoge los valores de los indicadores de exactitud, precisión, sensibilidad, especificidad y tasa de errores. En la Figura 5-55 se representa el análisis ROC de los distintos algoritmos estudiados. Y en la Figura 5-56 se representa la concordancia de los resultados usando los coeficientes kappa.

Algoritmo	Acierto	Errores	Rango	Distancia origen	Mérito
Distancia mínima	53.97%	46.03%	78%	0.91	35.80%
Máxima verosimilitud	36.51%	63.49%	100%	1.18	16.24%
Árboles de decisión	55.56%	44.44%	61%	0.76	46.46%
k-vecinos más próximos	36.51%	63.49%	93%	1.13	20.18%
SVM	33.33%	66.67%	90%	1.12	20.80%
Regresión logística	17.46%	82.54%	100%	1.30	8.31%
Redes neuronales	50.79%	49.21%	100%	1.11	21.19%
Función discriminante	23.81%	76.19%	100%	1.26	11.10%
Clasificador bayesiano	42.86%	57.14%	77%	0.96	32.39%

Tabla 5-15 Factor de mérito para clasificación de modelos ARIMA

Algoritmo	Exactitud	Tasa de errores	Precisión	Sensib.	Especif.
Distancia mínima	69.31%	30.69%	41.58%	43.86%	76.05%
Máxima verosimilitud	57.67%	42.33%	-	33.33%	66.67%
Árboles de decisión	70.37%	29.63%	63.17%	52.66%	77.48%
k-vecinos más próximos	57.67%	42.33%	58.05%	51.50%	72.39%
SVM	55.56%	44.44%	39.72%	47.39%	72.06%
Regresión logística	44.97%	55.03%	-	34.78%	67.30%
Redes neuronales	67.20%	32.80%	50.88%	36.23%	70.21%
Función discriminante	49.21%	50.79%	39.18%	40.58%	69.43%
Clasificador bayesiano	61.90%	38.10%	53.53%	54.73%	75.43%

Tabla 5-16 Indicadores para la evaluación de clasificadores de modelos ARIMA

Las Tabla 5-13 y Tabla 5-14 recogen el factor de mérito y los valores de los indicadores de exactitud, precisión, sensibilidad, especificidad y tasa de errores, respectivamente.

En la Figura 5-48 se representa el análisis ROC de los distintos algoritmos estudiados donde de nuevo el mejor clasificador es el árbol de decisión.

La conclusión tras analizar las distintas comparativas realizadas, es que la clasificación por árboles de decisión ofrece los mejores resultados cuando se usan modelos ARIMA.

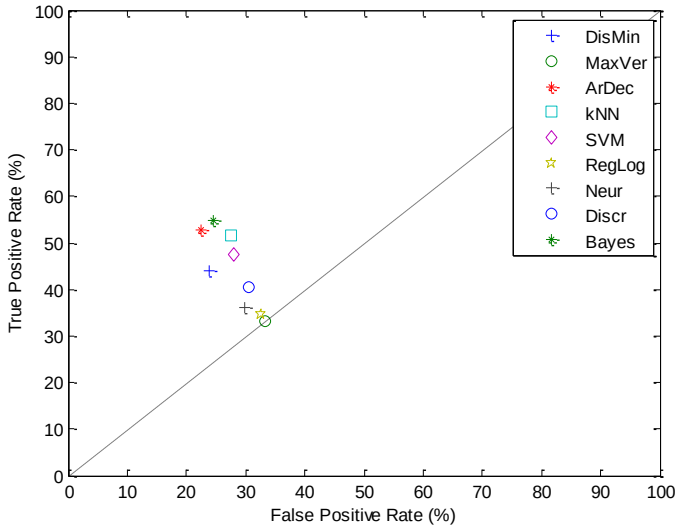


Figura 5-55 Comparación de los métodos de de modelos ARIMA mediantes análisis ROC

Figura 5-56 Comparación de los métodos de de modelos ARIMA mediantes coeficientes kappa de Cohen

5.8. Comparación de clasificadores secuenciales

A lo largo de este capítulo se han ido presentando distintas técnicas de clasificación secuencial. Es pues el momento de compararlas entre ellas y con las técnicas no secuenciales. En la Tabla 5-17 se recoge el factor de mérito obtenido en la clasificación de sonidos utilizando tanto técnicas secuenciales como no secuenciales. Las columnas corresponden a las siguientes técnicas:

- **P-18**: clasificación no secuencial de *frames* identificados mediante 18 parámetros.
- **P-5**: clasificación no secuencial de *frames* identificados mediante 5 parámetros, aplicando técnicas de reducción de dimensionalidad.
- **P-36**: clasificación no secuencial de *frames* identificados mediante 36 parámetros, los 18 originales y 18 parámetros temporales.
- **SW-5**: clasificación secuencial mediante ventana deslizante, de tamaño 5, sobre *frames* identificados mediante 5 parámetros.
- **RSW-5**: clasificación secuencial mediante ventana deslizante recursiva, de tamaño 5, sobre *frames* identificados mediante 5 parámetros.
- **ARIMA**: clasificación secuencial mediante modelos ARIMA sobre *frames* identificados mediante 5 parámetros.

En las filas, además de las técnicas ya exploradas en el capítulo anterior, se añaden la clasificación mediante Modelos Ocultos de Markov de *frames* identificados mediante 5 parámetros y aplicados a:

- Ventanas deslizantes de tamaño 5.
- El sonido completo.
- Segmentos de sonido de duración igual a la duración media de las ROIs.

Algoritmo		P-18	P-5	P-36	SW-5	RSW-5	ARIMA
Distancia mínima		16.84%	14.39%	37.50%	16.24%	16.24%	35.80%
Máxima verosimilitud		66.28%	55.80%	66.50%	63.90%	16.24%	16.24%
Árboles de decisión		81.42%	86.22%	76.96%	83.20%	26.11%	46.46%
k-vecinos más próximos		59.83%	56.74%	52.00%	46.54%	46.54%	20.18%
SVM		40.30%	21.19%	70.32%	35.83%	20.18%	20.80%
Regresión logística		50.91%	18.56%	69.57%	52.93%	31.90%	8.31%
Redes neuronales		26.71%	51.04%	44.56%	53.01%	16.24%	21.19%
Función discriminante		51.47%	18.82%	71.42%	23.87%	24.62%	11.10%
Clasificador bayesiano		35.25%	31.63%	49.82%	66.06%	16.24%	32.39%
HMM	Ventana deslizante (SW-5)		57.67%				
	Sonido completo		36.45%				
	ROI media		17.42%				

Tabla 5-17 Factor de mérito de diversas técnicas de clasificación

La Tabla 5-18 resume los resultados anteriores comparando los valores máximos del factor de mérito para cada una de las técnicas de clasificación. Aplicando este criterio se puede concluir que sólo la técnica de ventana deslizante ofrece resultados comparables con los de las técnicas no secuenciales.

Por otro lado, la Tabla 5-19 y la Tabla 5-20 realizan la comparación atendiendo a la tasa de error.

Algoritmo	Media	Máximo	Mínimo	Rango
P-18	47.67%	81.42%	16.84%	64.58%
P-5	39.38%	86.22%	14.39%	71.83%
P-36	59.85%	76.96%	37.50%	39.46%
SW-5	49.06%	83.20%	16.24%	66.96%
RSW-5	23.81%	46.54%	16.24%	30.30%
ARIMA	23.61%	46.46%	8.31%	38.15%
Markov SW-5	57.67%	57.67%	57.67%	0%
Markov sonido completo	36.45%	36.45%	36.45%	0%
Markov ROI media	17.42%	17.42%	17.42%	0%

Tabla 5-18 Factor de mérito de diversas técnicas de clasificación (resumen)

Algoritmo		P-18	P-5	P-36	SW-5	RSW-5	ARIMA
Distancia mínima		61.90%	68.25%	53.97%	63.49%	63.49%	46.03%
Máxima verosimilitud		20.63%	41.27%	25.40%	26.98%	63.49%	63.49%
Árboles de decisión		12.70%	15.87%	12.70%	14.29%	39.68%	44.44%
k-vecinos más próximos		26.98%	25.40%	31.75%	28.57%	28.57%	63.49%
SVM		26.98%	49.21%	12.70%	42.86%	52.38%	66.67%
Regresión logística		34.92%	57.14%	15.87%	34.92%	80.95%	82.54%
Redes neuronales		36.51%	39.68%	41.27%	28.57%	63.49%	49.21%
Función discriminante		33.33%	63.49%	19.05%	63.49%	57.14%	76.19%
Clasificador bayesiano		28.57%	42.86%	47.62%	20.63%	63.49%	57.14%
HMM	Ventana deslizante (SW-5)		46.03%				
	Sonido completo		23.81%				
	ROI media		60.32%				

Tabla 5-19 Tasa de error de diversas técnicas de clasificación

Algoritmo	Media	Máximo	Mínimo	Rango
P-18	31.39%	61.90%	12.70%	49.20%
P-5	44.80%	68.25%	15.87%	52.38%
P-36	28.93%	53.97%	12.70%	41.27%
SW-5	35.98%	63.49%	14.29%	49.20%
RSW-5	56.96%	80.95%	28.57%	52.38%
ARIMA	61.02%	82.54%	44.44%	38.10%
Markov SW-5	46.03%	46.03%	46.03%	0%
Markov sonido completo	23.81%	23.81%	23.81%	0%
Markov ROI media	60.32%	60.32%	60.32%	0%

Tabla 5-20 Tasa de error de diversas técnicas de clasificación (resumen)

De nuevo, la conclusión es que sólo la técnica de ventana deslizante ofrece unos resultados comparables, en este caso ligeramente mejores, que las técnicas no secuenciales. Aplicando un clasificador mediante árboles de decisión se obtiene un tasa de error del 14.29%, lo que se considera adecuado teniendo en cuenta la mala calidad de los archivos de sonidos procesados.

CAPÍTULO 6. CLASIFICACIÓN DE SERIES DERIVADAS

6.1. Introducción a las series derivadas

Antes de definir qué es una serie derivada, resulta conveniente recordar los procesos realizados en los anteriores capítulos, donde partiendo de los sonidos se ha llegado a realizar una clasificación secuencial automáticas de los mismos.

Partiendo de los archivos de sonidos, el primer paso es segmentar el archivo en *frames* sobre los que se realiza la extracción de características, basadas en 18 parámetros de la norma MPEG-7. Como se indicó en el capítulo 2 y 3, para la obtención de estos parámetros es necesaria la realización de distintas técnicas (Figura 6-1):

- Análisis espectro-temporal
    - Potencia total
    - Potencia relevante
    - Centroide de potencia
    - Dispersión espectral de potencia
    - Planitud del espectro
- Codificación predictiva lineal (LPC)
    - Picos armónicos o formantes
        - Frecuencia de los 3 primeros formantes
        - Anchos de banda de los 3 primeros formantes
    - Tono (*Pitch*)
    - Centroide armónico
    - Desviación armónica
    - Dispersión armónica
    - Variación armónica
- Análisis de armonicidad basadas en autocorrelación

- Razón de armonicidad
- Frecuencia límite de armonicidad

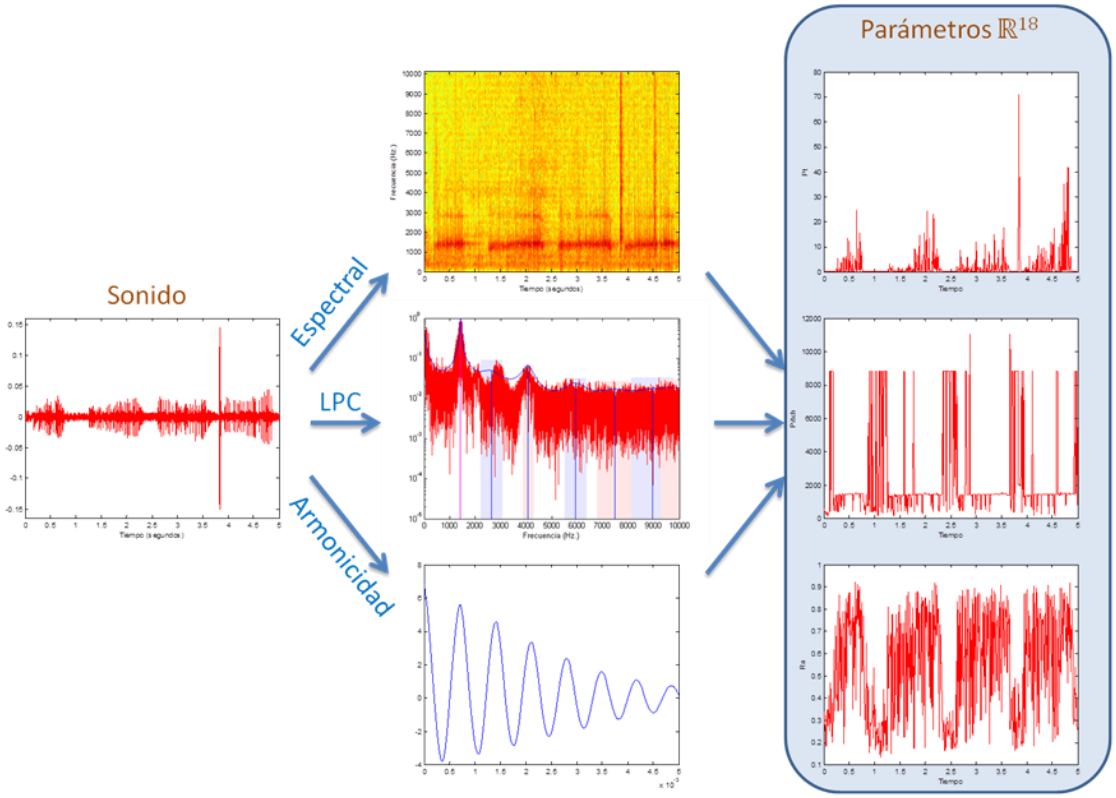


Figura 6-1 Obtención de parámetros MPEG-7 a partir de un archivo de sonido

Tras este primer paso, se dispone de una nube de puntos en $\mathbb{R}^{18}$ que representan a cada *frame* del archivo de audio. Sobre esta nube de punto se han aplicado distintas técnicas de clasificación supervisada: no secuencial y secuencial.

Como se vio en el capitulo anterior, los mejores resultados se obtenían con una clasificación secuencial basada en ventanas deslizantes de tamaño 5. Para este método resultaba interesante trabajar con un número menor de parámetros, por lo que se aplicaba una técnica de reducción de dimensionalidad, basada en (Luque et al., 2016) para obtener los 5 parámetros más significativos de cada uno de los algoritmos usados en la clasificación no secuencial. De esta forma, en lugar de trabajar con espacio $\mathbb{R}^{90}$ ($\mathbb{R}^{5\cdot 18}$) se trabajaba con un espacio $\mathbb{R}^{25}$ ($\mathbb{R}^{5\cdot 5}$) (Figura 6-2).

Llegados a este punto, en lugar de clasificar el sonido por conteo de la clase mayoritaria en función de la clasificación de los *frames*, se introduce el concepto de serie derivada. El clasificador ofrece para cada *frame* una probabilidad de pertenencia a cada una de las clases, los tres cantos de anuros más el ruido. Esta información se puede ver como 4 series temporales o una serie vectorial, que es lo que se denominará como serie derivada, definida en un espacio $\mathbb{R}^4$ (Figura 6-3).

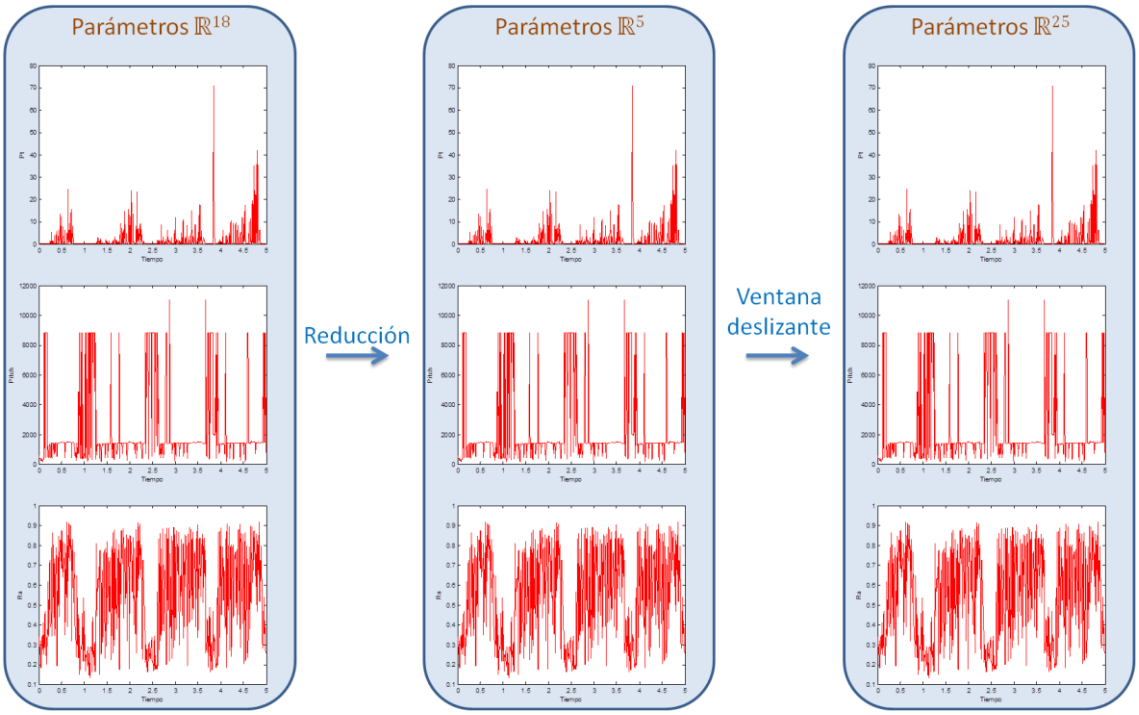


Figura 6-2 Reducción de dimensionalidad y aplicación de ventana deslizante

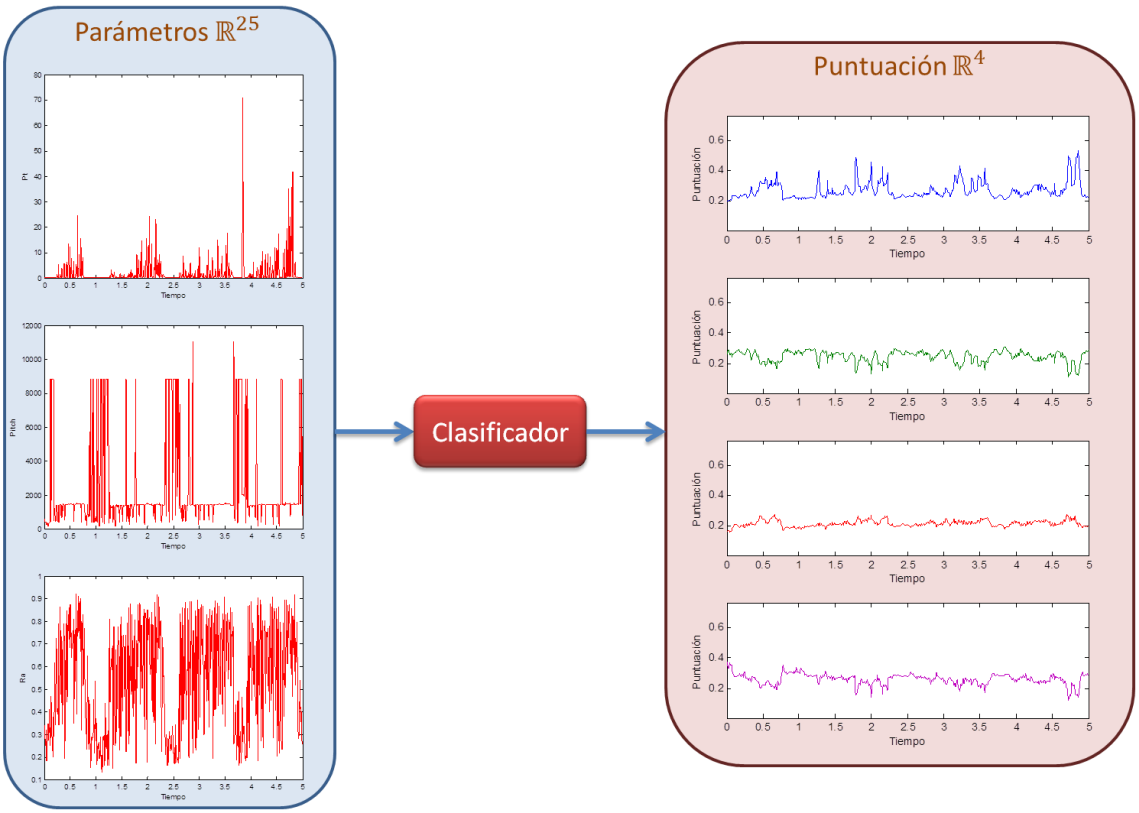


Figura 6-3 Reducción de dimensionalidad y aplicación de ventana deslizante

6.2. Clasificación de series derivadas

A partir de la nube de puntos en $\mathbb{R}^4$ que define la serie derivada se toma una decisión de clasificación del archivo de sonido. Esta clasificación se puede realizar de diversas formas, entre las cuales se encuentran: conteo, semejanza o paramétrica.

6.2.1. Clasificación por conteo

Este tipo de clasificación ha sido el utilizado en los capítulos anteriores. Para cada *frame* se clasifica como perteneciente a la clase de mayor puntuación. En la Figura 6-4 se ha indicado, en la parte superior del gráfico, una franja del color correspondiente al tipo de canto clasificado. Una vez clasificados todos los *frames* se realiza un conteo del número asignado a cada clase de sonido. Finalmente se clasifica el archivo de sonido como perteneciente a la clase de sonido con un valor mayor de conteo.

La Figura 6-5 muestra el resultado de la aplicación de uno de los algoritmos usados en capítulos anteriores, árboles de decisión, al conjunto de archivos de sonido disponibles. Este es el algoritmo que mejores resultados obtuvo en la clasificación secuencial con ventanas deslizantes. En horizontal se representan los archivos, ordenados por tipo de sonido: sapo corredor (zona azul); sapo corredor en canto de suelta (zona verde); sapo partero (zona roja). Por cada archivo existe una línea vertical cuyo color se corresponde con la clasificación realizada por el algoritmo (con el mismo código de colores anterior). En una clasificación perfecta el código de cada línea debería corresponder con la de la zona del gráfico. Cada discrepancia supone un error de clasificación. Por último, la altura de cada línea es la probabilidad que el algoritmo asigna a la clasificación realizada. En una clasificación perfecta el código de cada línea debería corresponder con la de la zona del gráfico. Cada discrepancia supone un error de clasificación. Por último, la altura de cada línea es la probabilidad que el algoritmo asigna a la clasificación realizada.

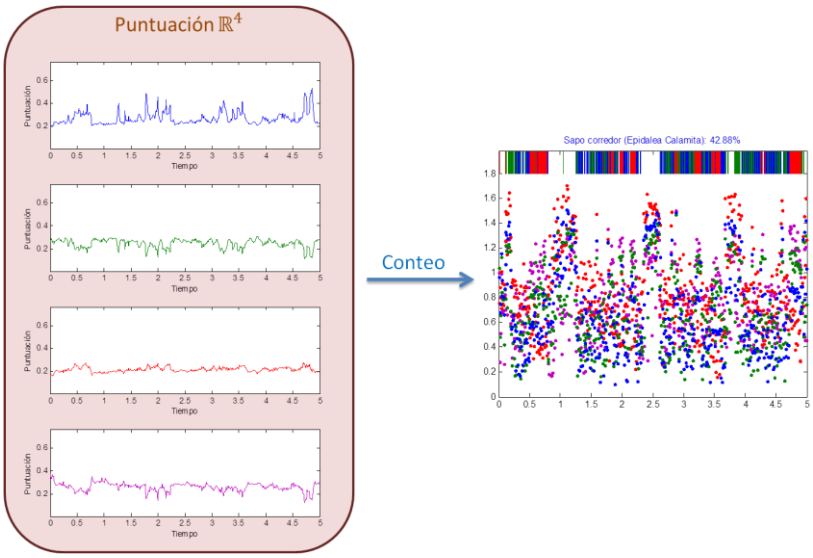


Figura 6-4 Clasificación por conteo

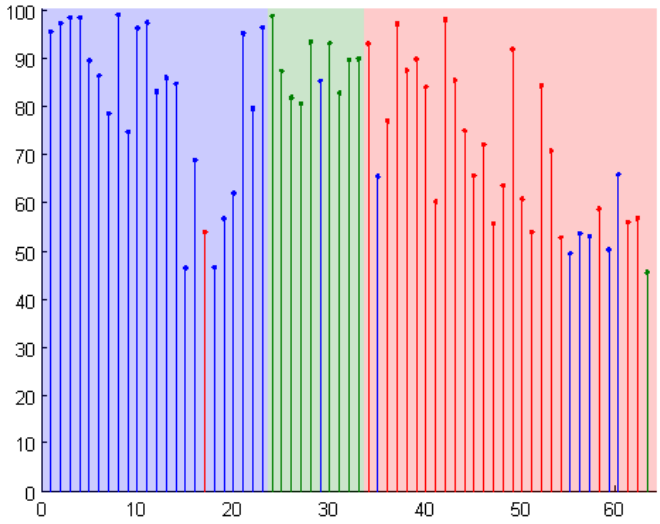


Figura 6-5 Clasificación por árbol de decisión (conteo)

El resultado global de la clasificación mediante árboles de decisión puede observarse en la Figura 6-6.

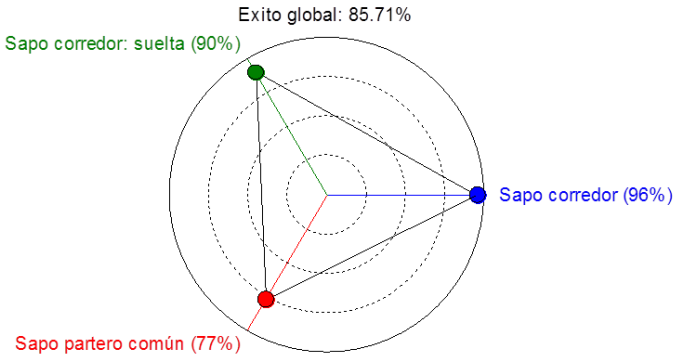


Figura 6-6 Resumen de la clasificación por árbol de decisión (conteo)

Hasta aquí no ha sido más que un resumen de todos los pasos realizados para la clasificación secuencial con ventanas deslizantes del capítulo anterior. Aunque es un método bastante simplista no necesariamente es sinónimo de ineficiencia.

6.2.2. Clasificación por semejanza

La aportación más interesante del apartado anterior es, si cabe, plantear el problema de la clasificación de sonidos bajo una nueva óptica. La obtención y consideración de las series derivadas lleva a plantear el método de clasificación de estas series, sin dar por hecho que el conteo sea el mejor de ellos.

En este sentido, son numerosas las posibles técnicas para abordar esta cuestión, algunas de las cuales (Modelos Ocultos de Markov y Modelos ARIMA), fueron ya presentadas en el capítulo anterior dentro del enfoque de la clasificación de secuencias temporales de parámetros de *frame*.

Por otro lado, la minería de datos ofrece un conjunto más amplio de técnicas para abordar el problema, existiendo en la literatura buenos resúmenes de estas técnicas (Esling & Agon, 2012; Fu, 2011). También es posible encontrar aportaciones con soluciones basadas en la aplicación de alguna técnica específica, como la redes neuronales (Koskela, 2003). O trabajos enfocados a problemas concretos de minería de datos, como la clasificación (Kadous & Sammut, 2005) o la predicción (Ahmed, Atiya, Gayar, & El-Shishiny, 2010).

La mayoría de las técnicas de minería de datos aplicadas a la clasificación de series derivadas buscan encontrar la semejanza entre una determinada serie temporal y un conjunto de series patrón. Una de las técnicas conceptualmente más simple de lograrlo, pero a la vez reputada como de alta eficiencia es la clasificación denominada 1NN-DTW (Xi, Keogh, Shelton, Wei, & Ratanamahatana, 2006). Esta técnica combina un algoritmo de deformación temporal dinámica (*Dynamic Time Warping*, DTW) para comparar series temporales, con un clasificador kNN (k-vecinos más próximos) de orden 1, visto en el capítulo 4.

El DTW es una técnica surgida del problema de la medida de distancia entre series temporales cuando no existe una sincronización o alineamiento temporal. Esta falta de alineamiento no obedece a una ley fija, sino que la duración, en distintos segmentos, puede aumentar o disminuir respecto la serie patrón.

El primer paso de la técnica 1NN-DTW es la alineación temporal de las series derivadas, la serie patrón con la serie a clasificar. Para ver más claro el proceso DTW se verá, en primer lugar, dos series temporales escalares (en $\mathbb{R}^1$), como las de la Figura 6-7, a las que se denominaran p (roja) y s (azul). Estas series se supondrán que tienen el mismo número de elementos n, es decir, que se corresponden con sonidos con el mismo número de *frames* y, por tanto, ambos de la misma duración.

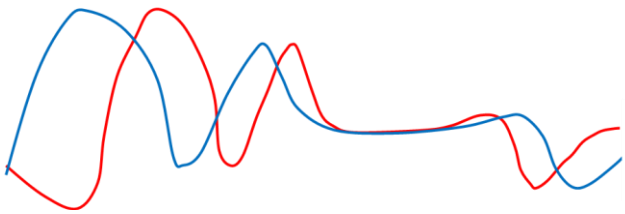


Figura 6-7 Series temporales de la misma duración

En este supuesto en el que ambas series tienen el mismo número de elementos, n, la distancia entre ambas se puede determinar comparando elemento a elemento (Figura 6-8), de acuerdo con la expresión

$$d(p,s) = \sum_{i=1}^{n} |p_i - s_i| \,. \qquad \textbf{(6.1)}$$

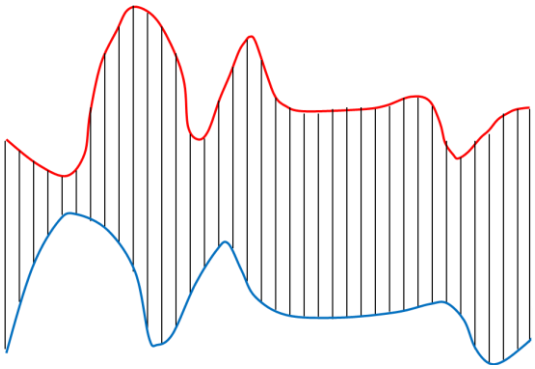


Figura 6-8 Distancia entre series temporales de la misma duración

En el caso, que una de las series tuviera más elementos m con $m > n$, se desecharía los valores comprendidos en el rango $[n+1, m]$, como se ilustra en la Figura 6-9, pudiéndose usar la anterior expresión (6.1) para la medida de la distancia entre ambas series. A priori, este no es un buen método pues se están perdiendo valores que podrían ser claves en la comparación.

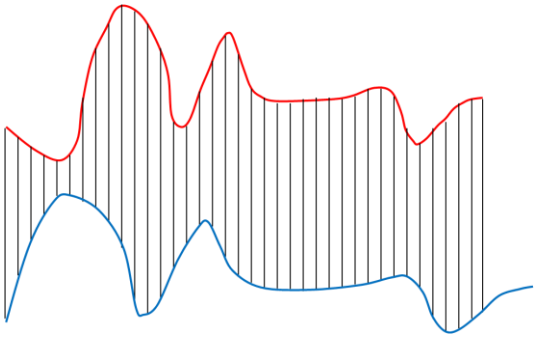


Figura 6-9 Distancia entre series temporales de distinta duración

Sin embargo, intuitivamente se aprecia que la semejanza entre ambas series es más alta que la que se obtiene por comparación elemento a elemento, ya que ambas coinciden si se deformaran, comprimiendo y expandiendo de forma no lineal, el eje de tiempo. Esto se consigue haciendo uso del algoritmo DTW donde se comparan los elementos de la serie que más semejanza tengan entre sí (Figura 6-10). Además, se resuelve el caso en el que las dos series no tienen la misma longitud.

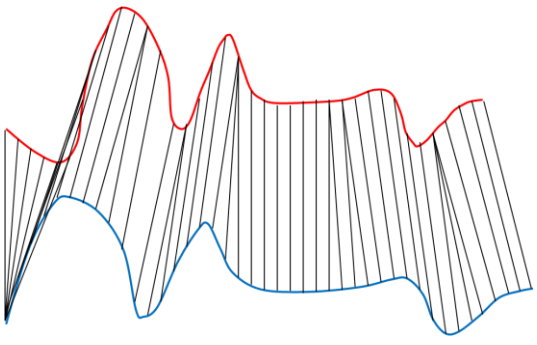


Figura 6-10 Distancia entre series temporales con DTW

Suponiendo que la serie p tiene n elementos y la serie s tiene m. Se comienza comparando los elementos p_1 y s_1, se concluye comparando los elementos p_n y s_m. Pero en la zona intermedia se van comparando los elementos que mayor semejanza presenten. Esto va trazando una ruta r de comparación, conocida como camino de alineamiento, que contiene q elementos (Figura 6-11).

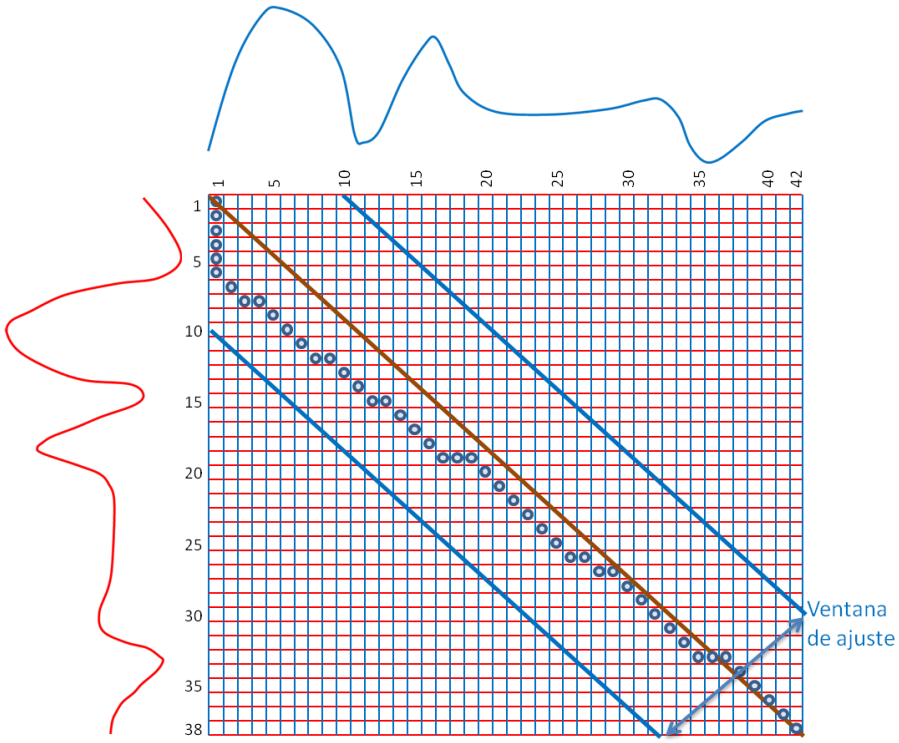


Figura 6-11 Distancia entre series temporales con DTW

Si el elemento k-ésimo de la ruta r compara los *frames* p_i y s_j, la distancia correspondiente será

$$d(r_k) = d(p_i, s_j) = \sum_{i=1}^{n} |p_i - s_j| \,. \tag{6.2}$$

La distancia entre las dos series será

$$d(p, s) = \sum_{k=1}^{q} r_k \,. \tag{6.3}$$

El algoritmo DTW va construyendo la ruta r óptima, de tal forma que la distancia sea mínima. El esfuerzo computacional del algoritmo es bastante elevado, por lo que habitualmente suele limitarse usando unas restricciones que pueden ser locales y/o globales. Las restricciones globales permiten reducir el número de cálculos limitando la cantidad máxima de compresión/expansión, de forma que no se permitan todos los puntos posibles en la parrilla de la Figura 6-11, limitándose el ámbito de la ruta a una

zona de ancho determinado w alrededor de la diagonal, denominada ventaja de ajuste. Por otro lado, las restricciones locales se usan para limitar el margen local del camino indicando sobre qué puntos vecinos al actual puede conectarse, dando lugar a la denominada restricción de pendiente.

El problema de calcular el camino de alineamiento se resuelve mediante técnicas de programación dinámica, motivo por el que se conoce a esta técnica como alineamiento temporal dinámico. Una alternativa a la programación dinámica podría ser calcular todos los caminos posibles, contando con las restricciones introducidas, y elegir posteriormente el de coste mínimo. No obstante, esta alternativa resulta impracticable a efectos de cálculo por la multiplicidad de caminos posibles.

Si las series temporales $\boldsymbol{p}$ y $\boldsymbol{s}$ son vectoriales de dimensión v, como es el caso de las series derivadas, entonces la distancia se calcula como

$$d(\boldsymbol{p}, \boldsymbol{s}) = \sqrt{\sum_{i=1}^{v} [d(p^i, s^i)]^2}\,, \tag{6.4}$$

expresión en la que p^i y s^i son las series escalares de la dimensión i-ésima de las series vectoriales $\boldsymbol{p}$ y $\boldsymbol{s}$ respectivamente.

Una vez comparada la serie derivada que se quiere clasificar, con las correspondientes series de los sonidos patrones mediante el cálculo de la distancia anterior, se aplica una clasificador 1NN (1-vecino más próximo), es decir, que el archivo se considera que pertenece a la clase de cuyo patrón se encuentre más cerca.

La Figura 6-12 refleja de forma gráfica la comparación del resultado de clasificación por semejanza de las distintas series derivadas. La altura de la barra refleja la tasa global de éxito para cada serie derivada. Los puntos (con el código de colores habitual) indican la tasa de éxito para cada tipo de canto.

Siguiendo con la comparación de resultados, la Figura 6-13 refleja el mérito de clasificación de cada serie derivada mediante un gráfico que enfrenta el Rango de la tasa de error con la Tasa de error.

La Tabla 6-1 recoge los valores de los indicadores de exactitud, precisión, sensibilidad, especificidad y tasa de errores de la clasificación por semejanza de las distintas series derivadas.

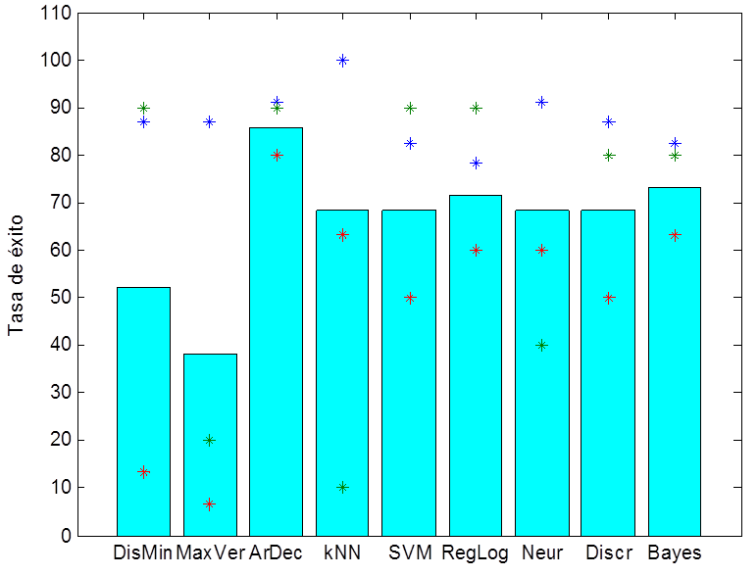


Figura 6-12 Resultados de la clasificación por semejanza de series derivadas

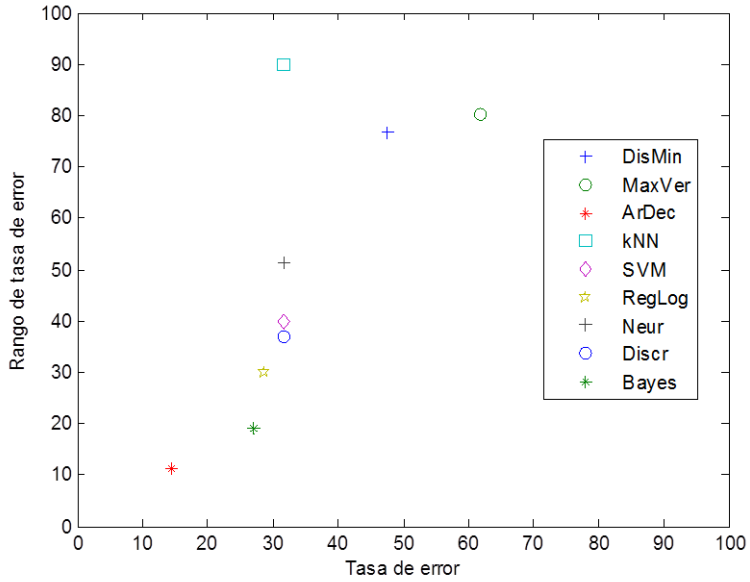


Figura 6-13 Tasa de error y su rango (unidades en %) para la clasificación por semejanza

Algoritmo	Exactitud	Tasa de errores	Precisión	Sensib.	Especif.
Distancia mínima	68.25%	31.75%	65.16%	63.43%	77.45%
Máxima verosimilitud	58.73%	41.27%	41.67%	37.87%	69.37%
Árboles de decisión	90.48%	9.52%	89.10%	87.10%	92.15%
k-vecinos más próximos	78.84%	21.16%	84.50%	57.78%	83.33%
SVM	78.84%	21.16%	71.41%	74.20%	84.97%
Regresión logística	80.95%	19.05%	73.57%	76.09%	85.08%
Redes neuronales	78.84%	21.16%	62.12%	63.77%	83.70%
Función discriminante	78.84%	21.16%	70.13%	72.32%	84.00%
Clasificador bayesiano	82.01%	17.99%	80.66%	75.31%	85.13%

Tabla 6-1 Indicadores para la evaluación de clasificación por semejanza

En la Figura 6-14 se representa el análisis ROC de los distintos algoritmos estudiados donde de nuevo el mejor clasificador es el árbol de decisión.

En la Figura 6-15 se representa la concordancia de los resultados usando los coeficientes kappa de Cohen.

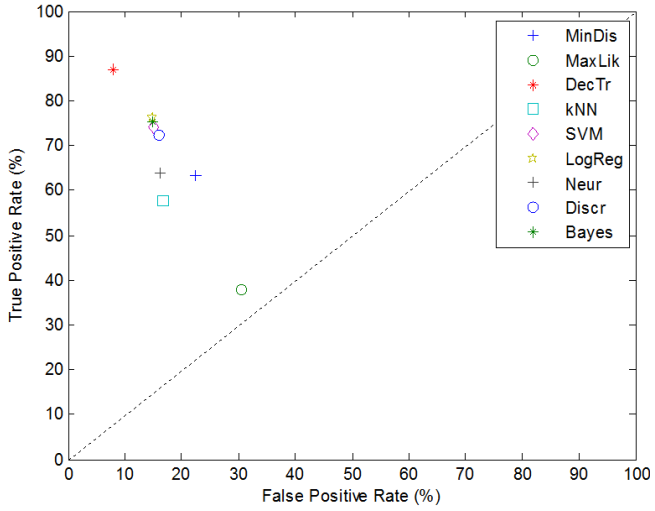


Figura 6-14 Comparación de los métodos de clasificación por semejanza mediante análisis ROC

Figura 6-15 Comparación de los métodos de clasificación por semejanza mediante coeficientes κ de Cohen

La Figura 6-16 muestra el resultado de la aplicación de esta técnica 1NN-DTW al conjunto de las series de puntuación derivadas obtenidas por la clasificación de los archivos de sonido disponibles mediante el algoritmo de árboles de decisión, que es el algoritmo que mejores resultados ha obtenido como se ha podido observar en las

anteriores comparaciones. En el cálculo se ha adoptado un ancho de la zona de deformación de $w = 100$. En horizontal se representan los archivos, ordenados por tipo de sonido: sapo corredor (zona azul); sapo corredor en canto de suelta (zona verde); sapo partero (zona roja). Por cada archivo existe una línea vertical cuyo color se corresponde con la clasificación realizada por el algoritmo (con el mismo código de colores anterior). En una clasificación perfecta el código de cada línea debería corresponder con la de la zona del gráfico. Cada discrepancia supone un error de clasificación. Por último, la altura de cada línea es la probabilidad que el algoritmo asigna a la clasificación realizada.

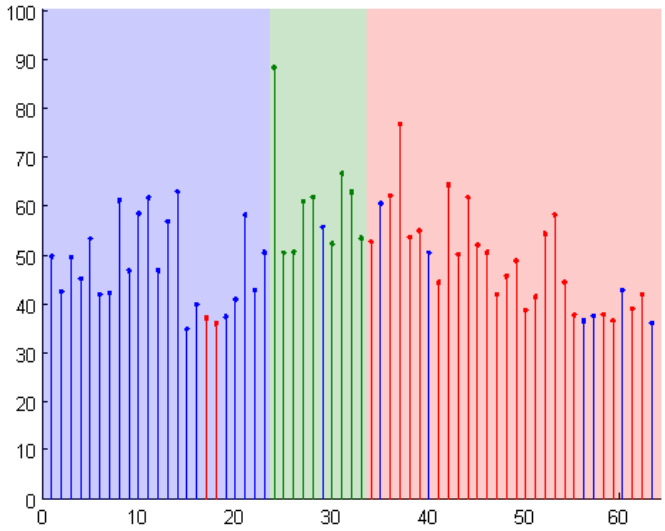


Figura 6-16 Clasificación por árbol de decisión (1NN-DTW)

El resultado global de la clasificación mediante árboles de decisión y 1NN-DTW puede resumirse en la Figura 6-17.

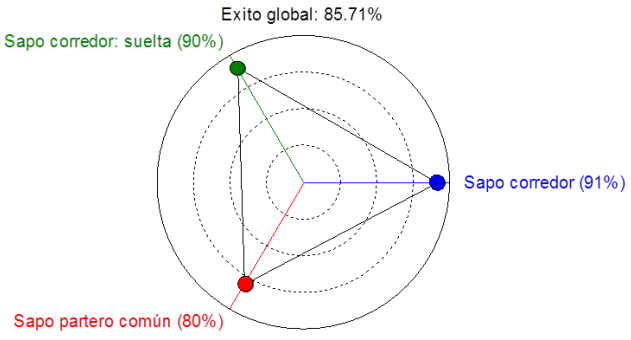


Figura 6-17 Resumen de la clasificación por árbol de decisión (1NN-DTW)

Por último, se compararan los resultados obtenidos para las clasificaciones por conteo y semejanzas. La Tabla 4-14 y Tabla 6-3 recogen la tasa de error y el factor de mérito respectivamente, en la clasificación de cada una de las series derivadas obtenidas con los nueves algoritmos usados en los capítulos anteriores.

Como puede comprobarse, la clasificación por semejanza ofrece resultados dispares en su comparación con la clasificación por conteo: la tasa de error mejora en 5 ocasiones (series derivadas), empeora en 3 y se mantiene igual en un caso (el de la serie derivada de árboles de decisión). El factor de mérito, sin embargo, mejora en 7 de las 9 series derivadas.

	Series derivadas								
	DM	MV	AD	kNN	SVM	RL	RN	FD	CB
Semejanza	47.62%	61.90%	14.29%	31.75%	31.75%	28.57%	31.75%	31.75%	26.98%
Conteo	63.49%	26.98%	14.29%	28.57%	42.86%	34.92%	28.57%	63.49%	20.63%

Tabla 6-2 Tasa de error de la clasificación por semejanza y conteo de series derivada

	Series derivadas								
	DM	MV	AD	kNN	SVM	RL	RN	FD	CB
Semejanza	36.18%	28.31%	87.12%	32.52%	63.89%	70.71%	57.34%	65.55%	76.55%
Conteo	16.24%	63.90%	83.20%	46.54%	35.83%	52.93%	53.01%	23.87%	66.06%

Tabla 6-3 Factor de mérito de la clasificación por semejanza y conteo de series derivada

De los datos anteriores cabría deducir una ligera superioridad de la clasificación por semejanza frente a la clasificación por conteo. Sin embargo, no será el método de elección por dos razones:

a) La mejor de las series derivadas es la del árbol de decisión y, en este caso, la clasificación por semejanza apenas ofrece mejora con respecto al conteo.
b) La carga computacional de la clasificación por semejanza es considerablemente más elevada que la de la clasificación por conteo.

6.2.3. Clasificación paramétrica

Para resolver el problema de la clasificación de series temporales, algunos trabajos hacen hincapié en la extracción de parámetros que las caractericen y, posteriormente, aplicar algún algoritmo de clasificación a dichos parámetros (Geurts, 2001; Povinelli, 1999).

Son diversos los parámetros que se han propuesto en la literatura para caracterizar a las series. En este trabajo, por mantener la coherencia de planteamiento, se propone considerar las series de puntuación como si fuesen sonidos, aun siendo conscientes de que no lo son, y caracterizarlas mediante parámetros MPEG-7.

Como se verá más adelante, este procedimiento quedará validado por los muy buenos resultados de clasificación que consigue alcanzar. Una visión global del proceso puede verse en la Figura 6-18.

Como se ha visto en el apartado anterior, en la obtención de las series derivadas se pueden utilizar diversos algoritmos clasificadores. En los capítulos precedentes se ha utilizado nueve de ellos, aunque el que mejor resultado se ha obtenido hasta ahora con la clasificación mediante árboles de decisión.

Partiendo, por tanto, de esa serie derivada y se procederá a su clasificación paramétrica. De las cuatro series escalares derivadas en la que se divide la serie vectorial derivada, sólo se usarán las tres asociadas a cada clase de sonido. Se desecha, por tanto, la cuarta serie que corresponde al ruido por no tener ningún interés a efectos de clasificación y porque es redundante, pues se puede calcular en función de las otras tres.

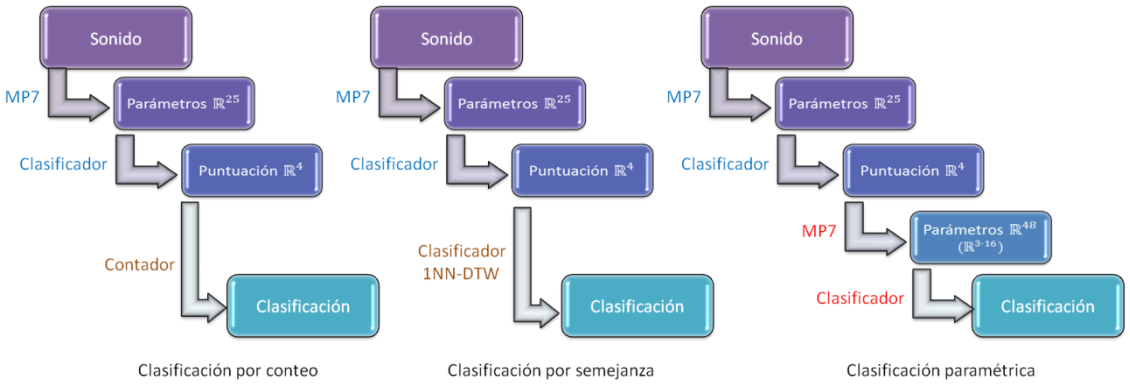


Figura 6-18 Resumen procedimiento de clasificación de series derivadas por conteo, semejanza y paramétrica

Cada serie escalar derivada seleccionada se divide en *frames* y se obtienen para cada *frame* 16 parámetros que derivan de la norma MPEG-7, construyendo una nube de puntos en $\mathbb{R}^{48}$ (3 series escalares con 16 parámetros por serie, $\mathbb{R}^{3\cdot16}$). En esta ocasión se usarán 16 parámetros, en lugar de los 18 parámetros usados anteriormente, al carecer de sentido dos de ellos: potencia relevante y variación armónica. El parámetro de potencia relevante se utiliza para eliminar ruidos fuera de la banda sonora útil. En el caso de las series derivadas, al no ser sonidos, no existen ruidos ni puede hablarse de banda de frecuencia relevante. El parámetro de variación armónica trata de capturar características de la secuencia de *frames* y no tanto las características de un *frame*.

Para el cálculo de los 16 parámetros se vuelven a aplicar las técnicas de: análisis espectro-temporal, codificación predictiva lineal (LPC) y análisis de armonicidad (basada en autocorrelación).

Un resumen de este primer paso para la clasificación paramétrica se puede ver en la Figura 6-19.

A partir de la nube de puntos en $\mathbb{R}^{48}$ y por homogeneidad en el estudio, se aplicarán los mismos nueves clasificadores que los capítulos anteriores.

Nótese que el proceso de clasificación de series derivadas, aunque es similar al enunciado para la clasificación de *frames*, presenta una diferencia notable: el número de sonidos disponibles para ser utilizados como patrón es mucho menor que el número de *frames* que cumplen esta característica.

En las clasificaciones de *frames* utilizadas en los capítulos anteriores se han utilizado más de 13.000 *frames* patrones (correspondientes a sólo algunos fragmentos de 5 sonidos). Por el contrario, para la clasificación de series derivadas, se dispone de un total de 63 sonidos y sólo una porción de ellos debe ser usada como patrón. En principio, este número reducido de patrones supone una cierta limitación en el método de clasificación que no se debe ignorar.

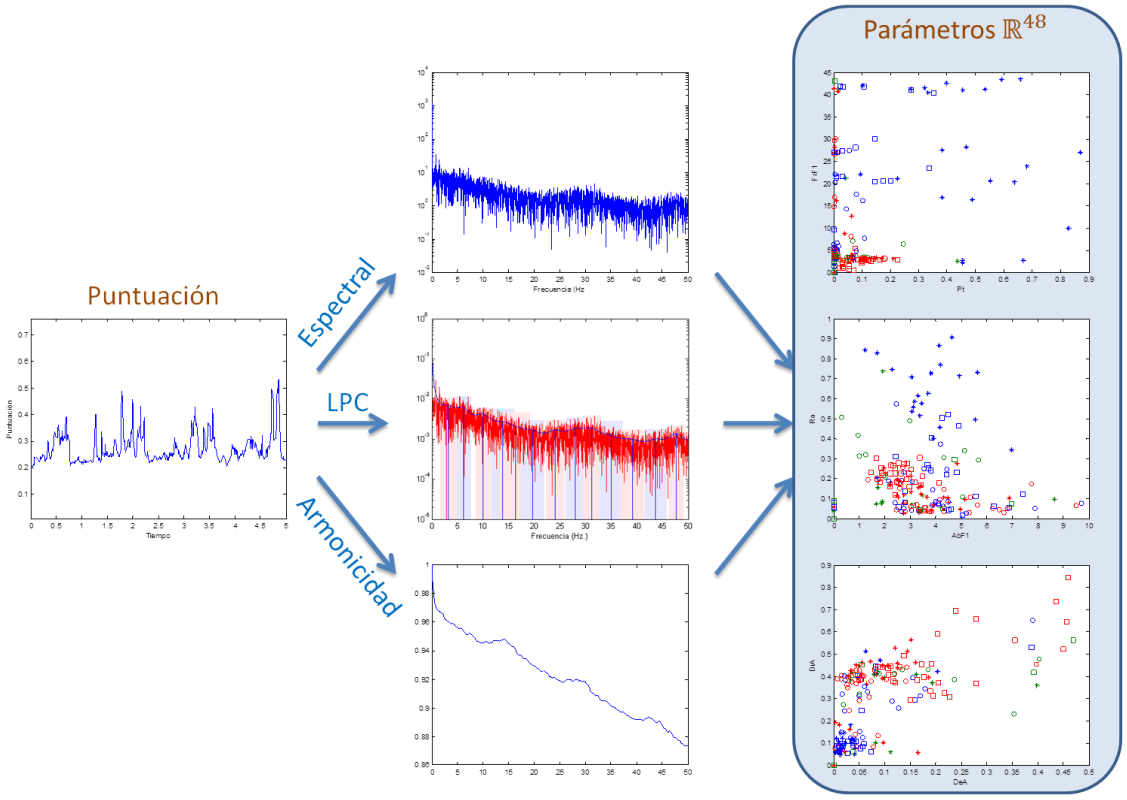


Figura 6-19 Clasificación paramétrica de series derivadas. Primer paso: obtención de parámetros MPEG-7

La Figura 6-20 recoge la dependencia de la tasa de éxito con respecto al porcentaje de archivos (π) que se usa como patrón. En esta gráfica, como ya se ha comentado en el inicio del apartado, se utiliza como serie derivada la puntuación del árbol de decisión, por ser la que mejor resultado ofreció en la clasificación por conteo, y se utilizan 9 algoritmos para clasificar dicha serie.

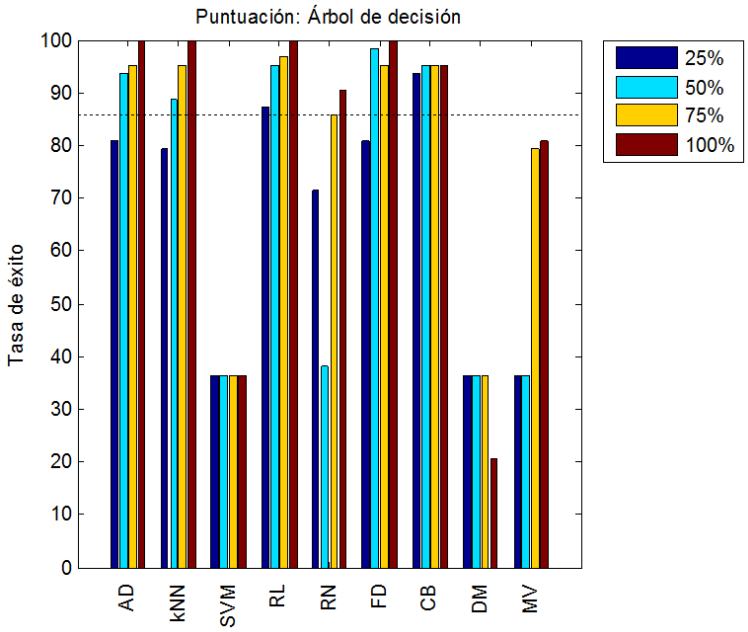


Figura 6-20 Tasa de éxito para diferentes porcentajes de archivos patrón

De la misma forma que en el capítulo anterior, por razones de eficiencia computacional, se reducirá la dimensionanilidad del problema de clasificación de $\mathbb{R}^{48}$ ($\mathbb{R}^{3.16}$) a $\mathbb{R}^{15}$ ($\mathbb{R}^{3\cdot5}$), usando sólo los 5 parámetros más significativos. Más adelante se considerará la influencia de este valor.

La línea negra discontinua que aparece en el gráfico se corresponde con la tasa de éxito en la clasificación por conteo.

La Figura 6-21 recoge el factor de mérito en las mismas circunstancias.

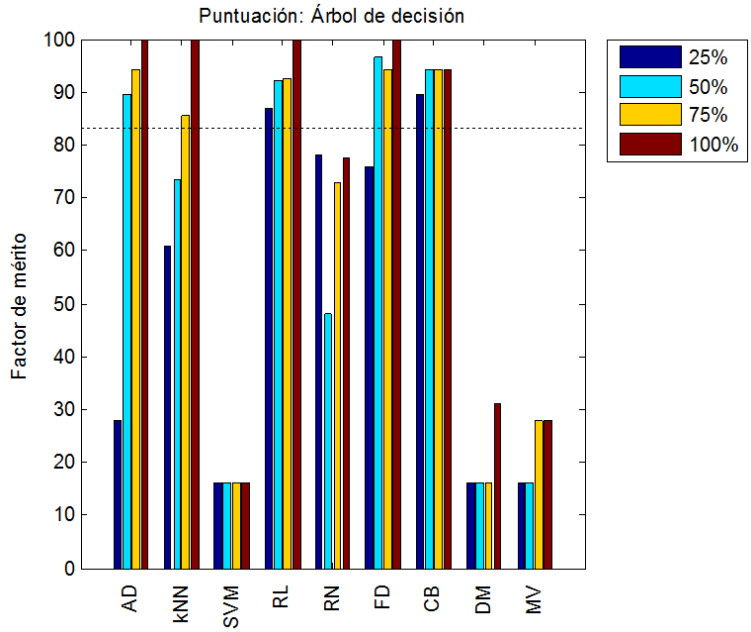


Figura 6-21 Factor de mérito para diferentes porcentajes de archivos patrón

Con estos resultados se puede ver que, usando un 25% de los archivos como patrón (16 patrones), sólo el clasificador bayesiano y, en menor medida, la regresión logística mejoran la clasificación por conteo. Sin embargo, usando un 50% o más de los archivos como patrón (32 patrones), son varios los algoritmos de clasificación que mejoran la clasificación original. Consideraremos provisionalmente estos dos supuestos.

La Figura 6-22 recoge la dependencia de la tasa de éxito con respecto al número de parámetros. En esta gráfica a partir de la serie derivada de puntuación por árbol de decisión, se aplican los 9 algoritmos para clasificar dicha serie utilizando un 25% de archivos como patrón (π=25%). La Figura 6-23 reflejan el caso en el que π=50%.

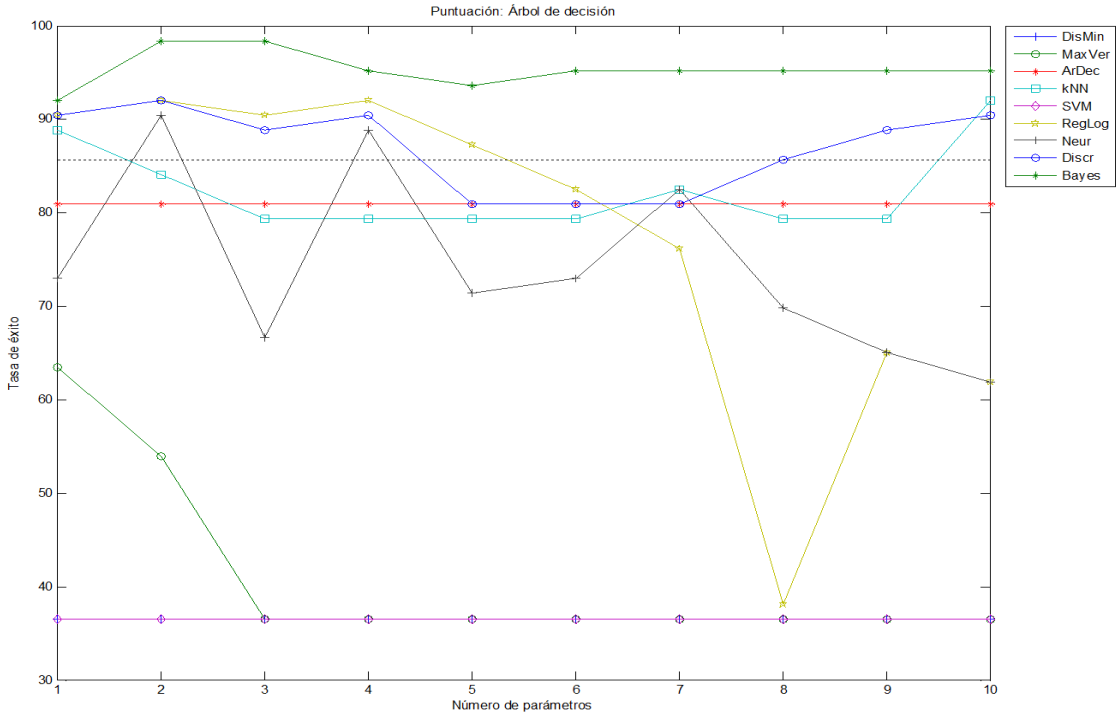


Figura 6-22 Tasa de éxito frente al número de parámetros para $\pi = 25\%$

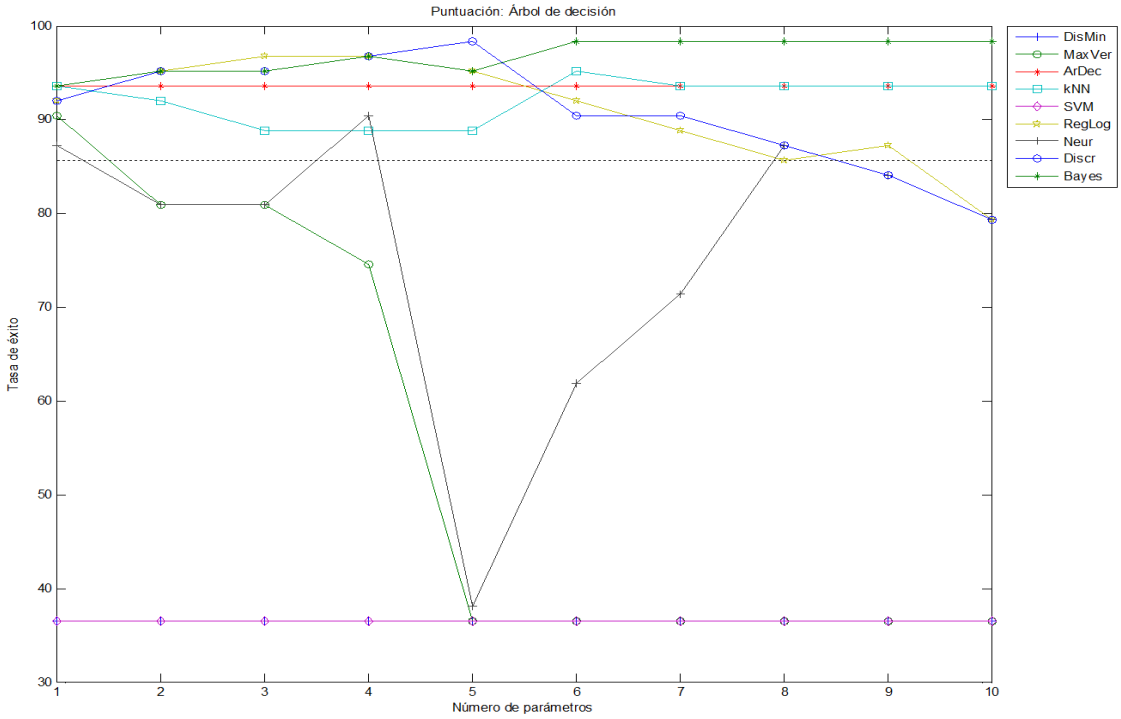


Figura 6-23 Tasa de éxito frente al número de parámetros para $\pi = 50\%$

Por otra parte, las correspondientes gráficas con el factor de mérito se recogen en la Figura 6-24 (π=25%) y en la Figura 6-25 (π=50%).

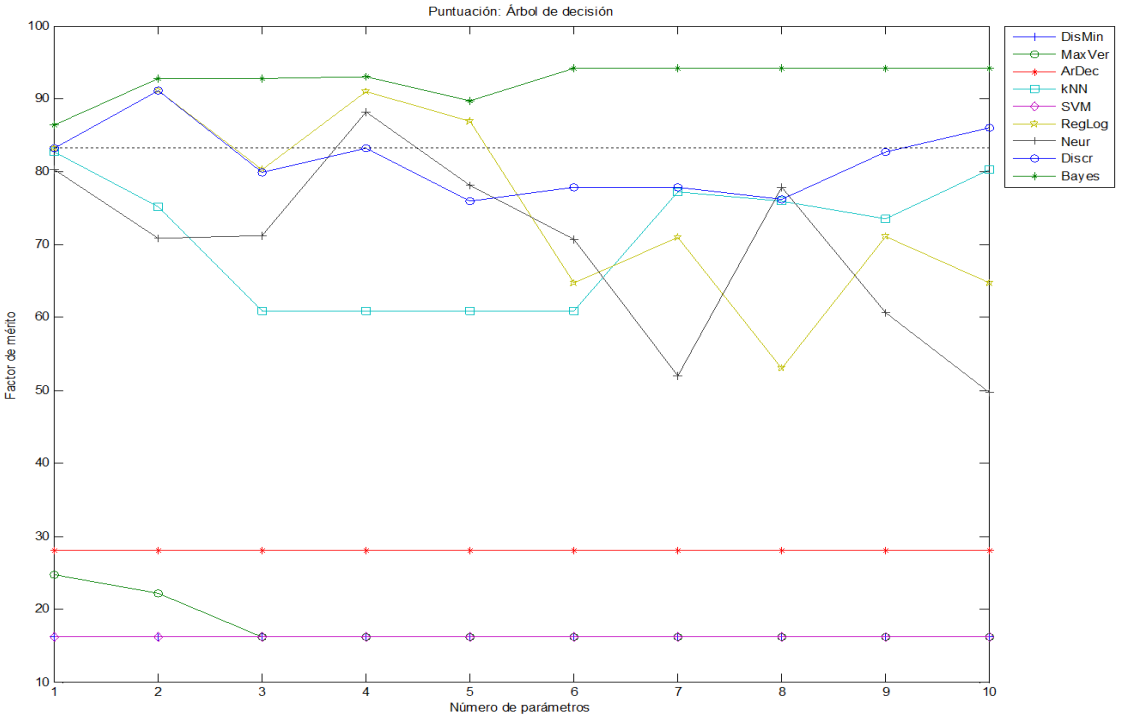


Figura 6-24 Factor de mérito frente al número de parámetros para $\pi = 25\%$

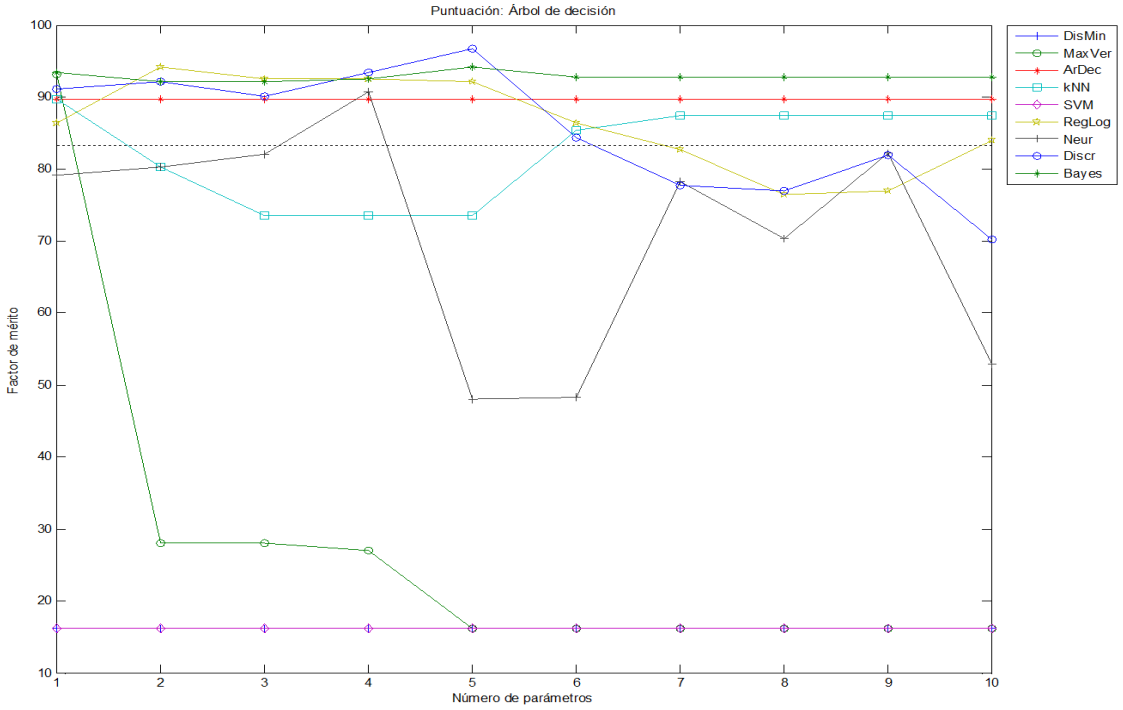


Figura 6-25 Factor de mérito frente al número de parámetros para $\pi = 50\%$

Del análisis de las gráficas anteriores se puede deducir que, si bien es posible usar un porcentaje de un 25% de archivos patrón (π=25%), esta elección tiene varios inconvenientes:

- Limita el número de algoritmos que mejoran el conteo.
- Reduce el valor de mejora sobre el conteo.
- Estrecha el rango de valores en el número de parámetros que mejoran el conteo.

Por todo ello, se trabajará con un 50% de archivos patrón (π=50%) que, si bien en términos relativos puede parecer elevado, se corresponde tan sólo con 32 patrones. Y hecha esta elección, se puede observar cómo para el caso de 5 parámetros las prestaciones de la clasificación son muy adecuadas con diversos algoritmos. Se toma por tanto la decisión de reducir la dimensionalidad a 5 parámetros o, lo que es lo mismo, realizar la clasificación paramétrica de series derivadas en un espacio $\mathbb{R}^{3\cdot5}$.

Una vez establecido el número de archivos patrón ($\pi = 50\%$) y el número de parámetros más significativos (5), se realizarán clasificaciones no secuenciales, usando los 9 algoritmos de clasificación utilizados en el capítulo 4, a las 9 series derivadas obtenidas. A continuación se recogen un resumen de los resultados de la clasificación para cada tipo de serie derivada.

6.2.3.1. Clasificación paramétrica de series derivadas a partir de la distancia mínima

La Figura 6-26 refleja de forma gráfica la comparación de los distintos algoritmos. La altura de la barra refleja la tasa global de éxito para cada clasificador. Los puntos, con el código de colores habitual, indican la tasa de éxito para cada tipo de canto.

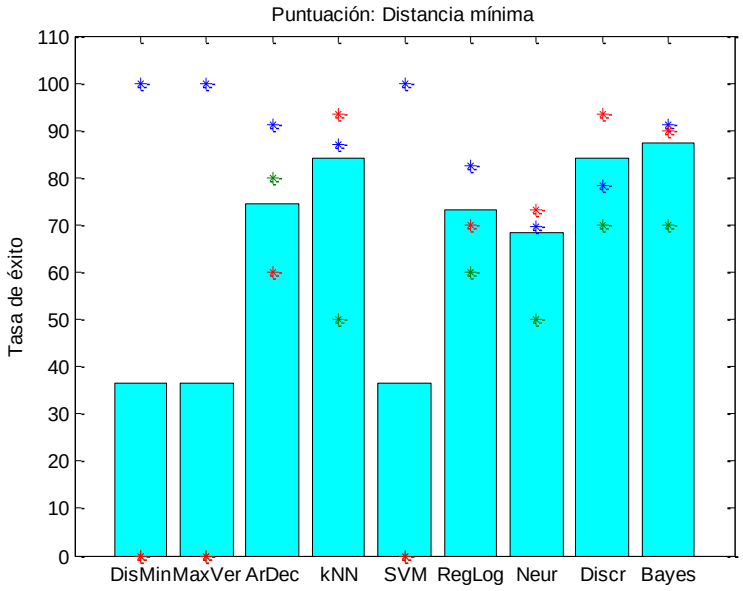


Figura 6-26 Resultados de la clasificación paramétrica (serie derivada: distancia mínima)

La Figura 6-27 refleja el mérito de cada técnica de clasificación mediante un gráfico que enfrenta el rango de la tasa de error frente la tasa de error.

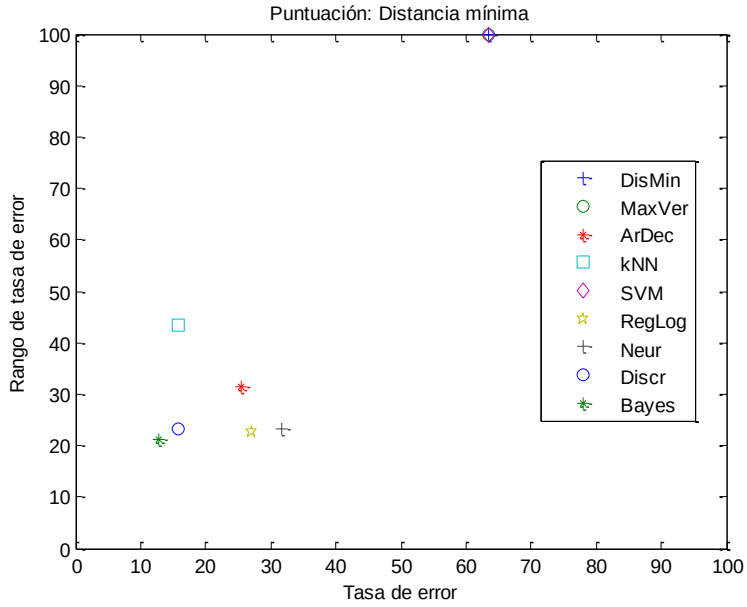


Figura 6-27 Tasa de error y su rango (unidades en %) para clasificación paramétrica (serie derivada: distancia mínima)

Por último, la Tabla 5-15 recoge el factor de mérito de cada uno de los clasificadores utilizados. Se puede ver que el mejor de todos ellos es el clasificador bayesiano, seguido de cerca por la función discriminante.

Algoritmo	Acierto	Errores	Rango	Distancia origen	Mérito
Distancia mínima	36.51%	63.49%	100%	1.18	16.24%
Máxima verosimilitud	36.51%	63.49%	100%	1.18	16.24%
Árboles de decisión	74.60%	25.40%	31%	0.40	71.50%
k-vecinos más próximos	84.13%	15.87%	43%	0.46	67.37%
SVM	36.51%	63.49%	100%	1.18	16.24%
Regresión logística	73.02%	26.98%	23%	0.35	75.11%
Redes neuronales	68.25%	31.75%	23%	0.39	72.14%
Función discriminante	84.13%	15.87%	23%	0.28	80.05%
Clasificador bayesiano	87.30%	12.70%	21%	0.25	82.46%

Tabla 6-4 Factor de mérito para clasificación paramétrica (serie derivada: distancia mínima)

6.2.3.2. Clasificación paramétrica de series derivadas a partir de la máxima verosimilitud

La Figura 6-28 refleja de forma gráfica la comparación de los distintos algoritmos. La altura de la barra refleja la tasa global de éxito para cada clasificador. Los puntos, con el código de colores habitual, indican la tasa de éxito para cada tipo de canto.

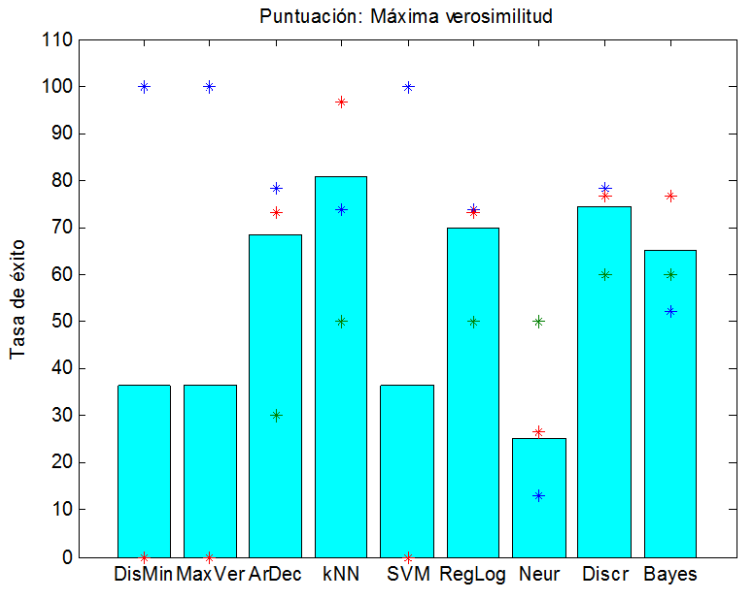


Figura 6-28 Resultados de la clasificación paramétrica (serie derivada: máxima verosimilitud)

La Figura 6-29 refleja el mérito de cada técnica de clasificación mediante un gráfico que enfrenta el rango de la tasa de error frente la tasa de error.

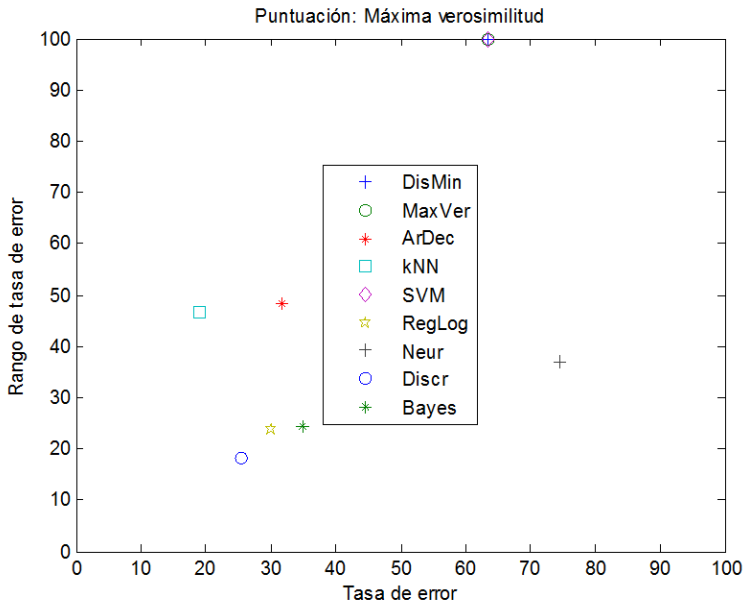


Figura 6-29 Tasa de error y su rango (unidades en %) para clasificación paramétrica (serie derivada: máxima verosimilitud)

Por último, la Tabla 6-5 recoge el factor de mérito de cada uno de los clasificadores utilizados. Se puede ver que el mejor de todos ellos es la función discriminante, seguida de cerca por la regresión logística.

Algoritmo	Acierto	Errores	Rango	Distancia origen	Mérito
Distancia mínima	36.51%	63.49%	100%	1.18	16.24%
Máxima verosimilitud	36.51%	63.49%	100%	1.18	16.24%
Árboles de decisión	68.25%	31.75%	48%	0.58	59.15%
k-vecinos más próximos	80.95%	19.05%	47%	0.50	64.36%
SVM	36.51%	63.49%	100%	1.18	16.24%
Regresión logística	69.84%	30.16%	24%	0.38	72.78%
Redes neuronales	25.40%	74.60%	37%	0.83	41.13%
Función discriminante	74.60%	25.40%	18%	0.31	77.88%
Clasificador bayesiano	65.08%	34.92%	24%	0.43	69.84%

Tabla 6-5 Factor de mérito para clasificación paramétrica (serie derivada: máxima verosimilitud)

6.2.3.3. Clasificación paramétrica de series derivadas a partir del árbol de decisión

La Figura 6-30 refleja de forma gráfica la comparación de los distintos algoritmos. La altura de la barra refleja la tasa global de éxito para cada clasificador. Los puntos, con el código de colores habitual, indican la tasa de éxito para cada tipo de canto.

La Figura 6-31 refleja el mérito de cada técnica de clasificación mediante un gráfico que enfrenta el rango de la tasa de error frente la tasa de error.

Por último, la Tabla 6-6 recoge el factor de mérito de cada uno de los clasificadores utilizados. Se puede ver que el mejor de todos ellos es la función discriminante, seguida de cerca por el clasificador bayesiano.

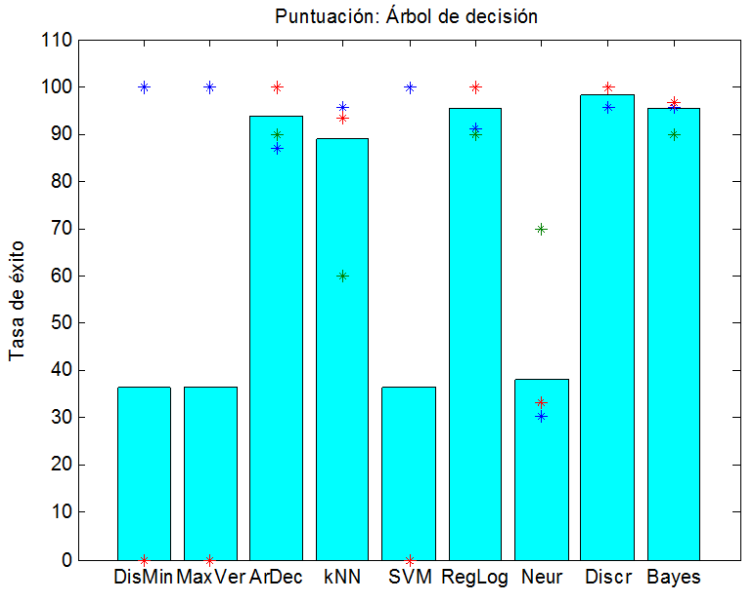


Figura 6-30 Resultados de la clasificación paramétrica (serie derivada: árbol de decisión)

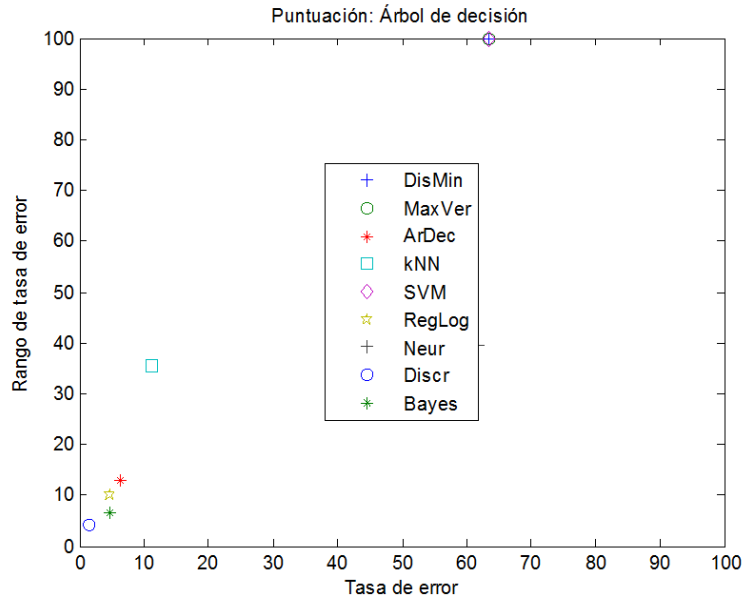


Figura 6-31 Tasa de error y su rango (unidades en %) para clasificación paramétrica (serie derivada: árbol de decisión)

Algoritmo	Acierto	Errores	Rango	Distancia origen	Mérito
Distancia mínima	36.51%	63.49%	100%	1.18	16.24%
Máxima verosimilitud	36.51%	63.49%	100%	1.18	16.24%
Árboles de decisión	93.65%	6.35%	13%	0.15	89.74%
k-vecinos más próximos	88.89%	11.11%	36%	0.37	73.59%
SVM	36.51%	63.49%	100%	1.18	16.24%
Regresión logística	95.24%	4.76%	10%	0.11	92.17%
Redes neuronales	38.10%	61.90%	40%	0.73	48.05%
Función discriminante	98.41%	1.59%	4%	0.05	96.73%
Clasificador bayesiano	95.24%	4.76%	7%	0.08	94.21%

Tabla 6-6 Factor de mérito para clasificación paramétrica (serie derivada: árbol de decisión)

6.2.3.4. Clasificación paramétrica de series derivadas a partir de los *k*-vecinos más próximos

La Figura 6-32 refleja de forma gráfica la comparación de los distintos algoritmos. La altura de la barra refleja la tasa global de éxito para cada clasificador. Los puntos, con el código de colores habitual, indican la tasa de éxito para cada tipo de canto.

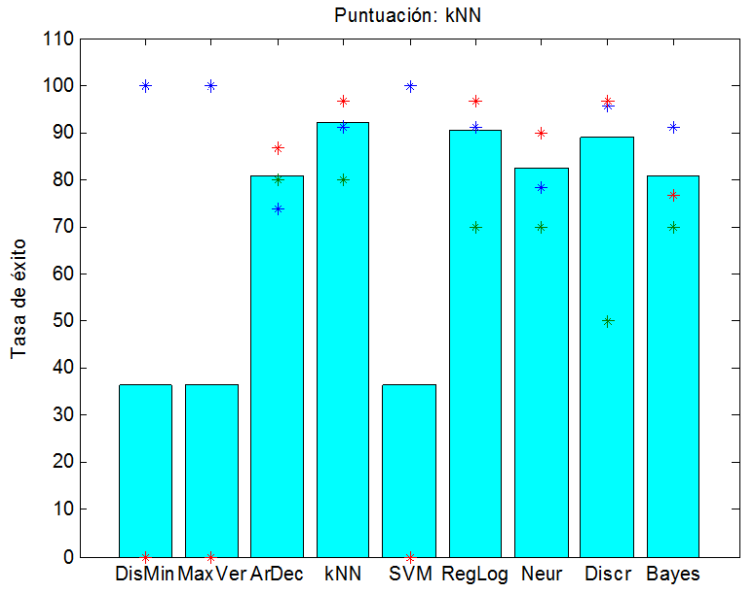


Figura 6-32 Resultados de la clasificación paramétrica (serie derivada: k-vecinos más próximos)

La Figura 6-33 refleja el mérito de cada técnica de clasificación mediante un gráfico que enfrenta el rango de la tasa de error frente la tasa de error.

Por último, la Tabla 6-7 recoge el factor de mérito de cada uno de los clasificadores utilizados. Se puede ver que el mejor de todos ellos es el algoritmo de los *k*-vecinos más próximos, seguido de cerca por los árboles de decisión.

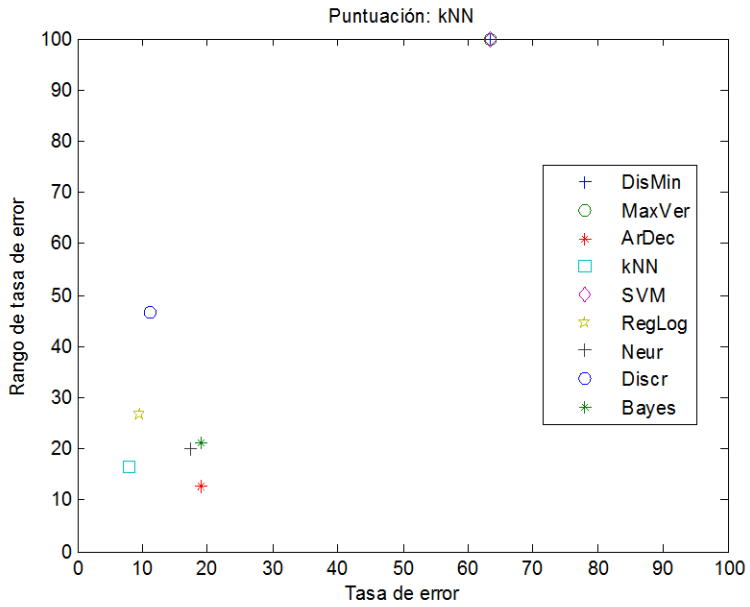


Figura 6-33 Tasa de error y su rango para clasificación paramétrica (serie derivada: k-vecinos más próximos)

Algoritmo	Acierto	Errores	Rango	Distancia origen	Mérito
Distancia mínima	36.51%	63.49%	100%	1.18	16.24%
Máxima verosimilitud	36.51%	63.49%	100%	1.18	16.24%
Árboles de decisión	80.95%	19.05%	13%	0.23	83.79%
k-vecinos más próximos	92.06%	7.94%	17%	0.18	86.95%
SVM	36.51%	63.49%	100%	1.18	16.24%
Regresión logística	90.48%	9.52%	27%	0.28	79.98%
Redes neuronales	82.54%	17.46%	20%	0.27	81.23%
Función discriminante	88.89%	11.11%	47%	0.48	66.08%
Clasificador bayesiano	80.95%	19.05%	21%	0.29	79.79%

Tabla 6-7 Factor de mérito para clasificación paramétrica (serie derivada: k-vecinos más próximos)

6.2.3.5. Clasificación paramétrica de series derivadas a partir de SVM

La Figura 6-34 refleja de forma gráfica la comparación de los distintos algoritmos. La altura de la barra refleja la tasa global de éxito para cada clasificador. Los puntos, con el código de colores habitual, indican la tasa de éxito para cada tipo de canto.

La Figura 6-35 refleja el mérito de cada técnica de clasificación mediante un gráfico que enfrenta el rango de la tasa de error frente la tasa de error.

Por último, la Tabla 6-8 recoge el factor de mérito de cada uno de los clasificadores utilizados. Se puede ver que el mejor de todos ellos son los árboles de decisión, seguido de cerca por los *k*-vecinos más próximos.

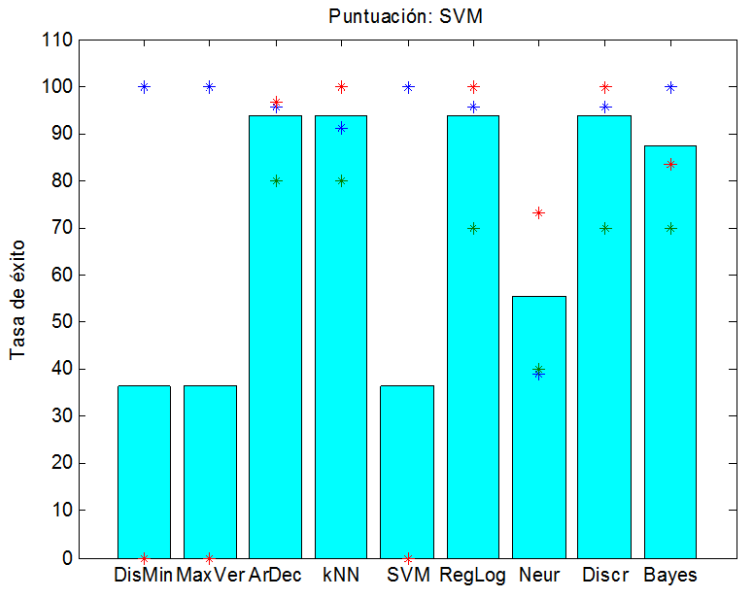


Figura 6-34 Resultados de la clasificación paramétrica (serie derivada: SVM)

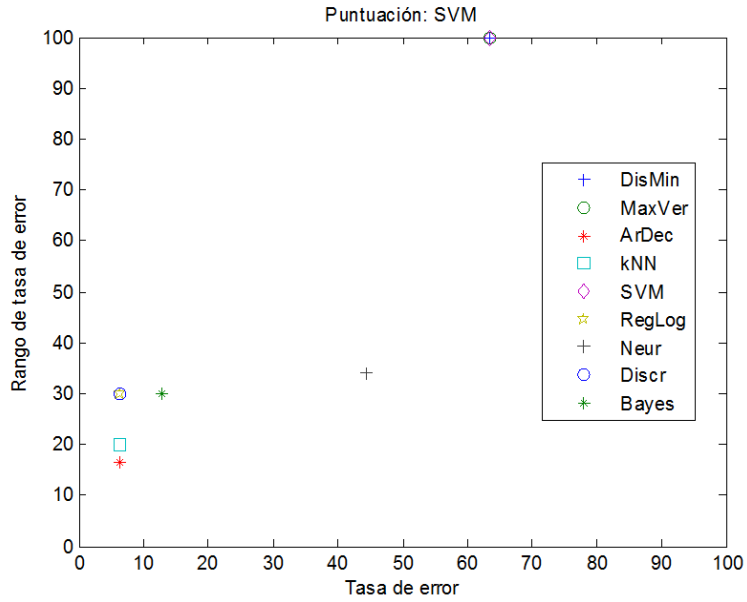


Figura 6-35 Tasa de error y su rango (unidades en %) para clasificación paramétrica (serie derivada: SVM)

Algoritmo	Acierto	Errores	Rango	Distancia origen	Mérito
Distancia mínima	36.51%	63.49%	100%	1.18	16.24%
Máxima verosimilitud	36.51%	63.49%	100%	1.18	16.24%
Árboles de decisión	93.65%	6.35%	17%	0.18	87.39%
k-vecinos más próximos	93.65%	6.35%	20%	0.21	85.16%
SVM	36.51%	63.49%	100%	1.18	16.24%
Regresión logística	93.65%	6.35%	30%	0.31	78.32%
Redes neuronales	55.56%	44.44%	34%	0.56	60.34%
Función discriminante	93.65%	6.35%	30%	0.31	78.32%
Clasificador bayesiano	87.30%	12.70%	30%	0.33	76.96%

Tabla 6-8 Factor de mérito para clasificación paramétrica (serie derivada: SVM)

6.2.3.6. Clasificación paramétrica de series derivadas a partir de regresión logística

La Figura 6-36 refleja de forma gráfica la comparación de los distintos algoritmos. La altura de la barra refleja la tasa global de éxito para cada clasificador. Los puntos, con el código de colores habitual, indican la tasa de éxito para cada tipo de canto.

La Figura 6-37 refleja el mérito de cada técnica de clasificación mediante un gráfico que enfrenta el rango de la tasa de error frente la tasa de error.

Por último, la Tabla 6-9 recoge el factor de mérito de cada uno de los clasificadores utilizados. Se puede ver que el mejor de todos ellos la función discriminante.

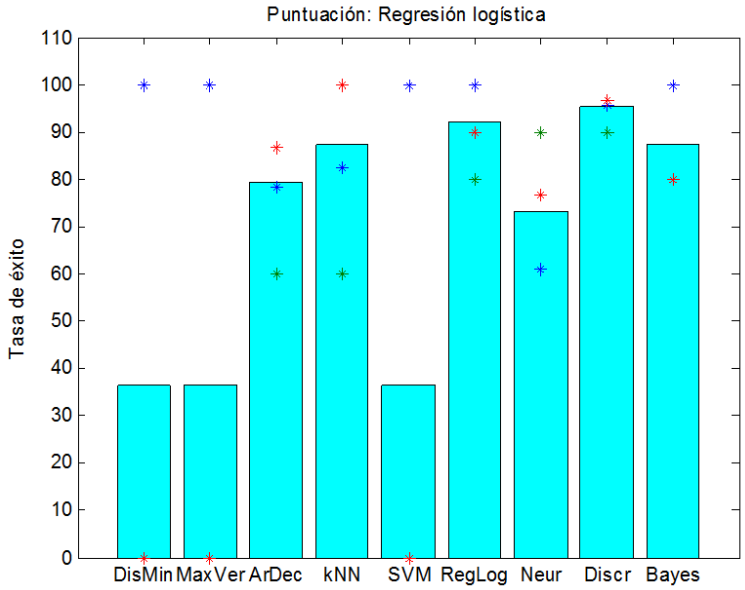


Figura 6-36 Resultados de la clasificación paramétrica (serie derivada: regresión logística)

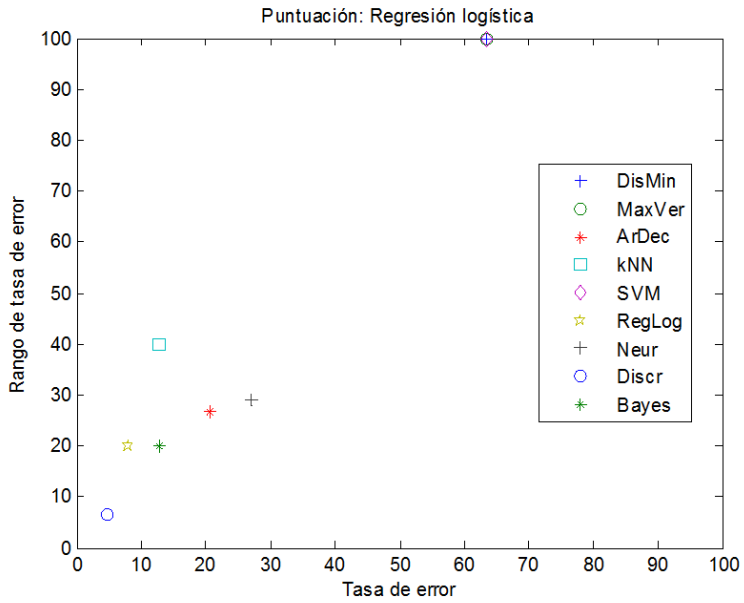


Figura 6-37 Tasa de error y su rango (unidades en %) para clasificación paramétrica (serie derivada: regresión logística)

Algoritmo	Acierto (%)	Errores (%)	Rango (%)	Distancia origen	Mérito (%)
Distancia mínima	36.51%	63.49%	100%	1.18	16.24%
Máxima verosimilitud	36.51%	63.49%	100%	1.18	16.24%
Árboles de decisión	79.37%	20.63%	27%	0.34	76.16%
k-vecinos más próximos	87.30%	12.70%	40%	0.42	70.32%
SVM	36.51%	63.49%	100%	1.18	16.24%
Regresión logística	92.06%	7.94%	20%	0.22	84.79%
Redes neuronales	73.02%	26.98%	29%	0.40	71.92%
Función discriminante	95.24%	4.76%	7%	0.08	94.21%
Clasificador bayesiano	87.30%	12.70%	20%	0.24	83.25%

Tabla 6-9 Factor de mérito para clasificación paramétrica (serie derivada: regresión logística)

6.2.3.7. Clasificación paramétrica de series derivadas a partir de red neuronal

La Figura 6-38 refleja de forma gráfica la comparación de los distintos algoritmos. La altura de la barra refleja la tasa global de éxito para cada clasificador. Los puntos, con el código de colores habitual, indican la tasa de éxito para cada tipo de canto.

La Figura 6-39 refleja el mérito de cada técnica de clasificación mediante un gráfico que enfrenta el rango de la tasa de error frente la tasa de error.

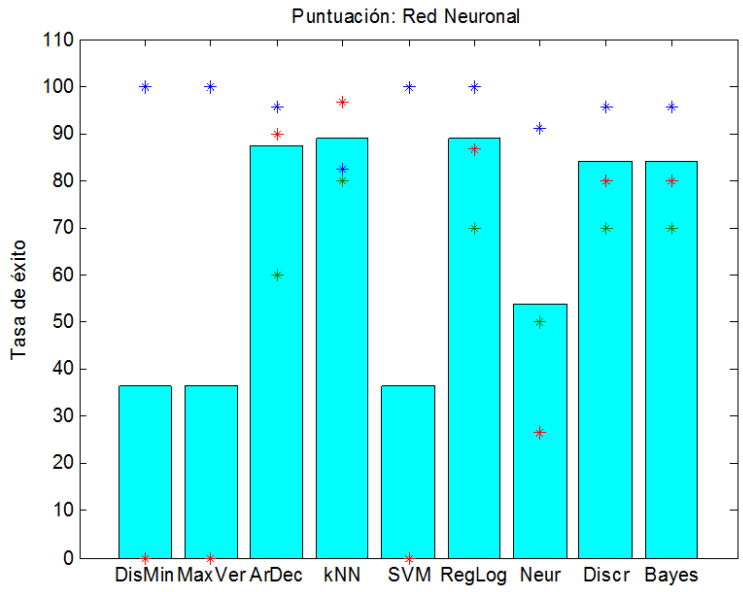


Figura 6-38 Resultados de la clasificación paramétrica (serie derivada: red neuronal)

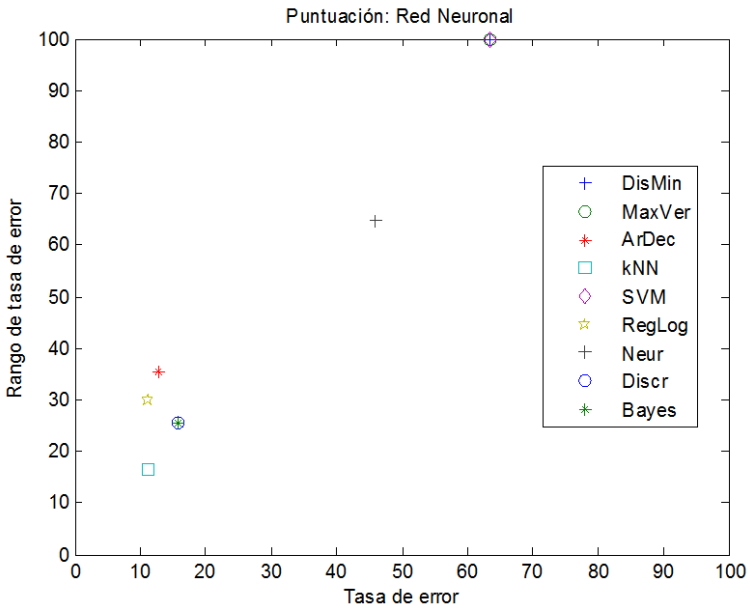


Figura 6-39 Tasa de error y su rango (unidades en %) para clasificación paramétrica (serie derivada: red neuronal)

Por último, la Tabla 6-10 recoge el factor de mérito de cada uno de los clasificadores utilizados. Se puede ver que el mejor de todos ellos es el algoritmo de los *k*-vecinos más próximos.

Algoritmo	Acierto	Errores	Rango	Distancia origen	Mérito
Distancia mínima	36.51%	63.49%	100%	1.18	16.24%
Máxima verosimilitud	36.51%	63.49%	100%	1.18	16.24%
Árboles de decisión	87.30%	12.70%	36%	0.38	73.24%
k-vecinos más próximos	88.89%	11.11%	17%	0.20	85.84%
SVM	36.51%	63.49%	100%	1.18	16.24%
Regresión logística	88.89%	11.11%	30%	0.32	77.38%
Redes neuronales	53.97%	46.03%	65%	0.79	43.89%
Función discriminante	84.13%	15.87%	26%	0.30	78.67%
Clasificador bayesiano	84.13%	15.87%	26%	0.30	78.67%

Tabla 6-10 Factor de mérito para clasificación paramétrica (serie derivada: red neuronal)

6.2.3.8. Clasificación paramétrica de series derivadas a partir de función discriminante

La Figura 6-40 refleja de forma gráfica la comparación de los distintos algoritmos. La altura de la barra refleja la tasa global de éxito para cada clasificador. Los puntos, con el código de colores habitual, indican la tasa de éxito para cada tipo de canto.

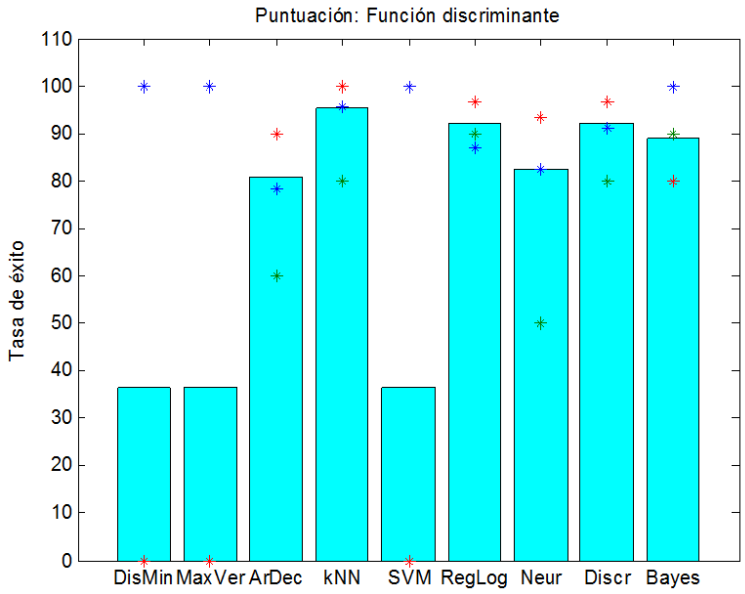


Figura 6-40 Resultados de la clasificación paramétrica (serie derivada: función discriminante)

La Figura 6-41 refleja el mérito de cada técnica de clasificación mediante un gráfico que enfrenta el rango de la tasa de error frente la tasa de error.

Por último, la Tabla 6-11 recoge el factor de mérito de cada uno de los clasificadores utilizados. Se puede ver que el mejor de todos ellos es la regresión logística.

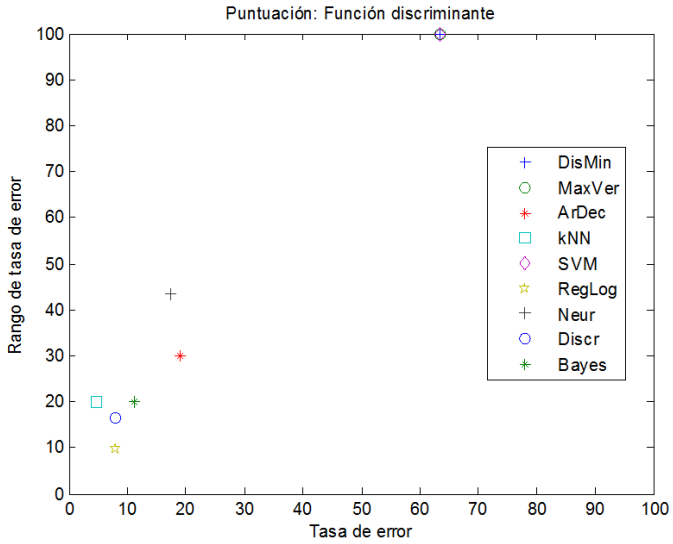


Figura 6-41 Tasa de error y su rango (unidades en %) para clasificación paramétrica (serie derivada: función discriminante)

Algoritmo	Acierto	Errores	Rango	Distancia origen	Mérito
Distancia mínima	36.51%	63.49%	100%	1.18	16.24%
Máxima verosimilitud	36.51%	63.49%	100%	1.18	16.24%
Árboles de decisión	80.95%	19.05%	30%	0.36	74.87%
k-vecinos más próximos	95.24%	4.76%	20%	0.21	85.46%
SVM	36.51%	63.49%	100%	1.18	16.24%
Regresión logística	92.06%	7.94%	10%	0.13	91.13%
Redes neuronales	82.54%	17.46%	43%	0.47	66.96%
Función discriminante	92.06%	7.94%	17%	0.18	86.95%
Clasificador bayesiano	88.89%	11.11%	20%	0.23	83.82%

Tabla 6-11 Factor de mérito para clasificación paramétrica (serie derivada: función discriminante)

6.2.3.9. Clasificación paramétrica de series derivadas a partir del clasificador bayesiano

La Figura 6-42 refleja de forma gráfica la comparación de los distintos algoritmos. La altura de la barra refleja la tasa global de éxito para cada clasificador. Los puntos, con el código de colores habitual, indican la tasa de éxito para cada tipo de canto.

La Figura 6-43 refleja el mérito de cada técnica de clasificación mediante un gráfico que enfrenta el rango de la tasa de error frente la tasa de error.

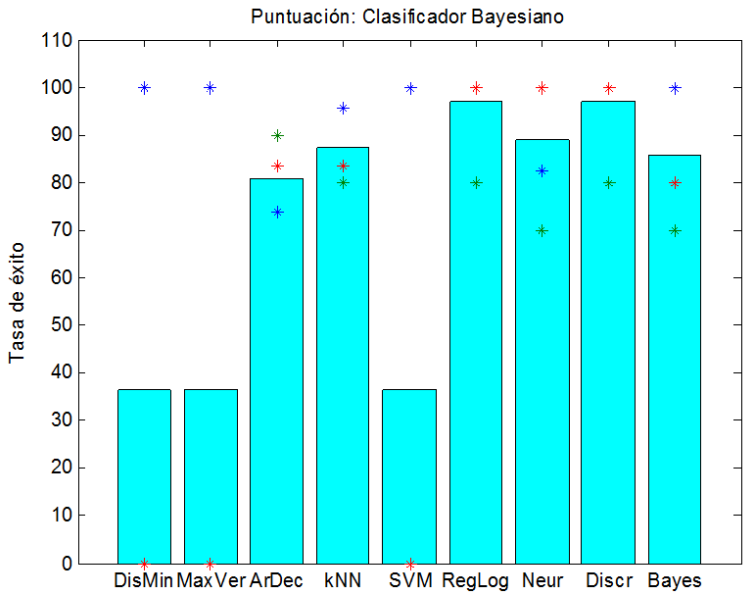


Figura 6-42 Resultados de la clasificación paramétrica (serie derivada: clasificador bayesiano)

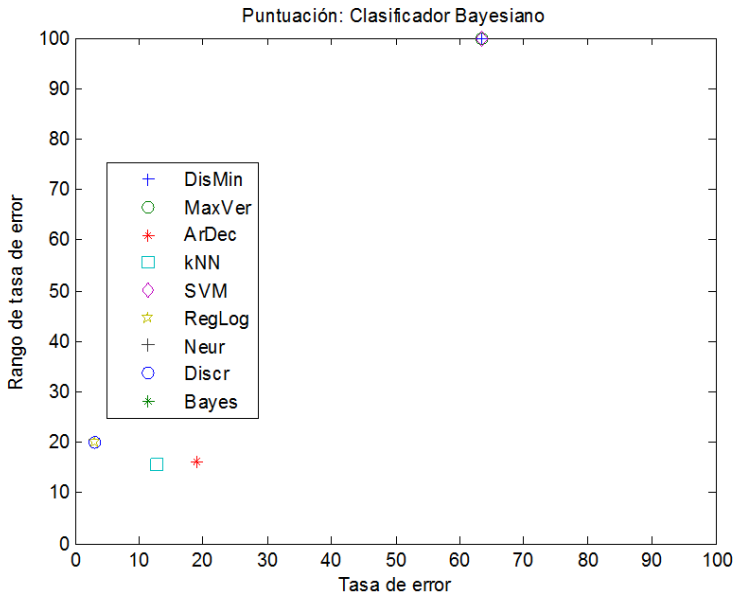


Figura 6-43 Tasa de error y su rango (unidades en %) para clasificación paramétrica (serie derivada: clasificador bayesiano)

Por último, la Tabla 6-12 recoge el factor de mérito de cada uno de los clasificadores utilizados. Se puede ver que los mejores resultados se obtienen de los algoritmos *k*-vecinos más próximos, regresión logística y función discriminante.

Algoritmo	Acierto	Errores	Rango	Distancia origen	Mérito
Distancia mínima	36.51%	63.49%	100%	1.18	16.24%
Máxima verosimilitud	36.51%	63.49%	100%	1.18	16.24%
Árboles de decisión	80.95%	19.05%	16%	0.25	82.37%
k-vecinos más próximos	87.30%	12.70%	16%	0.20	85.75%
SVM	36.51%	63.49%	100%	1.18	16.24%
Regresión logística	96.83%	3.17%	20%	0.20	85.68%
Redes neuronales	88.89%	11.11%	30%	0.32	77.38%
Función discriminante	96.83%	3.17%	20%	0.20	85.68%
Clasificador bayesiano	85.71%	14.29%	30%	0.33	76.50%

Tabla 6-12 Factor de mérito para clasificación paramétrica (serie derivada: clasificador bayesiano)

6.2.3.10. Comparación de resultados de la clasificación paramétrica de series derivadas

Una vez realizado el estudio de cada algoritmo sobre las distintas series derivadas se realiza una comparación de los resultados. Para este fin, la Tabla 6-13 ofrece un resumen de los resultados haciendo uso del factor de mérito.

Cada columna corresponde con una serie derivada, es decir, una serie con las puntuaciones obtenidas mediante la clasificación de *frames* de acuerdo con los algoritmos siguientes:

- DM: Distancia mínima
- MV: Máxima verosimilitud
- AD: Árboles de decisión
- kNN: k-vecinos más próximos
- SVM: Máquinas de vectores soporte
- RL: Regresión logística
- RN: Redes neuronales
- FD: Función discriminante
- CB: Clasificador bayesiano

Cada fila corresponde al tipo de clasificador aplicado a la serie derivada, donde se utilizan los mismos códigos anteriores. La última fila se corresponde con el factor de mérito obtenido mediante la clasificación por conteo.

Para comparar mejor la clasificación paramétrica con la clasificación por conteo, en la Tabla 5-14 se recoge el factor de mérito diferencial con respecto al valor por conteo.

		Series derivadas								
		DM	MV	AD	kNN	SVM	RL	RN	FD	CB
Clasificador	DM	16.24	16.24	16.24	16.24	16.24	16.24	16.24	16.24	16.24
	MV	16.24	16.24	16.24	16.24	16.24	16.24	16.24	16.24	16.24
	AD	71.50	59.15	89.74	83.79	87.39	76.16	73.24	74.87	82.37
	kNN	67.37	64.36	73.59	86.95	85.16	70.32	85.84	85.46	85.75
	SVM	16.24	16.24	16.24	16.24	16.24	16.24	16.24	16.24	16.24
	RL	75.11	72.78	92.17	79.98	78.32	84.79	77.38	91.13	85.68
	RN	72.14	41.13	48.05	81.23	60.34	71.92	43.89	66.96	77.38
	FD	80.05	77.88	96.73	66.08	78.32	94.21	78.67	86.95	85.68
	CB	82.46	69.84	94.21	79.79	76.96	83.25	78.67	83.82	76.50
Conteo		16.24	63.90	83.20	46.54	35.83	52.93	53.01	23.87	66.06

Tabla 6-13 Factor de mérito para clasificación paramétrica (valores en %)

		Series derivadas								
		DM	MV	AD	kNN	SVM	RL	RN	FD	CB
Clasificador	DM	0.00	-47.66	-66.96	-30.30	-19.59	-36.69	-36.77	-7.63	-49.82
	MV	0.00	-47.66	-66.96	-30.30	-19.59	-36.69	-36.77	-7.63	-49.82
	AD	55.26	-4.75	6.54	37.25	51.56	23.23	20.23	51.00	16.31
	kNN	51.13	0.46	-9.61	40.41	49.33	17.39	32.83	61.59	19.69
	SVM	0.00	-47.66	-66.96	-30.30	-19.59	-36.69	-36.77	-7.63	-49.82
	RL	58.87	8.88	8.97	33.44	42.49	31.86	24.37	67.26	19.62
	RN	55.90	-22.77	-35.15	34.69	24.51	18.99	-9.12	43.09	11.32
	FD	63.81	13.98	13.53	19.54	42.49	41.28	25.66	63.08	19.62
	CB	66.22	5.94	11.01	33.25	41.13	30.32	25.66	59.95	10.44
Conteo		16.24	63.90	83.20	46.54	35.83	52.93	53.01	23.87	66.06

Tabla 6-14 Factor de mérito para clasificación paramétrica: diferencial con el conteo (valores en %)

El valor correspondiente a la tasa de error de clasificación se recoge en la Tabla 6-15. Igualmente la Tabla 6-16 recoge la tasa de error diferencial con respecto al valor por conteo.

		Series derivadas								
		DM	MV	AD	kNN	SVM	RL	RN	FD	CB
Clasificador	DM	63.49	63.49	63.49	63.49	63.49	63.49	63.49	63.49	63.49
	MV	63.49	63.49	63.49	63.49	63.49	63.49	63.49	63.49	63.49
	AD	25.40	31.75	6.35	19.05	6.35	20.63	12.70	19.05	19.05
	kNN	15.87	19.05	11.11	7.94	6.35	12.70	11.11	4.76	12.70
	SVM	63.49	63.49	63.49	63.49	63.49	63.49	63.49	63.49	63.49
	RL	26.98	30.16	4.76	9.52	6.35	7.94	11.11	7.94	3.17
	RN	31.75	74.60	61.90	17.46	44.44	26.98	46.03	17.46	11.11
	FD	15.87	25.40	1.59	11.11	6.35	4.76	15.87	7.94	3.17
	CB	12.70	34.92	4.76	19.05	12.70	12.70	15.87	11.11	14.29
Conteo		63.49	26.98	14.29	28.57	42.86	34.92	28.57	63.49	20.63

Tabla 6-15 Tasa de error para clasificación paramétrica (valores en %)

Clasificador	Series derivadas								
	DM	MV	AD	kNN	SVM	RL	RN	FD	CB
DM	0.00	-36.51	-49.20	-34.92	-20.63	-28.57	-34.92	0.00	-42.86
MV	0.00	-36.51	-49.20	-34.92	-20.63	-28.57	-34.92	0.00	-42.86
AD	38.09	-4.77	7.94	9.52	36.51	14.29	15.87	44.44	1.58
kNN	47.62	7.93	3.18	20.63	36.51	22.22	17.46	58.73	7.93
SVM	0.00	-36.51	-49.20	-34.92	-20.63	-28.57	-34.92	0.00	-42.86
RL	36.51	-3.18	9.53	19.05	36.51	26.98	17.46	55.55	17.46
RN	31.74	-47.62	-47.61	11.11	-1.58	7.94	-17.46	46.03	9.52
FD	47.62	1.58	12.70	17.46	36.51	30.16	12.70	55.55	17.46
CB	50.79	-7.94	9.53	9.52	30.16	22.22	12.70	52.38	6.34
Conteo	63.49	26.98	14.29	28.57	42.86	34.92	28.57	63.49	20.63

Tabla 6-16 Tasa de error para clasificación paramétrica: diferencial con el conteo (valores en %)

De las tablas anteriores se deduce que la mejor opción es clasificar los *frames* con árboles de decisión (serie derivada: árbol de decisión) y clasificar luego las series derivadas mediante una función discriminante. Con esta combinación se obtiene una tasa de error del 1.59% y un factor de mérito de 96.73%, con casi 13 y 14 puntos de mejora sobre la clasificación por conteo respectivamente.

La Tabla 6-17 recoge los valores de los indicadores de exactitud, precisión, sensibilidad, especificidad y tasa de errores para los distintos algoritmos usados en las series derivadas creadas por árboles de decisión. Se puede ver que el mejor método de clasificación paramétrica para la serie derivada del árbol de decisión sigue siendo la función discriminante.

Algoritmo	Exactitud	Tasa de errores	Precisión	Sensib.	Especif.
Distancia mínima	57.67%	42.33%	-	33.33%	66.67%
Máxima verosimilitud	57.67%	42.33%	-	33.33%	66.67%
Árboles de decisión	95.77%	4.23%	93.00%	92.32%	96.52%
k-vecinos más próximos	92.59%	7.41%	85.66%	83.00%	94.05%
SVM	57.67%	42.33%	-	33.33%	66.67%
Regresión logística	96.83%	3.17%	92.42%	93.77%	97.91%
Redes neuronales	58.73%	41.27%	54.81%	44.59%	73.71%
Función discriminante	98.94%	1.06%	96.97%	98.55%	99.37%
Clasificador bayesiano	96.83%	3.17%	92.49%	94.11%	97.91%

Tabla 6-17 Indicadores para la evaluación de clasificación paramétrica (serie derivada: árbol de decisión)

En la Figura 6-44 se representa el análisis ROC de los distintos algoritmos estudiados donde de nuevo el mejor clasificador es la función discriminante.

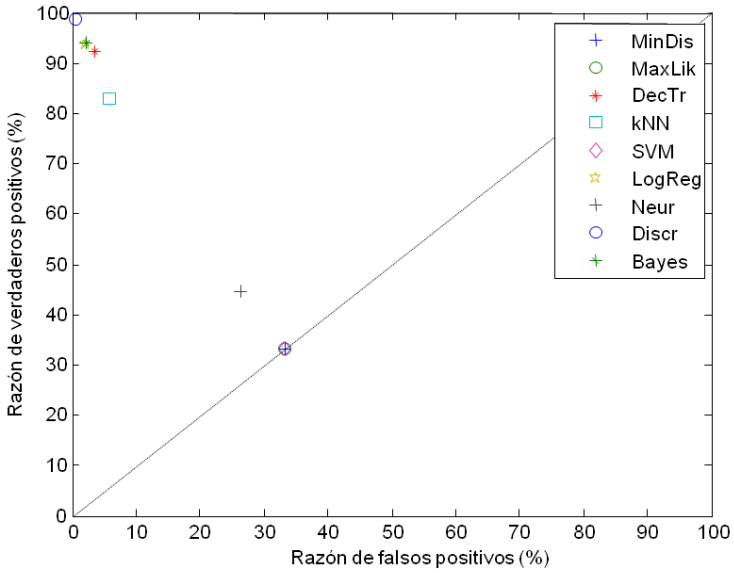


Figura 6-44 Clasificación por combinación árbol de decisión-función discriminante

En la Figura 6-45 se representa la concordancia de los resultados usando los coeficientes kappa de Cohen.

La conclusión tras analizar las distintas comparativas realizadas, es que la clasificación por función discriminante ofrece los mejores resultados en la clasificación paramétrica sobre series derivadas creadas a partir de árboles de decisión.

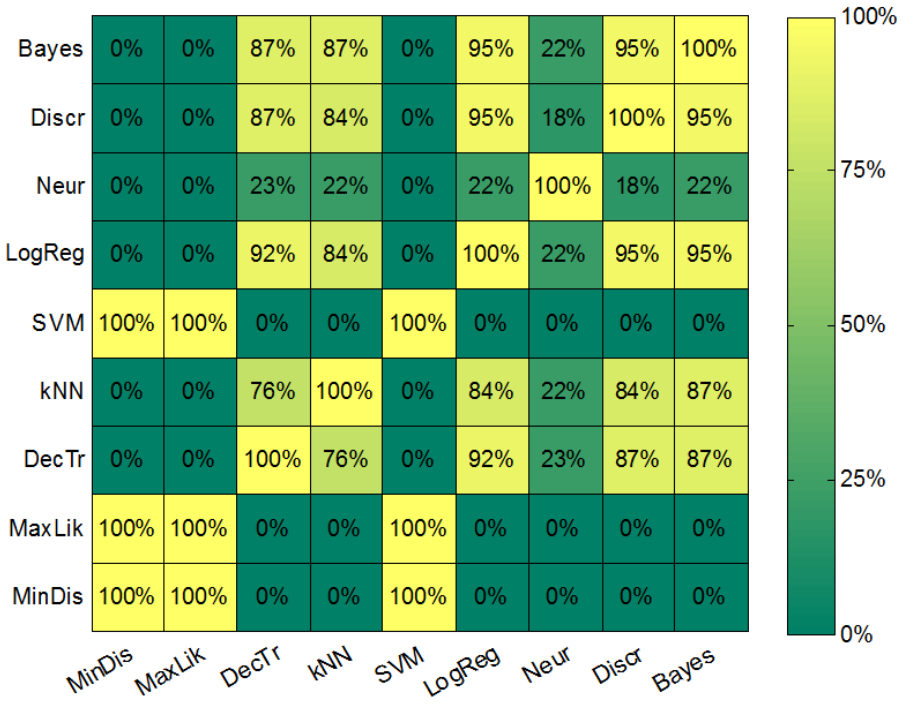


Figura 6-45 Comparación de los métodos de clasificación paramétrica mediantes coeficientes kappa de Cohen

La Figura 6-46 muestra el resultado de la clasificación mediante árbol de decisión-función discriminante, aplicado al conjunto de archivos de sonido disponibles. El

resultado global de la clasificación puede resumirse en la Figura 6-47. La Tabla 6-18 muestra la matriz de confusión de dos formas: conteo del número de archivos clasificados y de forma porcentual.

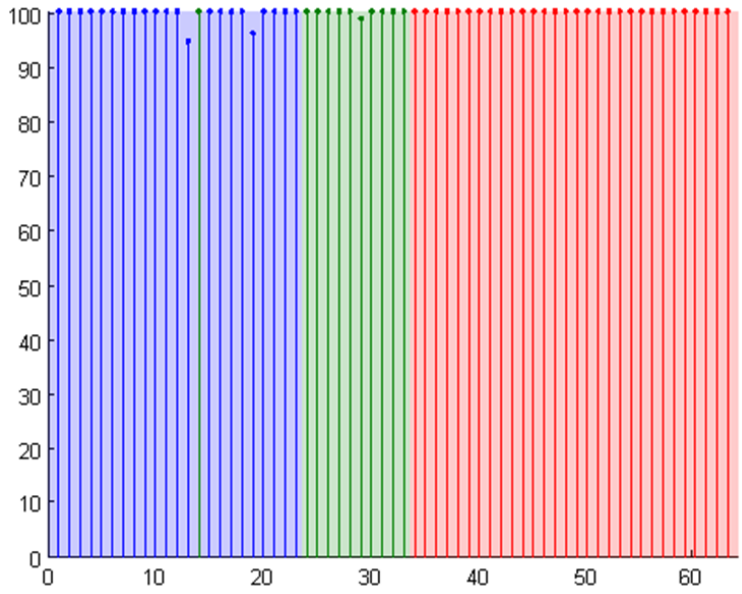


Figura 6-46 Clasificación por combinación árbol de decisión-función discriminante

		Clase obtenida		
		1	2	3
Clase real	1	22	1	0
	2	0	10	0
	3	0	0	30

		Clase obtenida		
		1	2	3
Clase real	1	95.65%	4.35%	0.00%
	2	0.00%	100.00%	0.00%
	3	0.00%	0.00%	100.00%

Tabla 6-18 Matriz de confusión de la clasificación por combinación árbol de decisión-función discriminante

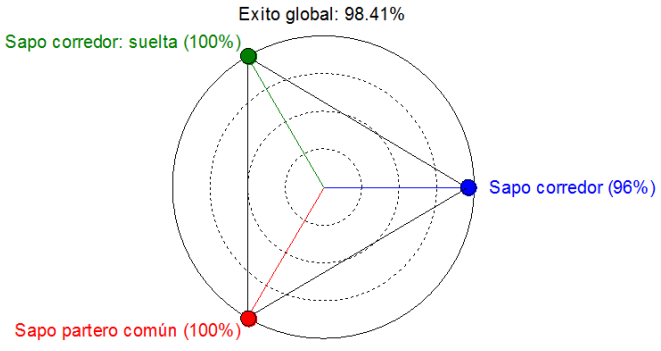


Figura 6-47 Resumen de la clasificación por combinación árbol de decisión-función discriminante

Como puede verse, esta combinación presenta unas excelentes prestaciones de clasificación. La aplicación sobre el conjunto de archivos bajo estudio produce un único error de clasificación.

6.3. Consideraciones de industrialización

A lo largo de esta tesis no sólo se ha realizado una importante aportación desde el punto de vista teórico, sino también se ha buscado una aplicación práctica en los sistemas automatizados de detección y clasificación de anuros.

Lo ideal es que la técnica de clasificación por combinación de árbol de decisión-función discriminante pueda integrarse en una red de sensores distribuidos en campo. Existen distintos ejemplos de estos nodos en la literatura, uno de los cuales se presenta en la Figura 6-48 (Mark E Cambron & Emberton, 2013).

En sistemas más avanzados, cada nodo sensor de la red debe tener capacidad, no sólo de capturar el sonido, sino también de procesarlo localmente, clasificarlo y enviar el resultado a un centro de control vía radio. La arquitectura así descrita constituye pues una Red de Sensores Inalámbricas o WSN (*Wireless Sensor Network*).

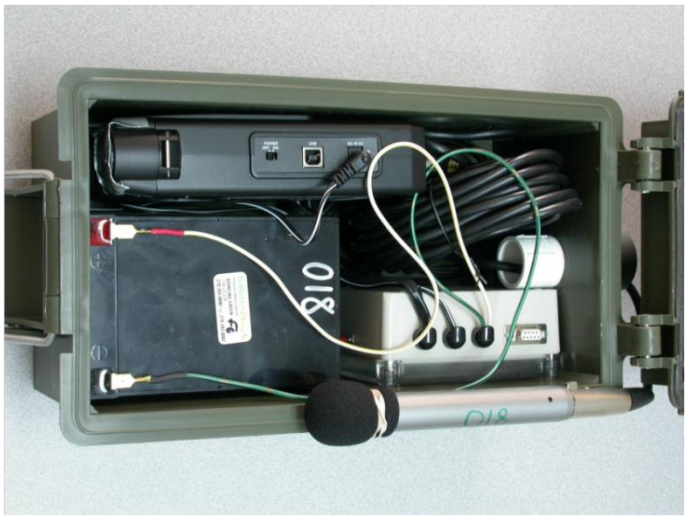


Figura 6-48 Ejemplo de nodo sensor

Precisamente, por el carácter aplicado del trabajo, la solución propuesta debe cumplir unas ciertas condiciones de industrialización. Las primeras de ellas afectan a la plataforma hardware que se elija. Entre estas condiciones destacamos las siguientes:

a) Capacidad de operación a la intemperie. Al estar instalado en campo el nodo debe poder soportar rangos de temperaturas elevados, radiación solar directa, lluvia, humedad, etc.
b) Autonomía elevada. Al ser una red que se puede llegar a desplegar en un amplio territorio es importante que la alimentación de cada nodo, habitualmente mediante baterías, tenga una autonomía elevada para reducir la necesidad de mantenimiento de la red.
c) Bajo consumo. Para aumentar la autonomía es deseable que el consumo del nodo sea lo más bajo posible.

d) Bajo coste. Si se desea cubrir un área geográfica amplia, el número de nodos de la red puede llegar a ser elevado. Por ello es muy importante que el coste de cada uno de los nodos sea reducido.
e) Bajo ancho de banda; proceso local. Los condicionantes anteriores, especialmente los dos últimos, hacen que el sistema de comunicaciones tenga que ser de un bajo ancho de banda. Ello dificulta enormemente la posibilidad de transmitir la señal sonora completa, siendo preferible realizar un proceso de clasificación local y enviar al centro de control la información ya agregada (detección y clasificación del anuro).

Estas condiciones, especialmente las de bajo consumo y bajo coste, hacen que la capacidad de proceso de los nodos sea limitada lo que introduce a su vez restricciones en el algoritmo de clasificación.

Por otra parte la industrialización del sistema impone, o al menos aconseja, algunas condiciones al algoritmo de clasificación. La primera de ellas es la conveniencia de trabajar con estándares. En este sentido el hecho de utilizar una doble clasificación paramétrica (árbol de decisión-clasificador bayesiano) basada en la norma MPEG-7, facilita la comprensión e interoperabilidad de los nodos, incluso si éstos estuviesen construidos por distintos fabricantes o programadores.

Otro aspecto a considerar es que el algoritmo debe ser tolerante ante perturbaciones sonoras. Una característica común a todas las grabaciones es que se realizan en el hábitat natural, por lo que están acompañadas de importantes “ruidos” (viento, agua, lluvia, tráfico, voz,…), lo que supone un desafío adicional en el tratamiento de las señales. Una parte significativa de este ruido se concentra en la zona de bajas frecuencias. Por ello se ha mostrado útil el proceder a realizar un filtrado del sonido. Este filtrado pretende eliminar todas las componentes que no pueden ser significativas, en concreto las de baja y muy alta frecuencia. Para ello se aplica un filtro paso de banda con frecuencias de corte en 300 Hz. y en 10 kHz. Los espectrogramas del sonido original y del filtrado se muestran en la Figura 6-49.

Una medida alternativa y/o complementaria para aumentar la tolerancia al ruido es distinguir dos parámetros de potencia: la total y la relevante. La potencia relevante de un *frame* es aquella que se encuentra dentro de una banda de frecuencias que se puede considerar relevantes. El valor por defecto utilizado como banda relevante es la de 500Hz a 5kHz. Si se realiza un filtrado paso de banda se limita muy considerablemente este ruido de baja frecuencia, haciendo que la potencia relevante coincida sensiblemente con la potencia total.

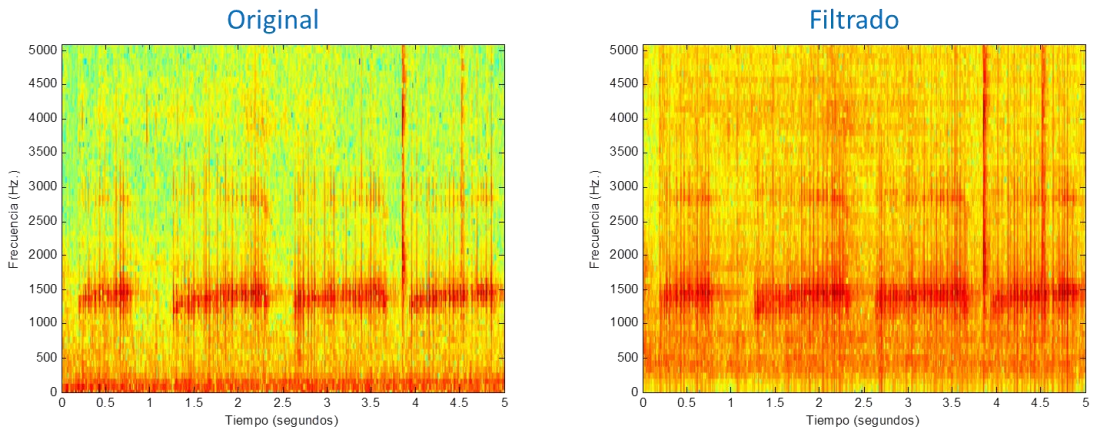


Figura 6-49 Espectrograma de uno de los archivos estudiados antes y después del filtrado

Adicionalmente, la implementación que se realice en el nodo debe ser capaz de realizar el proceso de clasificación en tiempo real. Esto se deriva de la condición de diseño donde se prefiere que el proceso se realice en forma local. Es decir, que los tiempos de proceso deben ser compatibles con la plataforma elegida. Estos tiempos de proceso se pueden dividir de acuerdo con lo reflejado en la Figura 6-50.

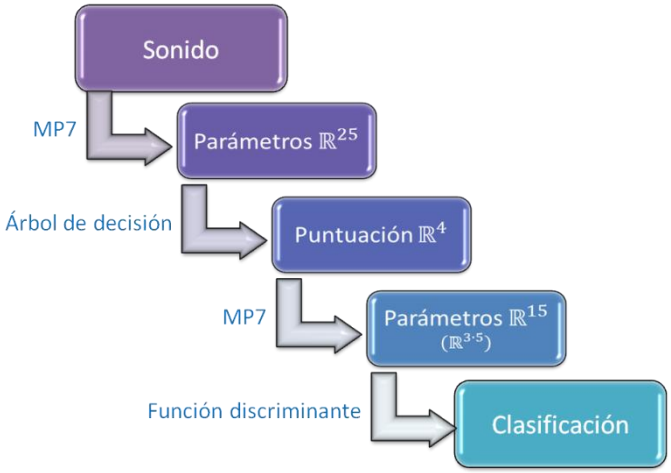


Figura 6-50 Proceso global de clasificación de sonidos

Los estudios de tiempo que siguen a continuación se han realizado utilizando un procesador Intel Core i7-4770 a 3.4 GHz con 8 GB de RAM bajo Windows 8 y ejecutando un programa MATLAB.

Si bien cabría esperar peores resultados con procesadores menos potentes, también pueden mejorarse:

- Por el uso de procesadores específicos para tratamiento de señales (DSP).
- Por reducción de la frecuencia de muestreo, ya que las pruebas se han realizado con muestreos a 44.1 kHz, sin embargo se ha filtrado por encima de 10 kHz.
- Por el uso de lenguajes de programación de menor nivel que MATLAB.

- Por optimización de la eficiencia de los algoritmos utilizados.

En definitiva, se puede considerar que la validez de los resultados en esta plataforma de diseño es un buen indicador de la viabilidad del algoritmo.

El primero de los tiempos de proceso que se debe considerar es el de cálculo de los 18 parámetros MPEG-7 de cada *frame*, que tiene un valor de aproximadamente $3.2\ ms$, siendo la duración del *frame*, la normalizada por MPEG-7, de 10 mseg. Por tanto, el tiempo de cálculo de los parámetros ocupa aproximadamente un tercio del tiempo del *frame*, un 32%. La reducción de dimensionalidad hasta dejar sólo 5 parámetros relevantes no disminuye sensiblemente este tiempo. Y por otra parte, el mecanismo de ventana deslizante para 5 *frames*, si bien multiplica por 5 el tiempo de cálculo de los 25 parámetros MPEG-7, lo hace en el tiempo de 5 *frames*, con lo que le tiempo por *frame* es el mismo.

El segundo de los tiempos de proceso es el de clasificación de *frames* mediante un algoritmo de árbol de decisión. Para este caso se ha obtenido un valor de 0.67 mseg., un tiempo claramente dentro de los $10\ ms$ de duración del *frame*, un 6.7%.

El siguiente paso, la obtención de los parámetros MPEG-7 de las series derivadas, requiere un tiempo de proceso de aproximadamente $0.03\ ms$ por cada *frame*, un 0.3% del tiempo total del *frame*. Nótese que este tiempo incluye la generación de los parámetros de cada una de las 4 series derivadas, una por cada sonido, más la correspondiente al ruido. Pero, a diferencia de la obtención de parámetros MPEG-7 del primer paso, cada serie derivada contiene un único valor en cada *frame*, el valor de la puntuación. Mientras que el sonido original tiene 441 valores en cada *frame*, a una frecuencia de muestreo de 44.1 kHz. Como puede verse se cumple que

$$\frac{3.2\,\frac{ms}{frame}}{441\,\frac{valores}{frame}} = 7.3\,\frac{\mu s}{valor} \approx \frac{30\,\frac{\mu s}{frame}}{4\,\frac{valores}{frame}} = 7.5\,\frac{\mu s}{valor}\ , \qquad \textbf{(6.5)}$$

es decir, que los tiempos de cálculo de los parámetros MPEG-7 son consistentes en los 2 pasos en los que se requieren.

El último paso del algoritmo, la clasificación de las series derivadas mediante la función discriminante, requiere un tiempo de proceso aproximado de $0.50\ ms$. De nuevo un valor claramente dentro de los $10\ ms$ de duración del *frame*, un 5%.

La Tabla 6-19 resume los tiempos empleados en cada paso del proceso de cada *frame*. Como puede comprobarse el tiempo total es inferior a los $10\ ms$ de duración del *frame* por lo que se comprueba que el proceso completo puede realizarse en tiempo real.

Proceso	Tiempo (mseg.)	Tiempo relativo (%)
Obtención de parámetros MP7 de un *frame*	3.20	32%
Clasificación de un *frame*	0.67	6.7%
Obtención de parámetros MP7 de las serie derivada	0.03	0.3%
Clasificación de la serie derivada	0.50	5%
Total tiempo proceso	**4.40**	**44%**

Tabla 6-19 Tiempos de proceso

En la Figura 6-51 se resumen gráficamente el reparto del tiempo entre los procesos.

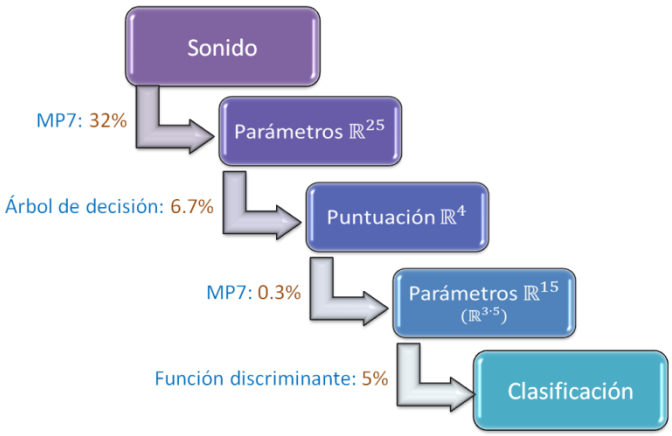


Figura 6-51 Porcentaje de tiempo del *frame* por cada proceso

CAPÍTULO 7. RESUMEN Y CONCLUSIONES

7.1. Resumen

En el presente trabajo de tesis se ha estudiado la aplicabilidad de técnicas de minerías de datos en la clasificación de secuencias temporales, usando como parámetros los obtenidos a partir de los descriptores de bajo nivel del sonido propuestos por el estándar MPEG-7. En concreto los temas abordados han sido los siguientes:

- Resumen del estado del arte.
    - Se justifica la conveniencia de utilizar el tamaño y distribución de las poblaciones de anuros como un indicador del cambio climático. La presencia y la categorización de los anuros existentes en un territorio se determina mediante la clasificación de sus cantos.
    - Se describen las principales técnicas de procesado de sonidos que permiten su caracterización.
    - Se presentan los principales clasificadores utilizados en minería de datos, tanto para datos secuenciales como para datos no secuenciales.
- Aportaciones más significativas.
    - Se seleccionan y definen un conjunto de 18 parámetros basados en la norma MPEG-7, resultando ser muy adecuados para la posterior clasificación de sonidos.
    - Se seleccionan 9 técnicas de clasificación sobre datos no secuenciales y se comparan sus resultados. Es técnicas son las siguientes: distancia mínima; máxima verosimilitud; árboles de decisión; k-vecinos más próximos; máquinas de vectores soporte; regresión logística; redes neuronales; función discriminante; y clasificador bayesiano.
    - Se presentan 4 métodos que permiten tener en cuenta el carácter secuencial de los sonidos utilizando como base los 9 clasificadores anteriores. Se comparan los resultados de estos métodos entre sí, así

como con los obtenidos por los 9 clasificadores no secuenciales. Los 4 métodos son: parámetros temporales; ventana deslizante; ventana deslizante recursiva; y clasificación de parámetros ARIMA.

- Se configuran unos Modelos Ocultos de Markov como clasificador secuencial puro, comparando sus resultados con los obtenidos por otras técnicas.
- Se presenta el concepto de serie vectorial derivada, se proponen 3 métodos para su clasificación y se comparan sus resultados. Estos 3 métodos son: clasificación por conteo; clasificación por semejanza; clasificación paramétrica.
- Se propone la clasificación de sonidos mediante la clasificación de sus series derivadas.

- Aplicación a un caso real.
  - Se aplican estas propuestas a la clasificación de un conjunto de 63 grabaciones reales realizadas en campo, con más de hora y media de duración acumulada. Conviene destacar que las condiciones de grabación de los sonidos provoca que éstos sean de baja o muy baja calidad.
  - Se construye un prototipo en laboratorio, con más de 10.000 líneas de código, que explora y compara el conjunto de soluciones propuestas.
  - Se ajusta la solución propuesta para tener en cuenta aspectos de industrialización tales como: la tolerancia ante ruidos y perturbaciones; la normalización de la representación de los sonidos; la capacidad de proceso en tiempo real; y la integrabilidad en sistemas de bajo consumo y bajo coste.

Un resumen de los resultados obtenidos con las distintas técnicas analizadas puede verse en la (Figura 6-3).

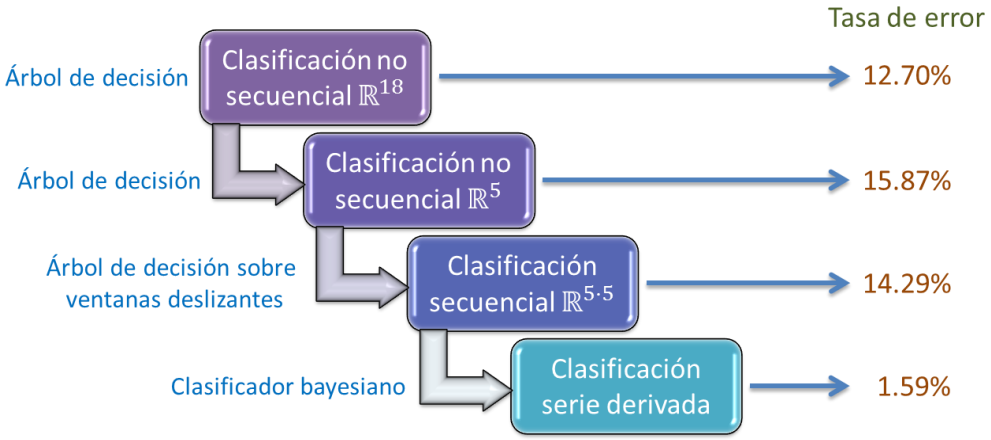


Figura 7-7-1 Reducción de dimensionalidad y aplicación de ventana deslizante

7.2. Conclusiones

Del desarrollo del trabajo realizado se pueden destacar las siguientes conclusiones:

- El uso de parámetros MPEG-7 supone una alternativa que
  - Ofrece excelentes resultados de clasificación.
  - Utiliza definiciones y algoritmos normalizados.
- El árbol de decisión es el clasificador no secuencial que proporciona mejores resultados.
- El mejor manejo del carácter secuencial de los sonidos lo realiza el método de ventana deslizante apoyado en un clasificador de árbol de decisión. Los resultados obtenidos superan claramente a los de los Modelos Ocultos de Markov, el clasificador secuencial puro recomendado por la norma MPEG-7.
  - Una ventana de tamaño 5 ha ofrecido buenos resultados.
- El tratamiento posterior de las series derivadas mejora sensiblemente las prestaciones de la clasificación. El mejor resultado se ha obtenido con una clasificación paramétrica que utiliza un clasificador bayesiano sobre la serie derivada obtenida mediante un árbol de decisión.
- La tasa de error final de clasificación es inferior al 2%, una cifra realmente baja considerando la baja calidad de los sonidos tratados.
- La solución propuesta tiene todas las características solicitadas para ser considerada industrial ya que:
  - La representación de los sonidos se realiza en base a parámetros normalizados.
  - Funciona muy bien ante ruidos y perturbaciones de la señal sonora.
  - Los algoritmos elegidos pueden utilizarse en tiempo real.
  - Es integrable en sistemas de bajo consumo y bajo coste.

7.3. Líneas de continuación

Derivadas de este trabajo y como continuación de la investigación realizada, se plantean distintas líneas e ideas que permitirían continuar el desarrollo iniciado:

- Extensión de los resultados de esta tesis a un conjunto más amplio de sonidos perteneciente a un mayor número de clases.
- Consideración de otras técnicas de extracción de características de sonidos y, en particular, comparación con las basadas en parámetros MFCC.
- Consideración de otras técnicas que reflejen el carácter secuencial de los sonidos basadas en la construcción de características y, en particular, comparación con las basadas en parámetros Δ-MFCC.
- Formalización de las técnicas de reducción de dimensionalidad abordadas de forma heurística en este trabajo.

- Desarrollo de un prototipo completo de una Red de Sensores Inalámbricos (*WSN*), que tenga la capacidad de capturar, procesar y clasificar localmente los sonidos, enviando el resultado a un centro de control vía radio.
- Extensión de la aplicación del método de clasificación propuesto sobre otros tipos de datos con estructura de secuencial, entre las que convendría destacar, señales procedentes de electroencefalograma (EGG) y electrocardiograma (ECG).

CAPÍTULO 8. REFERENCIAS

Aggarwal, C. C. (2007). Data streams: models and algorithms. *Springer Science & Business Media*, *31*.

Ahmed, N. K., Atiya, A. F., Gayar, N. El, & El-Shishiny, H. (2010). An Empirical Comparison of Machine Learning Models for Time Series Forecasting. *Econometric Reviews*, *29*(5–6), 594–621. http://doi.org/10.1080/07474938.2010.481556

Aide, T. M., Corrada-Bravo, C., Campos-Cerqueira, M., Milan, C., Vega, G., & Alvarez, R. (2013). Real-time bioacoustics monitoring and automated species identification. *PeerJ*, *1*, e103.

Akaike, H. (1974). A new look at the statistical model identification. *Automatic Control, IEEE Transactions*, *19(6)*, 716–723.

Animalsoundarchive. (2015). http://www.animalsoundarchive.org/.

Araujo, B. (2006). Aprendizaje automático: conceptos básicos y avanzados. *Aspectos Prácticos Utilizando El Software Weka. …*. Retrieved from http://dspace.ucbscz.edu.bo/dspace/handle/123456789/10111

Avaro, O., & Salembier, P. (2001). MPEG-7 Systems: overview. *Circuits and Systems for Video Technology, IEEE Transactions on*, *11*(6), 760–764. http://doi.org/10.1109/76.927437

Bardeli, R. (2009). Similarity search in animal sound databases. *IEEE Transactions on Multimedia*, *11*(1), 68–76.

Baum, L. E., & Eagon, J. A. (1967). An inequality with applications to statistical estimation for probabilistic functions of Markov processes and to a model for ecology. *Bull. Amer. Math. Soc*, *73(3)*(360–363).

Bellis, E. D. (1957). The effects of temperature on salientian breeding calls. *Copeia*, 85–89.

Benesty, J. (2008). Springer handbook of speech processing. *Springer Science &*

Business Media.

Bernal, J., & Gómez, Pedro, Bobadilla, J. (1999). Una visión práctica en el uso de la Transformada de Fourier como herramienta para el análisis espectral de la voz. *Estudios de Fonética Experimental*, *10*, 75–105.

Bernal Bermúdez, J., Bobadilla Sancho, J., & Gómez Vilda, P. (2000). *Reconocimiento de voz y fonética acústica*. Madrid : RA-MA. Retrieved from http://fama.us.es/record=b1526034~S5*spi

Bishop, C. M. (2006). Pattern recognition and machine learning. *Springer*.

Boser, B. E., Guyon, I. M., & Vapnik, V. N. (1992). A Training Algorithm for Optimal Margin Classifiers. *Proceedings of the Fifth Annual ACM Workshop on Computational Learning Theory*, 144–152. Retrieved from http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.21.3818

Box, G. E., Jenkins, G. M., & Reinsel, G. C. (2011). Time series analysis: forecasting and control. *John Wiley & Sons*, *734*.

Bradbury, J. W., & Vehrencamp, S. L. (1998). Principles of animal communication. *Sinauer Associates*.

Brand, M. (1997). Coupled hidden Markov models for modeling interacting processes. *Tech. Rep. 405, MIT Media Lab*.

British Library. (2015). http://www.bl.uk/soundarchive.

Brookes, M. (1998a). Description of disteusq. Retrieved from http://www.ee.ic.ac.uk/hp/staff/dmb/voicebox/doc/voicebox/disteusq.html

Brookes, M. (1998b). Description of kmeanlbg. Retrieved from http://www.ee.ic.ac.uk/hp/staff/dmb/voicebox/doc/voicebox/kmeanlbg.html

Cambron, M. E., & Bowker, R. G. (2006). An automated digital sound recording system: the Amphibulator. *ISM'06. Eighth IEEE International Symposium on Multimedia*, 592–600.

Cambron, M. E., & Emberton, A. C. (2013). Amphibulator II. *IEEE*.

Casey, M. (2001). MPEG-7 sound-recognition tools. *IEEE Transactions on Circuits and Systems for Video Technology*, *11*(6), 737–747.

Chang, S. F., Sikora, T., & Puri, A. (2001). Overview of the MPEG-7 standard. *IEEE Transactions on Circuits and Systems for Video Technology*, *11*(6), 688–695. http://doi.org/10.1109/76.927421

Cohen, J. (1960). A Coefficient of Agreement for Nominal Scales. *Educational and Psychological Measurement*, *20*(1), 37–46. Retrieved from http://epm.sagepub.com/cgi/doi/10.1177/001316446002000104

Cortes, C., & Vapnik, V. (1995). Support-Vector Networks. *Machine Learning*, *20*(3),

273–297.

Cover, T., & Hart, P. (1967). Nearest neighbor pattern classification. *IEEE Transactions on Information Theory*, *13*(1), 21–27. http://doi.org/10.1109/TIT.1967.1053964

Cramer, J. S. (2005). Logit Models From Economics and Other Fields. *Technometrics*. http://doi.org/10.1198/tech.2005.s829

Cristianini, N., & Shawe-Taylor, J. (2000). An introduction to Support Vector Machines. *Cambridge University Press, Cambrige*.

Day, N., & Martinez, J. M. (2001). Introduction to MPEG-7 (v. 3.0). *International Organization for Standardization, ISO/IEC JTC1/SC29/WG11, Coding of Moving Pictures and Audio N*, *4032*, 1--10.

Deng, K., Moore, A. W., & Nechyba, M. C. (1997). Learning to recognize time series: Combining ARMA models with memory-based learning. *In Computational Intelligence in Robotics and Automation, 1997. CIRA'97, Proceedings, 1997 IEEE International Symposium on (Pp. 246-251). IEEE.*

Deutsch, C. A., Tewksbury, J. J., Huey, R. B., Sheldon, K. S., Ghalambor, C. K., Haak, D. C., & Martin, P. R. (2008). Impacts of climate warming on terrestrial ectotherms across latitude. *Proceedings of the National Academy of Sciences*, *105*(18), 6668–6672.

Diaz, J. J., Nakamura, E. F., Yehia, H. C., Salles, J., & Loureiro, A. (2012). On the Use of Compressive Sensing for the Reconstruction of Anuran Sounds in a Wireless Sensor Network. *In Green Computing and Communications (GreenCom), 2012 IEEE International Conference*, 394–399.

Dietterich, T. G. (2002). Machine learning for sequential data: A review. *In Structural, Syntactic, and Statistical Pattern Recognition (Pp. 15-30). Springer Berlin Heidelberg*.

Dobson, A. J., & Barnett, A. (2008). *An introduction to generalized linear models*. CRC press.

Du, K. L., & Swamy, M. N. S. (2013). *Neural Networks and Statistical Analyses*. Springer Science & Business Media.

Duarte, H., Tejedo, M., Katzenberger, M., Marangoni, F., Baldo, D., Beltrán, J. F., & Gonzalez-Voyer, A. (2012). Can amphibians take the heat? Vulnerability to climate warming in subtropical and temperate larval amphibian communities. *Global Change Biology*, *18*(2), 412–421.

Esling, P., & Agon, C. (2012). Time-series data mining. *ACM Computing Surveys (CSUR)*, *45*(1), 12.

ETSI, E. (2002). 202 050 v1. 1.3: Speech Processing, Transmission and Quality Aspects (STQ); Distributed Speech Recognition; Advanced Front-end Feature Extraction Algorithm; Compression Algorithms. *ETSI Standard*.

Faúndez Zanuy, M. (2000). *Tratamiento digital de voz e imagen y aplicación a la multimedia*. Barcelona : Marcombo. Retrieved from http://fama.us.es/record=b1489137~S5*spi

Fawcett, T. (2006). An introduction to ROC analysis. *Pattern Recognition Letters*, *27*(8), 861–874.

Fay, R. R., & Popper, A. N. (2012). Comparative hearing: fish and amphibians. *Springer Science & Business Media*.

Flach, P. (2012). Machine learning: the art and science of algorithms that make sense of data. *Cambridge University Press*.

Fonozoo. (2015). Http://www.fonozoo.com.

Fu, T. (2011). A review on time series data mining. *Engineering Applications of Artificial Intelligence*, *24*(1), 164–181. http://doi.org/10.1016/j.engappai.2010.09.007

Fulop, S. A. (2011). *Speech Spectrum Analysis*. Berlin, Heidelberg: Springer Berlin Heidelberg. http://doi.org/10.1007/978-3-642-17478-0

Galindo Riaño, P. L. (1996). *Introducción al Reconocimiento de la Voz*. Cádiz: Universidad de Cádiz.

Gayou, D. C. (1984). Effects of temperature on the mating call of Hyla versicolor. *Copeia*, 733–738.

Gerhardt, H. C., & Huber, F. (2002). Acoustic communication in insects and anurans: common problems and diverse solutions. *University of Chicago Press*.

Gerhardt, H. C., & Mudry, K. M. (1980). Temperature effects on frequency preferences and mating call frequencies in the green treefrog, Hyla cinerea (Anura: Hylidae). *Journal of Comparative Physiology*, *137*(1), 1–6.

Geurts, P. (2001). Pattern extraction for time series classification. *In Principles of Data Mining and Knowledge Discovery (Pp. 115-127). Springer Berlin Heidelberg*.

Golub, G. H., & Van Loan, C. F. (1996). Matrix Computations. *Physics Today*. http://doi.org/10.1063/1.3060478

Gopi, E. S. (2014). *Digital Speech Processing Using Matlab*. New Delhi: Springer India. http://doi.org/10.1007/978-81-322-1677-3

Gorunescu, F. (2011). Data mining: Concepts, models and techniques. *Intelligent Systems Reference Library*, *12*.

Han, J., Kamber, M., & Pei, J. (2011). Data Mining : Concepts and Techniques Third Edition. *ELSEVIER*, 770.

Härdle, W., & Simar, L. (2015). Applied Multivariate Statistical Analysis. *Technometrics*, (4), 581. http://doi.org/10.1198/tech.2005.s319

Hastie, T., Tibshirani, R., & Friedman, J. (2005). *The Elements of Statistical Learning:*

Data Mining, Inference, and Prediction, Second Edition. *Springer series in statistics*. http://doi.org/10.1007/978-0-387-84858-7

Hevia, C. (2008). Maximum likelihood estimation of an ARMA (p, q) model. *The World Bank, DECRG*.

Hornik, K., Stinchcombe, M., & White, H. (1989). Multilayer feedforward networks are universal approximators. *Neural Networks*, *2*(5), 359–366. http://doi.org/10.1016/0893-6080(89)90020-8

Huang, C. J., Yang, Y. J., Yang, D. X., & Chen, Y. J. (2009). Frog classification using machine learning techniques. *Expert Systems with Applications, 36(2), 3737-3743*.

Huey, R. B., Deutsch, C. A., Tewksbury, J. J., Vitt, L. J., Hertz, P. E., Pérez, H. J. Á., & Garland, T. (2009). Why tropical forest lizards are vulnerable to climate warming. *Proceedings of the Royal Society of London B: Biological Sciences*, *rspb-2008*.

Hyvärinen, A., Hoyer, P. O., & Inki, M. (2001). Topographic independent component analysis. *Neural Computation*, *13*(7), 1527–1558. http://doi.org/10.1162/089976601750264992

ISO. (2001). *Information Technology - Multimedia Content Description Interface - Part 4: Audio*.

ISO/MPEG. (1998). MPEG Requirements Group. MPEG-7: context and objectives. *N24600. MPEG Atlantic City Meeting.*

Kadous, M. W., & Sammut, C. (2005). Classification of Multivariate Time Series and Structured Data Using Constructive Induction. *Machine Learning*, *58*(2–3), 179–216. http://doi.org/10.1007/s10994-005-5826-5

Kearney, M., Shine, R., & Porter, W. P. (2009). The potential for behavioral thermoregulation to buffer “cold-blooded” animals against climate warming. *Proceedings of the National Academy of Sciences*, *106*(10), 3835–3840.

Kim, H. G., Moreau, N., & Sikora, T. (2004). Audio classification based on MPEG-7 spectral basis representations. *IEEE Transactions on Circuits and Systems for Video Technology*, *14*(5), 716–725.

Kim, H. G., Moreau, N., & Sikora, T. (2005). *MPEG-7 audio and beyond*. *Communication*.

Kim, H. G., & Sikora, T. (2004). How efficient is MPEG-7 for general sound recognition? *Audio Engineering Society Conference: 25th International Conference: Metadata for Audio. Audio Engineering Society*.

Koenen, R., & Pereira, F. (2000). MPEG-7: A standardized description of audiovisual content. *Signal Processing: Image Communication*, *16*(1–2), 5–13. http://doi.org/10.1016/S0923-5965(00)00014-X

Koskela, T. (2003). *Neural network methods in analysing and modelling time varying processes*. *Helsinki University of Technology*. Retrieved from

http://lib.tkk.fi/Diss/2003/isbn9512268183/

Le Cam, L. (1979). Maximum Likelihood: An Introduction. *Statistics Branch, Department of Mathematics, University of Maryland*.

Levinson, L. (1947). The Wiener {RMS} Criterion in Filter Design and Prediction, *25*(4), 261–278.

Li, R.-H., & Belford, G. G. (2002). Instability of decision tree classification algorithms. *Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining - KDD '02*, 570. http://doi.org/10.1145/775047.775131

Li, Y., Dong, M., & Ma, Y. (2008). Feature selection for clustering with constraints using Jensen-Shannon divergence. *In Pattern Recognition, 2008. ICPR 2008. 19th International Conference on (Pp. 1-4). IEEE*.

Linde, Y., Buzo, A., & Gray, R. M. (1980). An algorithm for vector quantizer design. *Communications, IEEE Transactions On, 28(1), 84-95*.

Llusia, D., Márquez, R., Beltrán, J. F., Benítez, M., & do Amaral, J. P. (2013). Calling behaviour under climate change: Geographical and seasonal variation of calling temperatures in ectotherms. *Global Change Biology*, *19*(9), 2655–2674.

Luque, A., Carrasco, A., Barbancho, J., & Romero, J. (2016). Sistema de identificación de sonidos mediante clasificación paramétrica de series derivadas. Patente n^{o} P201600960: Oficina Española de Patentes y Marcas.

Macaulaylibrary. (2015). http://macaulaylibrary.org/.

Márquez, R., & Bosch, J. (1995). Advertisement calls of the midwife toads Alytes (Amphibia, Anura, Discoglossidae) in continental Spain. *Journal of Zoological Systematics and Evolutionary Research*, *33*(3–4), 185–192.

Martínez. (2004). MPEG-7 Overview (version 10). *ISO/IEC JTC1/SC29/WG11 N*, *3158*.

Martínez, J. M. (2002). MPEG-7 overview of MPEG-7 description tools, Part 2. *IEEE Multimedia*, *9*, 83–93. http://doi.org/10.1109/MMUL.2002.1022862

McCulloch, W. S., & Pitts, W. (1943). A Logical Calculus of the Idea Immanent in Nervous Activity. *Bulletin of Mathematical Biophysics*, *5*, 115–133. http://doi.org/10.1007/BF02478259

Meadows, D. H., Meadows, D. L., Randers, J., & Behrens, W. W. (1972). The limits to growth. *New York*, *102*.

MPEG. (2005). Class hierarchy of MPEG-7 Audio Low Level Descriptors. Retrieved from http://mpeg.chiariglione.org/standards/mpeg-7/audio

Nieto, O. (2006). *Diseño de un reconocedor de comandos de voz para el DSP TMS320C6711*.

Noda, J. J., Travieso, C. M., & Sánchez-Rodríguez, D. (2016). Methodology for automatic bioacoustic classification of anurans based on feature fusion. *Expert Systems with Applications2*, *50*, 100–106.

Ntalampiras, S., Potamitis, I., & Fakotakis, N. (2008). Automatic recognition of urban soundscenes. *Springer Berlin Heidelberg*, *New Direct*, 147–153.

Núñez, P. V., & Fernández, D. G. (2002). Sistemas de descripción de contenidos multimedia. *Comunicaciones de Telefónica I+D*, *24*(TELEFÓNICA INVESTIGACIÓN Y DESARROLLO).

Pereira, F. (1996). MPEG-7 : A STANDARD FOR DESCRIBING AUDIOVISUAL INFORMATION.

Pires, A., & Hoy, R. R. (1992). Temperature coupling in cricket acoustic communication. *Journal of Comparative Physiology A*, *171*(1), 79–92.

Pörtner, H. O., & Knust, R. (2007). Climate change affects marine fishes through the oxygen limitation of thermal tolerance. *Science*, *315*(5808), 95–97.

Potamitis, I. (2015). Unsupervised dictionary extraction of bird vocalisations and new tools on assessing and visualising bird activity. *Ecological Informatics*, *26*, 6–17.

Povinelli, R. J. (1999). Time series data mining: identifying temporal patterns for characterization and prediction of time series events. *(Doctoral Dissertation, Faculty of the Graduate School, Marquette University)*, 193.

Quackenbush, S., & Lindsay, A. (2001). Overview of MPEG-7 Audio. *Transactions on Circuits and Systems for Video Technology*, *11*(6), 725–729. http://doi.org/10.1109/76.927430

Quariteri, T. F. (2002). *Discrete Time Speech Signal Processing: Principles and Practice*. (U. S. R. NJ, Ed.). Prentice Hall.

Quinlan, J. R. (1986). Induction of Decision Trees. *Machine Learning*, *1*(1), 81–106.

Rabiner, L., & Juang, B. H. (1993). Fundamentals of speech recognition. *Prentice Hall*.

Rabiner, L. L., & Juang, B.-H. B. (1993). *Fundamentals of Speech Recognition*. *Prentice Hall* (Vol. 103). Retrieved from http://cmp.felk.cvut.cz/cmp/support/phd112.html

Rabiner, L. R. (1989). A tutorial on hidden Markov models and selected applications in speech recognition. *Proceedings of the IEEE*, *77*(2), 257–286.

Rabiner, L. R., & Juang, B. H. (1986). An introduction to hidden Markov models. *ASSP Magazine, IEEE*, *3*(1), 4–16.

Rabiner, L. R., & Schafer, R. W. (1978). *Digital Processing of Speech Signals*. PRENTICE_HALL.

Real Academia Española. (2014). DICCIONARIO DE LA LENGUA ESPAÑOLA. Retrieved from http://lema.rae.es/drae/?val=practica

Riesz, F., & Nagy, S. (1990). *Functional analysis*. New York: Dover Publications, Inc.

Rokach, L., & Maimon, O. (2008). *Data Mining With Decision Trees: Theory and Applications*. *Series in Machine Perception and Artificial Intelligence, World Scientific* (Vol. 1). http://doi.org/10.1142/9789812771728

Salembier, P., Llach, J., & Garrido, L. (2002). Visual segment tree creation for MPEG-7 Description Schemes. *Pattern Recognition*, *35*(3), 563–579. http://doi.org/10.1016/S0031-3203(01)00060-7

Salembier, P., & Smith, J. R. (2001). MPEG-7 multimedia description schemes. *IEEE Transactions on Circuits and Systems for Video Technology*, *11*(6), 748–759. http://doi.org/10.1109/76.927435

Schneider, H. (1974). Structure of the mating calls and relationships of the European tree frogs (Hylidae, Anura). *Oecologia*, *14*(1–2), 99–110.

Shawe-Taylor, J., & Cristianini, N. (2004). *Kernel Methods for Pattern Analysis*. *Elements* (Vol. 47). http://doi.org/10.2277

Snell, R. C., & Milinazzo, F. (1993). Formant location from LPC analysis data. *IEEE Transactions on Speech and Audio Processing*, *1*(2), 129–134. http://doi.org/10.1109/89.222882

Sotoca, J. M., & Pla, F. (2010). Supervised feature selection by clustering using conditional mutual information-based distances. *Pattern Recognition*, *43*(6), 2068–2081.

Spackman, K. A. (1989). Signal detection theory: valuable tools for evaluating inductive learning. In *Proceedings of the Sixth International Workshop on Machine Learning* (pp. 160–163). http://doi.org/10.1016/b978-1-55860-036-2.50047-3

Stamp, M. (2012). A revealing introduction to hidden Markov models. *Department of Computer Science San Jose State University*.

Theodoridis, S., & Chellappa, R. (2013). Academic Press Library in Signal Processing: Signal Processing Theory and Machine Learning (Vol. 1). *Academic Press*.

Tiberiu, C. (2013). Sliding hidden markov model for evaluating discrete data. In *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* (Vol. 8168 LNCS, pp. 251–262).

Vicente, P. J. V. (2005). El estándar MPEG-7. *Revista de Ingeniería Informática Del CIIRM, Murcia (Espanha)*, (3), 1–5. article.

Wacker, A. G., & Landgrebe, D. A. (1971). The Minimum Distance Approach to Classification. *Purdue University. Information Note 100771*.

Walker, T. J. (1957). Specificity in the response of female tree crickets (Orthoptera, Gryllidae, Oecanthinae) to calling songs of the males. *Annals of the Entomological Society of America*, *50*(6), 626–636.

Walker, T. J. (1962). Factors responsible for intraspecific variation in the calling songs of crickets. *Evolution*, 407–428.

Weninger, F., & Schuller, B. (2011). Audio recognition in the wild: Static and dynamic classification on a real-world database of animal vocalizations. *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 337–340.

Whitney, A. W. (1971). A direct method of nonparametric measurement selection. *IEEE Transactions on Computers*, (9), 1100–1103.

Wikipedia. (2016). Principal component analysis. Retrieved June 20, 2002, from https://en.wikipedia.org/w/index.php?title=Principal_component_analysis&oldid=705236413

Xi, X., Keogh, E., Shelton, C., Wei, L., & Ratanamahatana, C. A. (2006). Fast time series classification using numerosity reduction. *In Proceedings of the 23rd International Conference on Machine Learning (Pp. 1033-1040). ACM*.

Xiang, S., Nie, F., & Zhang, C. (2008). Learning a Mahalanobis distance metric for data clustering and classification. *Pattern Recognition*, *41*(12), 3600–3612. http://doi.org/10.1016/j.patcog.2008.05.018

Zheng, F., Zhang, G., & Song, Z. (2001). Comparison of different implementations of MFCC. *Journal of Computer Science and Technology*, *16*(6), 582–589.

Zhong, S., & Ghosh, J. (2002). HMMs and coupled HMMs for multi-channel EEG classification. *In Proceedings of the IEEE International Joint Conference on Neural Networks*, *2*, 1254–1159.