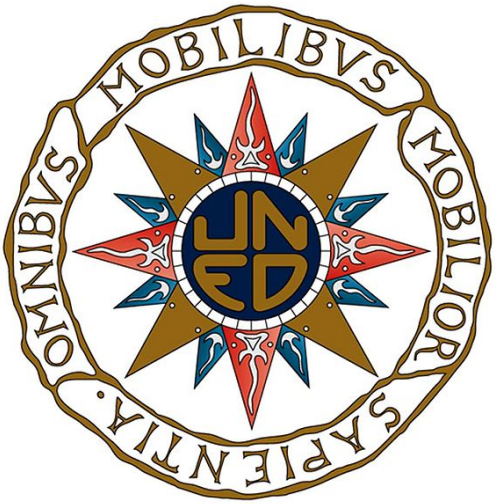




Máster en Ingeniería de Sistemas y Control

Trabajo de Fin de Máster

# Estudio de algoritmos para la identificación automática de componentes en mezclas

Alumno: José Ángel Martínez Fragoso

Directores: Raquel Dormido Canto

Natividad Duro Carralero

Ángel Mur Güeri

Curso 2019-2020

Convocatoria: septiembre 2020

Máster en Ingeniería de Sistemas y Control UNED/UCM

# Estudio de algoritmos para la identificación automática de componentes en mezclas

Trabajo de Fin de Máster de modalidad específica

Alumno: José Ángel Martínez Fragoso

Directores: Raquel Dormido Canto

Natividad Duro Carralero

Ángel Mur Güeri

Curso 2019-2020

Convocatoria: septiembre 2020

## Autorización:

Autorizo a la Universidad Complutense de Madrid (UCM) y a la Universidad Nacional de Educación a Distancia (UNED) a difundir y utilizar con fines académicos, no comerciales y mencionando expresamente a su autor, tanto la memoria de este Trabajo Fin de Máster, como el código, la documentación y/o prototipo desarrollado.

Firmado: José Ángel Martínez Fragoso

Firma de alumno

# Estudio de algoritmos para la identificación automática de componentes en mezclas


## Resumen:

La separación ciega de fuentes es una técnica que nos permite realizar la estimación de un conjunto de señales fuente, de las cuales a priori no se conoce su naturaleza, a partir de una serie de medidas que son una función desconocida de dichas señales.

Este trabajo pretende estudiar, dentro de la gran variedad de algoritmos y técnicas descritos en la literatura, el método que mejor se adapte a un caso concreto de separación como es la identificación de los componentes químicos que componen una mezcla multicomponente y la concentración de estos en ella.

En este trabajo se analizan los principales conjuntos de técnicas desarrollados, como el análisis de componentes independientes (ICA), el análisis de componentes dispersos (SCA), el análisis de componentes morfológicos (GMCA) o la factorización de matrices no negativas (NMF), detallando sus principales ventajas e inconvenientes para determinar cuál de ellos se desenvuelve mejor en el escenario planteado.

Posteriormente se expone una selección de algoritmo, como el algoritmo nGMCA (análisis de componentes morfológicos no negativos) y alguna de sus variantes, a un conjunto de pruebas empleando datos de diferente naturaleza para determinar cuál es el más adecuado para realizar la identificación de componentes en mezclas, para finalmente evaluar su rendimiento en la identificación de los componentes que componen un conjunto de mezclas reales y su concentración en ellas.



## Palabras clave:

Separación ciega de fuentes, Análisis de componentes independientes, Análisis de componentes dispersos, Análisis de componentes morfológicos, Factorización de matrices no negativas, Análisis de componentes morfológicos no negativos, Mezclas multicomponente.

# Analysis of algorithms for automatic identification of components in mixtures


## Abstract:

The blind source separation is a technique that allows to estimate a set of source signals from a series of measurements that are an unknown function of that signals which nature is not known a priori.

This paper aims to analyze, within the great variety of algorithms and techniques described in the literature, the method that best suits a specific case of separation, such as the identification of the chemical components that compound a multicomponent mixture and its concentration.

In this paper, the main sets of techniques developed are analyzed, such as independent component analysis (ICA), sparse component analysis (SCA), morphological component analysis (GMCA) and non-negative matrix factorization (NMF), detailing its main advantages and disadvantages to determine which of them performs better in the proposed scenario.

Subsequently, a selection of algorithms, such as the nGMCA (non-negative morphological component analysis) algorithm and some of its variants, are exposed to a set of tests using data of a different nature to determine which is the most appropriate to perform the identification of components in mixtures, to finally evaluate their performance identifying the components that compound a set of real mixtures and its concentration.



## Keywords:

Blind source separation, Independent component analysis, Sparse component analysis, Morphological component analysis, Non-negative matrix factorization, Non-negative morphological component analysis, Multicomponent mixtures.

# Índice de Contenidos

# Índice de figuras

# Índice de tablas

# Capítulo 1

# Introducción

## 1.1 Objetivo

Las técnicas de espectroscopía FTIR (Fourier Transform Infra-Red) proporcionan información sobre la distribución de los componentes en una mezcla. Existen una amplia variedad de técnicas basadas en la separación ciega de fuentes que podrían permitirnos separar tanto los componentes originales de una mezcla como sus respectivas concentraciones en la misma. Este trabajo pretende estudiar algoritmos automáticos que den solución a este problema concreto.

El objetivo con el que se enfoca este trabajo es triple. Por un lado, seleccionar mediante criterios objetivos alguna de las diferentes técnicas descritas en la literatura para resolver el problema de la separación ciega de fuentes aplicado a la identificación de componentes en mezclas. Por otra parte, evaluar diferentes algoritmos en escenarios controlados y medir su idoneidad para realizar esta tarea concreta. Por último, aplicar estos algoritmos a un ejemplo real evaluando su desempeño y constatando los diferentes obstáculos con los que podemos encontrarnos al tratar con datos no simulados.

## 1.2 Estructura de la memoria

Esta memoria se estructura en seis capítulos. El primero de ellos es una introducción, los dos siguientes son capítulos de corte teórico en los que se expondrán los conceptos en los que se asienta el trabajo práctico definido en los capítulos cuarto y quinto. El último capítulo corresponde a las conclusiones del trabajo y a la consideración de trabajos futuros en este ámbito. A continuación, se expone el contenido de cada uno de los capítulos:

**Capítulo 1.** *Introducción.*

Este capítulo define los objetivos del trabajo e introduce unos conceptos básicos que nos permiten afrontar de una manera más informada el desarrollo de la memoria.

**Capítulo 2.** *Técnicas de separación ciega de fuentes.*

Este segundo capítulo detalla diferentes técnicas BSS. Se exponen en él las diferentes técnicas recogidas en la literatura para separar fuentes, así como sus principales características y restricciones. Esta sección finaliza con una experiencia ilustrativa que justifica la elección de la "factorización de matrices no-negativas" (NMF) y el "análisis

generalizado de componentes morfológico no negativos" (nGMCA) como las técnicas más adecuadas para su uso en el ámbito de la identificación de componentes en mezclas.

**Capítulo 3.** *NMF y dispersión.*

Esta sección recoge los principales algoritmos que se evaluarán en este trabajo. Se muestra la evolución de las técnicas que desarrollan NMF para la resolución del problema BSS y la posterior inclusión de restricciones de dispersión, creando una nueva familia de algoritmos híbridos.

**Capítulo 4.** *Evaluación de Algoritmos.*

En este capítulo se evalúan, desde un enfoque práctico, los algoritmos descritos en el capítulo anterior. Los distintos algoritmos se exponen a diferentes conjuntos de datos y casos de prueba para medir su comportamiento y poder seleccionar uno o varios candidatos para llevar a cabo la tarea propuesta en el capítulo 5.

**Capítulo 5.** Aplicación en mezclas reales.

El objetivo en este punto es desarrollar la separación de componentes sobre un conjunto real de mezclas. Se contrastará la eficacia en la separación del algoritmo seleccionado en el capítulo 4 y se comprobará la problemática de un entorno real de separación.

**Capítulo 6.** *Conclusiones y trabajos futuros*.

Finalmente, en el último capítulo se exponen las conclusiones extraídas de las pruebas de evaluación realizadas en este trabajo y se ofrecen diferentes líneas sobre las que desarrollar trabajos futuros en este ámbito.

## 1.3 Conceptos básicos

### 1.3.1 Espectroscopia

La espectroscopia estudia la interacción entre la radiación electromagnética y la materia. La radiación incidente sobre la materia es diferente a la saliente por efecto de dicha interacción y el resultado de esta proporciona información útil sobre la sustancia involucrada con relación a su estructura molecular.

La espectroscopia infrarroja, también conocida como FTIR, o simplemente IR, estudia los fenómenos de interacción entre la radiación de origen infrarrojo y la materia. Esencialmente la energía de la radiación, localizada en determinada longitud de onda del infrarrojo, es

absorbida por una molécula (o parte de ella) que se encuentra vibrando en su estado basal a la misma longitud de onda que la radiación infrarroja incidente, provocando con ello un cambio en la intensidad de la vibración.

La espectroscopía FTIR es uno de los métodos preferidos para la detección infrarroja de especies separadas cromatográficamente. La popularidad de FTIR se debe principalmente a su capacidad de exploración rápida y múltiple mediante interferómetro, que es un tipo de sensor relativamente económico y fácil de construir. La interferometría produce una forma de onda compleja que es una suma de contribuciones de todas las longitudes de onda emitidas por la fuente. La discriminación de longitud de onda se deriva de la propiedad de que las contribuciones de longitud de onda se modulan en diferentes frecuencias.

FTIR es una técnica de espectroscopia de absorción, por la que se hace pasar una luz infrarroja de longitud de onda media (400-4000 nanómetros) a través de una muestra, algunas de las longitudes de onda son absorbidas mientras que otras simplemente pasan a través de la muestra sin verse afectadas. Diferentes enlaces moleculares absorben una cantidad específica de energía y estas pérdidas de energía corresponde a los picos devueltos que pueden leerse en las gráficas de una espectroscopia, como por ejemplo en la figura 1.1.

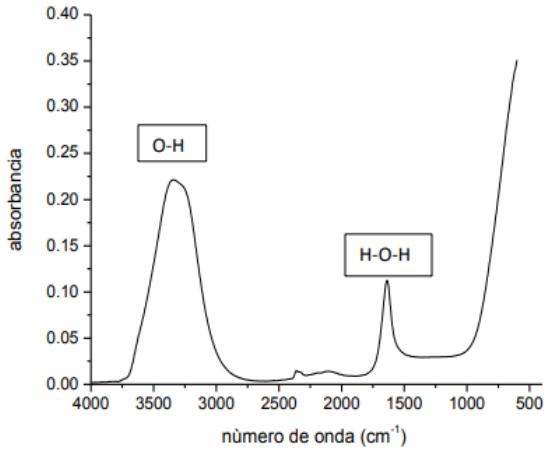



*Figura 1.1 Espectro infrarrojo del Agua. Presenta dos bandas de absorción intensas asignadas a los enlaces O-H y H-O-H [Mondragón 2017]*

Los métodos espectroscópicos de análisis tienen un amplio uso en una gran variedad de campos. En la industria, la medicina, en la conservación del patrimonio histórico y artístico, o en el desarrollo científico en general se utiliza el análisis del espectro óptico de compuestos para llevar a cabo infinidad de trabajos [White, 1990], [Sánchez, 2003], [Pérez-Alonso et alt., 2006], [Rakesh and Charmi, 2014].

En el caso particular de la espectroscopia infrarroja una de las mayores dificultades en el proceso de identificación de compuestos es determinar qué tipo de elementos componen una mezcla y en qué proporción se encuentran en ella. La mayoría de las técnicas espectroscópicas empleadas para la identificación y cuantificación de elementos se basan

en la comparación contra espectros de referencia, pero sería muy útil poder determinar los elementos y su concentración en una mezcla sin tener un conocimiento apriorístico.

En el presente trabajo se estudian algoritmos para la separación de componentes en mezclas basados en espectros FTIR, aunque también pueden aplicarse a otros espectros cromatográficos como es el caso del espectro Raman.

### 1.3.2 "Cocktail party problem" y Separación ciega de fuentes

La identificación de componentes en mezclas guarda relación con un problema descrito por el ingeniero y neurocientífico británico Colin Cherry en 1953 que se denominó "Cocktail Party Problem" (CPP). El CPP es un fenómeno psico-acústico que se refiere a la notable capacidad humana de atender selectivamente y reconocer una fuente de entrada auditiva en un entorno ruidoso, donde la interferencia auditiva se produce por un discurso en competencia de sonidos, o una variedad de ruidos, que a menudo se supone que son independientes entre sí [*Cherry, 1953*]. Tras los primeros trabajos pioneros se han dedicado numerosos esfuerzos al CPP en diversos campos: fisiología, neurobiología, psicofisiología, psicología cognitiva, biofísica, informática e ingeniería. Mas de medio siglo después del trabajo seminal de Cherry el enigma sobre la maravillosa capacidad de percepción auditiva de los seres humanos sigue siendo un misterio. Para desvelar el misterio e imitar la capacidad humana con una máquina, los neurocientíficos computacionales, los informáticos y los ingenieros han intentado modelar y simplificar esta compleja tarea perceptiva como un problema matemático para el cual se busca una solución computacional manejable.

Podríamos describir el CPP como muestra la figura 1.2. Imaginemos que colocamos dos micrófonos en una sala en la que están hablando dos personas simultáneamente. Estos dos micrófonos nos proporcionarán dos señales que llamaremos $x_1$ y $x_2$ compuestas por una mezcla de los sonidos que provienen de las señales fuente $S_1$ y $S_2$.

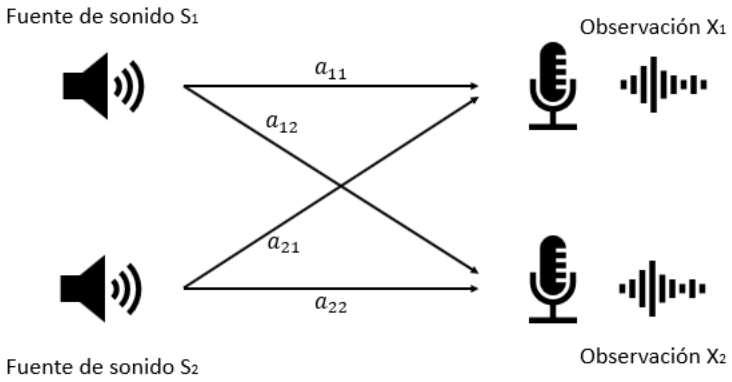



*Figura 1.2 Representación gráfica del CPP*

Las señales de observación o señales de mezcla $x_1(t)$ y $x_2(t)$ son las grabaciones en cada instante $(t)$ tomadas por los micrófonos y serán el resultado de una combinación lineal de las señales

emitidas representadas en el dominio del tiempo como muestra la ecuación (1.1) [Haykin and Chen, 2005].

$$\begin{aligned} x_1(t) &= a_{11}s_1(t) + a_{12}s_2(t) \\ x_2(t) &= a_{21}s_1(t) + a_{22}s_2(t) \end{aligned} \tag{1.1}$$

Siendo $a_{ij}$ los denominados parámetros de mezclado que representan la contribución de cada fuente a cada mezcla xi(t). Estos parámetros son constantes dependientes de la distancia de cada uno de los micrófonos a las fuentes emisoras de sonido.

El objetivo fundamental del CPP es separar cada una de las fuentes de sonido originales a partir del conocimiento de las mezclas. Este proceso sería trivial si los parámetros $a_{ij}$ fueran conocidos a priori, pero la complejidad del CPP deriva del hecho de que generalmente se carece de este conocimiento apriorístico.

La metodología de separación ciega de fuentes (BSS) proporciona un marco matemático aportando un conjunto de técnicas para la resolución del CPP. La metodología BSS nos permite recuperar señales independientes desconocidas a partir de una combinación lineal de las mismas.

Este marco teórico puede aplicarse a una gran cantidad de problemas de múltiples dimensiones, pero en este trabajo nos centraremos en cómo nos puede ayudar a encontrar los elementos originales de una mezcla multicomponente y su concentración en ella.

### 1.3.3 Formulación del problema

La separación ciega de fuentes aplicado a la identificación de componentes en mezclas se formula de la siguiente manera [Rapin et alt., 2013].

Supongamos un sistema donde partimos de una serie de $\boldsymbol{n}$ componentes desconocidos $S_1(f), S_2(f), \ldots, S_n(f)$. Estos componentes se mezclan en diferentes concentraciones dando lugar a un conjunto de diferentes observaciones de $\boldsymbol{m}$ mezclas $(\boldsymbol{m \geq n})$. Cada una de estas observaciones $Y_i(f)$ se obtendrá de una mezcla de los componentes fuente $S_i(f)$ más un ruido externo $N_i(f)$ que es incorporado al sistema. La relación entre las observaciones y los componentes originales vendrá dada por el sistema de ecuaciones (1.2).

$$\begin{aligned} Y_1(f) &= a_{11}S_1(f) + a_{12}S_2(f) + \cdots + a_{1n}S_n(f) + \cdots + N_1(f) \\ Y_2(f) &= a_{21}S_1(f) + a_{22}S_2(f) + \ldots + a_{2n}S_n(f) + \cdots + N_2(f) \\ &\vdots \\ Y_m(f) &= a_{m1}S_1(f) + a_{m2}S_2(f) + \ldots + a_{mn}S_n(f) + \cdots + N_m(f) \end{aligned} \tag{1.2}$$

O expresándolo de forma análoga de manera matricial mediante la ecuación (1.3) [Rapin et alt., 2014].

$$\mathbf{Y} = \mathbf{AS} + \mathbf{N} \tag{1.3}$$

Siendo $\mathbf{Y} = [\boldsymbol{Y_1}, \dots, \boldsymbol{Y_m}]^{\boldsymbol{t}}$ el vector de muestra de las mezclas observadas, $\mathbf{A}$ la matriz de mezcla de dimensiones $\boldsymbol{m\ x\ n}$ que contiene las proporciones de cada componente en la mezcla, $\mathbf{S} = [\boldsymbol{S_1}, \dots, \boldsymbol{S_n}]^{\boldsymbol{t}}$ el vector de muestra de los componentes fuente originales y siendo $\mathbf{N} = [\boldsymbol{N_1}, \dots, \boldsymbol{N_n}]^{\boldsymbol{t}}$ el vector de ruido cuyas componentes son estadísticamente independientes de las fuentes.

Por lo tanto, el problema BSS consiste en encontrar una matriz $\mathbf{B}$ de dimensiones $\boldsymbol{n\ x\ m}$ de forma que al pasarle las observaciones $\boldsymbol{Y_m}$ nos permita extraer unas salidas de componentes $\mathbf{Z} = [\boldsymbol{Z_1}, \dots, \boldsymbol{Z_n}]^{\boldsymbol{t}}$ lo más parecidas posible a los componentes originales de la mezcla.

$$\mathbf{Z} = \mathbf{BY} = \mathbf{BAS} + \mathbf{BN} \tag{1.4}$$

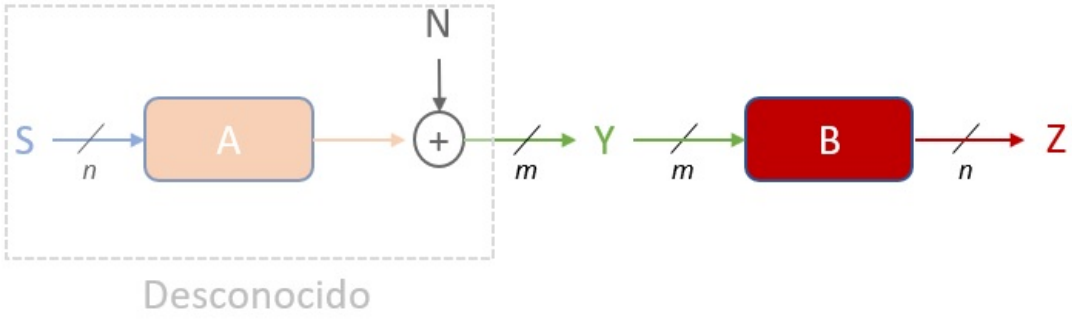



*Figura 1.3 Diagrama de bloques del problema BSS*

El problema de la separación ciega consistirá entonces en la estimación de los coeficientes $\boldsymbol{A_{i,j}}$ de la matriz de mezcla $\mathbf{A}$ y de los componentes $\boldsymbol{S_i}$ a partir de las observaciones $\boldsymbol{Y_i}$ formadas como una combinación lineal de los componentes, como muestra la ecuación (1.5).

$$\boldsymbol{Y_i} = \boldsymbol{A_{i,1}S_i} + \boldsymbol{A_{i,2}S_2} + \cdots + \boldsymbol{A_{i,m}S_m} \tag{1.5}$$

Una vez hemos planteado el problema de la BSS queda por definir la forma de solucionarlo.

# Capítulo 2

# Técnicas de Separación Ciega de Fuentes

## 2.1 Un enfoque múltiple

Como se expone en el capítulo anterior, la identificación de los componentes que integran una mezcla y sus correspondientes proporciones en ella no es más que un problema de separación ciega de fuentes y como tal ha de ser tratado. En este tipo de problemas, tanto la matriz de mezcla **A** como las fuentes **S** son desconocidas y deben estimarse conjuntamente. La separación de fuentes es fundamentalmente una cuestión de contraste y diversidad para separar los componentes de un conjunto, ya que dependiendo de las características de las fuentes originales que empleemos para diferenciarlas podremos clasificar las técnicas de separación en diferentes tipos. La mayoría de las técnicas BSS se pueden clasificar en cuatro clases principales [Stark et alt., 2010].

1. Por un lado, encontramos las técnicas que explotan la diversidad estadística de las fuentes [Hyvärinen et al., 2000]. Estos métodos asumen que las fuentes poseen algún tipo de independencia estadística entre sí y hacen uso de esta característica para poder diferenciar unas de otras. Aunque esta clase de métodos ya han proporcionado resultados exitosos en una amplia gama de aplicaciones, poseen algunas limitaciones que impiden su uso en cierto tipo de problemas. ICA (Independent Component Analysis) es uno de los métodos más empleados de esta clase y el que será tratado más en profundidad en este capítulo.

2. Otro tipo de técnicas son las que emplean la dispersión de las fuentes como elemento de contraste. Estas técnicas entienden que una fuente es dispersa cuando la mayoría de sus muestras son cero o próximas a cero y solo un bajo porcentaje de estas toman valores significativos. Este conocimiento a priori de las fuentes permite su separación en problemas BSS subdeterminados (cuando existen menos observaciones que fuentes) [Bofill and Zibulevsky, 2001]. Los algoritmos SCA (Sparse Component Analysis) son los que explotan esta vía de separación.

3. En los últimos años se ha venido desarrollando un tipo de técnicas que emplea la diversidad morfológica como característica diferenciadora a la hora de separar fuentes. Estas técnicas asumen que las fuentes son dispersas en alguna base determinada $\Phi$ o en diferentes diccionarios, esta dispersión hace que las fuentes sean morfológicamente distintas entre sí y que puedan ser separadas con precisión. De esta clase son los algoritmos GMCA (Generalized Morphological Component Analysis) [Stark et alt., 2010] que analizaremos posteriormente.

4. Por último, encontramos las técnicas que emplean la factorización de matrices no negativas NMF (Nonnegative Matrix Factorization) [Kim and Park, 2008]. Estas se basan en la no negatividad de las muestras de las fuentes a extraer y de sus matrices de mezcla. En el mundo real existen infinidad de escenarios en los que los datos son no negativos y sus componentes subyacentes solo tienen sentido físico cuando se da esta no negatividad.

No es el objetivo de este trabajo profundizar en la inmensa cantidad de técnicas y algoritmos desarrollados para resolver los problemas BSS, sin embargo, su finalidad es dar una visión general de los métodos más ampliamente utilizados para este cometido y así fundamentar las razones que llevan a seleccionar una familia específica de ellos para este caso concreto.

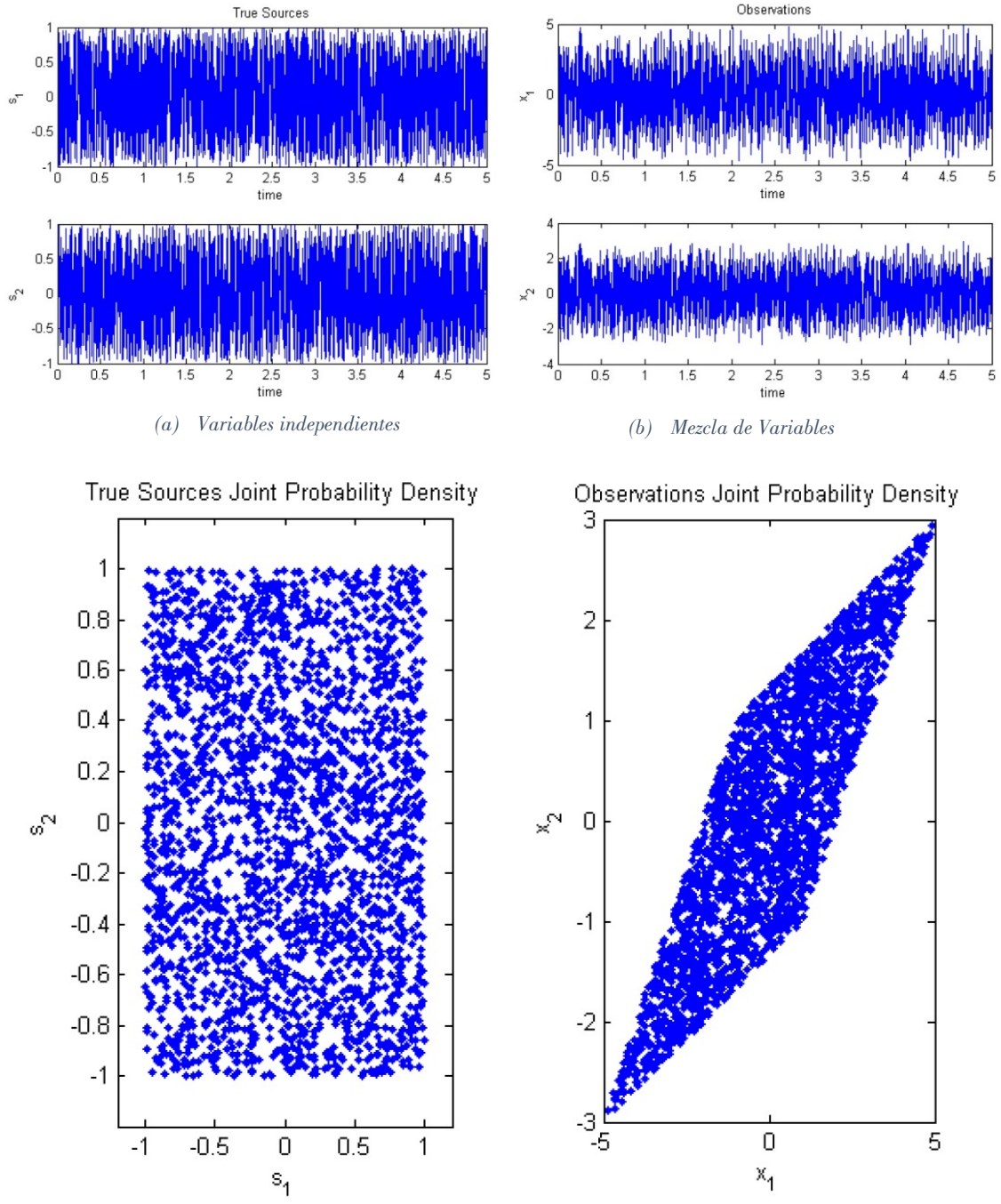
## 2.2 Análisis de Componentes Independientes (ICA)

El Análisis de Componentes Independientes (ICA) es una técnica desarrollada para extraer una representación de cada una de las distintas componentes o señales independientes entre sí que forman parte de una señal de mezcla.

El modelo ICA asume que las mezclas observadas están compuestas por una combinación lineal de componentes tal y como muestran las ecuaciones 1.2 y 1.3. Por lo tanto, su objetivo será estimar tanto la matriz $\mathbf{A}$ como los componentes $\mathbf{S}$ a partir de dichas mezclas. De manera alternativa, definimos ICA como el problema de obtención de la transformación lineal dada por la matriz $\mathbf{B}$ tal que las variables aleatorias estimadas en la matriz $\mathbf{Y}$ mediante la ecuación 1.4 sean tan independientes como sea posible [Hyvärinen et al., 2001]. Una vez se obtiene la matriz $\mathbf{B}$, la matriz $\mathbf{A}$ se obtiene como su inversa.

Podemos ilustrar de forma sencilla el funcionamiento de ICA haciendo uso de su definición gráfica recogida en [Hyvärinen and Oja, 2000] que se ilustra en la figura 2.1. En ella se consideran dos variables aleatorias generadas con distribución uniforme y rango [-1,1] denominadas $\mathbf{S} = [\boldsymbol{s_1}, \boldsymbol{s_2}]^{\boldsymbol{T}}$ . Estas dos variables se muestran en la figura 2.1(a) junto a su función de densidad de probabilidad 2.1(c) donde vemos como ambas son independientes entre sí ya que, si fijamos el valor de una de ellas, no podemos determinar el valor de la otra.

Si consideramos una matriz de mezcla $\mathbf{A} = \begin{pmatrix} \mathbf{2} & \mathbf{3} \\ \mathbf{2} & \mathbf{1} \end{pmatrix}$, mediante la combinación lineal $\mathbf{AS}$ obtenemos dos nuevas variables aleatorias $\mathbf{X} = [\boldsymbol{x_1}, \boldsymbol{x_2}]$ como las que pueden verse en la figura 2.1(b), las cuales ya no son independientes entre sí ya que ambas tienen información de $\mathbf{S}$ cómo se observa en su función de densidad de probabilidad conjunta ilustrada en la figura 2.1(d).




*(a) Variables independientes*

*(b) Mezcla de Variables*

*(c) Densidad de probabilidad S*

*(d) Densidad de probabilidad X*

*Figura 2.1 Representación gráfica del método ICA [Hyvärinen and Oja, 2000]*

La gráfica que muestra la figura 2.1(d) ilustra de forma intuitiva la manera en la que ICA obtiene la matriz **A,** ya que las direcciones seguidas por los ejes del paralelogramo representado coinciden con las columnas de **A**.

ICA es un método empleado en infinidad de problemas por lo que son innumerables los algoritmos que se han desarrollado basados en él. Aunque la mayoría de los algoritmos

basados en ICA son similares tanto en la teoría como en la práctica [Lee et alt., 2000], podemos citar dos como los más empleados en BSS:

- **FastICA**. Es un algoritmo desarrollado por Aapo Hyvärinen y Erkki Oja [Hyvärinen and Oja, 2000] llamado así por su rapidez con respecto a otros algoritmos. Es una adaptación de un algoritmo de punto fijo de una regla de aprendizaje de una red neuronal, encontrando las componentes independientes una a una. Utiliza técnicas estadísticas y analíticas, llegando a utilizar estadísticos de 4º orden.
- **JADE**. (Joint Approximate Diagonalization of Eigenmatrices). Fue desarrollado por J.F. Cardoso [Cardoso and Souloumiac, 1993]. Al igual que FastICA, se trata de un algoritmo de tipo estadístico que utiliza técnicas analíticas y estadísticas para obtener las componentes independientes utilizando también estadísticos de hasta 4º orden, aunque, a diferencia de este, JADE no realiza una estimación del número de componentes independientes en el conjunto de señales observadas, si no que permite identificar el número de componentes independientes en que se desean separar.

### 2.2.1 Limitaciones de ICA

El método ICA ha sido ampliamente utilizado en los últimos años para resolver problemas BSS ya que se adapta a una amplia variedad de escenarios. No obstante, cabe destacar una serie de restricciones que es necesario observar para que el modelo que plantea ICA tenga solución. Estas consideraciones son las siguientes [Hyvärinen et al., 2001]:

- Los componentes han de ser estadísticamente independientes. Este es el principio fundamental sobre el que se desarrolla ICA, ya que utiliza esta independencia para generar la separación de estos.
- El número de observaciones ha de ser mayor o igual al número de componentes independientes a estimar. Por lo tanto, ICA solo será un método útil de separación en problemas con sobredeterminación.
- ICA considera que la matriz de mezcla ha de ser cuadrada. Esta característica facilita la obtención de $\mathbf{B}$ mediante la matriz inversa $\mathbf{A^{-1}}$.
- Los componentes a separar han de presentar distribuciones no-gaussianas. ICA se basa en estadísticos de orden superior que toman valor nulo para distribuciones gaussianas, por lo que el método no es viable para la separación de componentes con este tipo de distribución.

Existen también dos aspectos a considerar a la hora de aplicar ICA para la resolución de problemas BSS [Hyvärinen and Oja, 2000]. Por una parte, ICA no puede determinar el orden en el que los componentes son separados. Esto se debe al desconocimiento previo de las matrices $\mathbf{A}$ y $\mathbf{S}$ siendo posible los intercambios posicionales tanto de las columnas de $\mathbf{A}$ como de las filas de $\mathbf{S}$ sin que por ello varíe el resultado final. Por otro lado, ICA tampoco puede determinar las energías de los componentes, esto introduce incertidumbre en la

escala de los componentes recuperados siendo posible incluso un cambio de signo en los mismos.

## 2.3 Análisis de Componentes Dispersos (SCA)

Durante años ICA ha sido la herramienta preferida para abordar problemas de BSS, pero como hemos visto en el punto anterior, cuando tratamos problemas con subdeterminación (más componentes que mezclas) o cuando las mezclas contienen componentes ruidosos con distribución gaussiana, se hace necesario utilizar algún otro elemento de separación que no sea la independencia estadística de las fuentes.

SCA (Sparse Component Analysis) hace uso de la dispersión de las fuentes, no solo para resolver el problema en el caso de mezclas ruidosas, sino que puede aplicarse a casos de BSS en escenarios de subdeterminación [Vielva et alt., 2001]. SCA asume que las fuentes pueden ser representadas por una combinación lineal de señales elementales $\boldsymbol{\varphi_k}$ llamadas átomos y que cada fuente a separar se puede representar mediante la ecuación (2.1) [Comon and Jutten, 2010].

$$s(t) = \sum_{k=1}^{K} c(k)\varphi_k(t) \tag{2.1}$$

Donde $c(k)$ corresponde a cada una de las muestras de la fuente y siendo el diccionario $\Phi$ una matriz de dimensiones **KxT** cuyas k líneas están compuestas por átomos $\varphi_k(t)$ siendo $1 \leq t \leq T$ y $k = 1, \dots, K$ pudiendo de este modo representar el problema SCA con la ecuación (2.2).

$$\boldsymbol{S} = \boldsymbol{C_s}\Phi \tag{2.2}$$

El funcionamiento básico de SCA consiste principalmente en la aplicación de cuatro pasos.

- El primer paso consiste en generar una representación dispersa de las mezclas mediante algún tipo de transformación lineal. Una aproximación de representación dispersa de una mezcla $\boldsymbol{x}$ tomará la siguiente forma $\boldsymbol{x} \approx \boldsymbol{C_x}\Phi$, donde:

$$C_x = \begin{bmatrix} c_{x1} \\ \cdots \\ c_{xp} \end{bmatrix} = [C_x(1) \quad \dots \quad C_x(K)] \tag{2.3}$$

  Para este tipo de transformaciones lineales se emplean generalmente Wavelets o transformadas de Fourier (STFT). Si mediante esta representación de las mezclas se consigue una dispersión suficiente, entonces el gráfico de dispersión

$\{C_x(k)\}_{k=1}^{K}$ estará compuestos de puntos casi alineados con las columnas de la matriz de mezcla como muestra la figura (2.2) y por tanto podrán ser utilizados para estimar **A** mediante clustering.

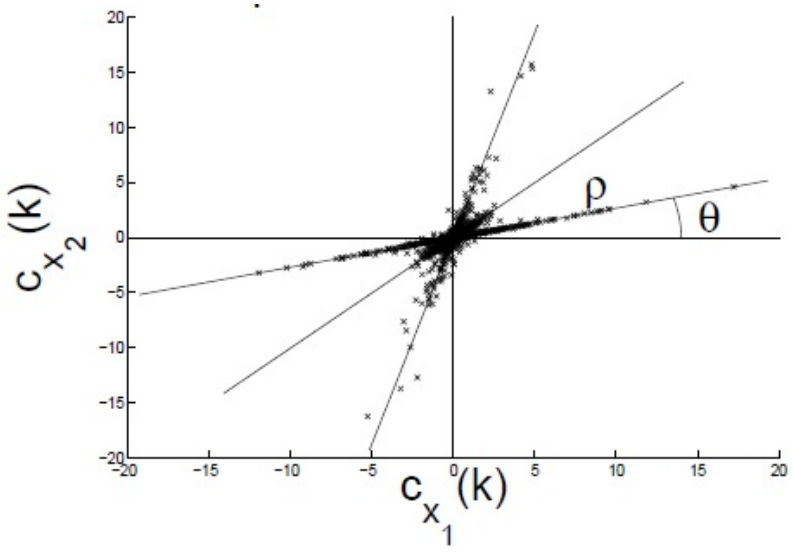



*Figura 2.2 Gráfico de dispersión Cx*

- El segundo paso es precisamente estimar la matriz de mezcla **A** a partir del gráfico de dispersión $C_x$, para ello se emplean generalmente diferentes variantes de algoritmos de clustering basados en K-medias. Para que estas técnicas funcionen bien debe darse la hipótesis de que a lo sumo una de las fuentes contribuya significativamente a cada punto del gráfico de dispersión. En la práctica esta hipótesis es, en el mejor de los casos, válida para la mayoría de los casos de diagrama de dispersión siempre que las fuentes no solo tengan una representación dispersa sino también disjunta [Baland and Rosca, 2000].
- El tercer paso consiste en estimar la representación de las fuentes asumiendo su dispersión. En un escenario sin ruido agregado la representación de las fuentes $\hat{C}_s$ puede estimarse del siguiente modo [Bofill and Zibulevsky, 2001]:

$$\hat{C}_s(k) \coloneqq \arg \min_{C|C_x(k)=\widehat{A}C} \|C\|_q, \quad q = 1 \tag{2.4}$$

  Lo que puede interpretarse como una estimación de máxima verosimilitud asumiendo que los coeficientes de las fuentes tengan una distribución Laplaciana.
- El último paso es la reconstrucción de las fuentes mediante la inversión de la transformación lineal realizada en el primer paso. Para transformaciones como la transformada discreta Wavelet o la transformada de Fourier, esta reconstrucción se obtiene simplemente mediante $\hat{\boldsymbol{s}} = \widehat{\boldsymbol{C}}_{\boldsymbol{s}}\Phi$.

Este método SCA basado en transformaciones lineales ha conducido al desarrollo de algoritmos muy eficaces en ciertos escenarios BSS, este es el caso del algoritmo DUET [Yilmaz and Rickard, 2004].

### 2.3.1 Limitaciones de SCA

Aunque SCA se ha mostrado como una herramienta muy exitosa para la separación de fuentes en escenarios subdeterminados o con ruido agregado, una de sus mayores limitaciones radica en la complejidad computacional de los algoritmos de optimización que utiliza.

Otro problema que se debe afrontar en el uso de SCA en BSS es la apropiada elección del diccionario $\Phi$. En ocasiones resulta de gran complejidad encontrar un diccionario común en el cual las fuentes estén dispersamente representadas.

Por último, cabe destacar que, como sucede en el caso de ICA, no resulta sencillo limitar la no-negatividad de las fuentes y la matriz de mezcla, suponiendo un problema para su utilización en cierto tipo de escenarios de separación de fuentes.

## 2.4 Análisis de Componentes Morfológicos (GMCA)

Otro método enfocado en la dispersión de las fuentes para generar separación es el Análisis Generalizado de Componentes Morfológicos GMCA. Este método propuesto en [Bobin et alt., 2006] asume que las $\boldsymbol{N_s}$ fuentes $(\boldsymbol{s_i})_{\boldsymbol{i=1....N_s}}$ que componen una mezcla son dispersas, no solo en una base o diccionario como se asume en SCA, sino en diferentes diccionarios. Este modelo aprovecha el hecho de que las fuentes sean morfológicamente distintas para diferenciarlas con precisión.

El problema a resolver se formula del siguiente modo. GMCA asume que las fuentes a separar son dispersas en lo que podemos llamar un diccionario espacial $\Phi$ formado por la concatenación de un número $\boldsymbol{K}$ de bases ortonormales $(\Phi_{\boldsymbol{k}})_{\boldsymbol{k=1....K_s}}$ donde $\Phi = [\Phi_{\boldsymbol{1}}, \dots, \Phi_{\boldsymbol{K}}]$. Por lo tanto, en GMCA se entiende que cada fuente $s_i$ está modelada mediante una combinación lineal de $\boldsymbol{K}$ componentes morfológicos, los cuales son dispersos en una base o diccionario específico, como expresa la ecuación (2.5)

$$s_i = \sum_{k=1}^{K} x_{i,k} = \sum_{k=1}^{K} \Phi_{\boldsymbol{k}} \alpha_{i,k} = \Phi \alpha_i \qquad (2.5)$$

GMCA busca un esquema de separación, a través de la estimación de la matriz A, que conduzca a las fuentes más dispersas S en cada diccionario Φ. Esto se expresa mediante el problema de optimización recogido en la ecuación (2.6).

$$\min_{A,\alpha_{1,1},\ldots,\alpha_{N_S,K}} \frac{1}{2}\left\|Y - A\alpha\Phi^T\right\|_F^2 + \lambda \sum_{i=1}^{N_s}\sum_{k=1}^{K}\left\|\alpha_{i,k}\right\|_p^p \quad s.t \ \ \|\alpha_i\|_2 = 1 \tag{2.6}$$

La ecuación (2.6) define un problema de optimización no convexo notoriamente complejo. Aplicándolo a escenarios de BSS y siguiendo la ecuación (1.3) podemos descomponer el producto $\boldsymbol{AS}$ en $\boldsymbol{N_s \cdot K}$ componentes morfológicos como se muestra en (2.7).

$$AS = \sum_{i,k} a_y \, s_{i,k}^T = \sum_{i,k} (a_y \, \alpha_{i,k}^T)\Phi_k^T \tag{2.7}$$

Basándose en esta descomposición, GMCA produce un algoritmo de minimización para estimar iterativamente un término cada vez [Bobin et alt., 2007].

El método GMCA se adapta muy bien a escenarios donde se manejan datos contaminados con ruido gaussiano, asumiendo que la matriz $\boldsymbol{N}$ de la ecuación (1.3) es ruido blanco añadido a cada mezcla estimándolo igual para cada una de ellas, simplificando así su eliminación.

Al igual que en el caso de SCA la complejidad de resolver el problema de optimización de la ecuación 2.6 es un hándicap para su utilización. Por este motivo, uno de los algoritmos más empleados de este método es el FastGMCA cuyo objetivo es reducir la complejidad computacional del método original. Para ello asume que el diccionario Φ no es redundante y lo reduce a una única ortobase (por ejemplo, *K=1*), transformando así el problema de optimización en uno más sencillo, recogido en la ecuación (2.8).

$$\min_{A,\alpha} \frac{1}{2}\|\beta - A\alpha\|_F^2 + \lambda \sum_{i=1}^{N_s}\|\alpha_i\|_p^p \quad s.t \ \ \|\alpha_i\|_2 = 1 \tag{2.8}$$

Siendo $\boldsymbol{\beta = Y\Phi}$, consiguiendo que el algoritmo ya no necesite aplicar los operadores de análisis y síntesis en cada iteración ya que solo las mezclas $\boldsymbol{Y}$ tienen que transformarse una vez en Φ.

### 2.4.1 Limitaciones de GMCA

Al igual que con el método SCA la principal limitación en el uso del método GMCA radica en la dificultad de encontrar los diccionarios o bases para representar de forma dispersa los diferentes componentes y conseguir así una diversidad morfológica suficiente que permita su separación precisa.

Su otra principal limitación es común a los métodos presentados anteriormente y consiste en la dificultad de limitar la no negatividad de las fuentes y la matriz de mezcla, aunque, como se verá más adelante, esta limitación puede solventarse mediante el uso de algoritmos híbridos.

## 2.5 Factorización de Matrices No Negativas (NMF)

La Factorización de Matrices No Negativas (Non-Negative Matrix Factorization, NMF), consiste en la descomposición de una matriz como producto de dos o más matrices. La única restricción que exige este método es que todos los coeficientes de las matrices han de ser positivos. Las primeras referencias que se tienen sobre NMF son de unos trabajos publicados en [Paatero and Tapper, 1994], donde se expone el método como una variante de la Factorización Positiva de Matrices (PMF), aunque fue con los trabajos de [Lee and Seung, 1999, 2001] publicados en *Nature and NIPS* cuando ganó popularidad, ya que estos aportaron los primeros algoritmos de aplicación. En la actualidad, NMF es uno de los métodos más usados en BSS.

La principal diferencia de NMF respecto a otros métodos de factorización, es la no negatividad de sus coeficientes. Muchos datos del mundo real son no negativos y sus componentes solo tienen significado físico cuando son positivos. Esto ocurre en varios campos como el tratamiento de imagen y vídeo, economía y por supuesto en el de las mezclas multicomponente.

NMF es un modelo aditivo, en el que un valor cero representa la ausencia de componentes de la magnitud con la que se esté tratando y un número positivo representa la presencia de alguna componente, lo que permite que cada una de las partes que conforman la suma pueda ser considerada como parte de los datos originales. Gracias a esto, podemos mantener un buen equilibrio entre la interpretabilidad de los datos y la fidelidad estadística de los mismos, hecho que hace al método óptimo para nuestro trabajo.

El problema básico de NMF se puede expresar de la siguiente manera. Dada una matriz de coeficientes no negativos $\mathbf{Y}$ de dimensiones $\boldsymbol{JxT}$ donde $\mathbf{Y} \geq \mathbf{0}$, y un rango reducido $\boldsymbol{J}$ donde $\boldsymbol{J} \leq \boldsymbol{min(I,T)}$, el objetivo es encontrar dos matrices $\boldsymbol{A}$ y $\boldsymbol{B}$ no negativas donde $\boldsymbol{A} = [\boldsymbol{a_1}, \boldsymbol{a_2}, \dots, \boldsymbol{a_J}]$ sea de dimensiones $\boldsymbol{IxJ}$, y $\boldsymbol{S} = \boldsymbol{B^T} = [\boldsymbol{b_1}, \boldsymbol{b_2}, \dots, \boldsymbol{b_J}]$ sea de dimensiones $\boldsymbol{JxT}$, tales que factoricen $\mathbf{Y}$ lo mejor posible, eso es:

$$\mathbf{Y} = \mathbf{AS} + \mathbf{E} = \mathbf{A}\boldsymbol{B^T} + \mathbf{E} \tag{2.9}$$

donde la matriz $\mathbf{E}$ representa el error aproximado en la descomposición, la matriz $\boldsymbol{A}$ representa la matriz de mezcla y $\boldsymbol{S}$ los componentes fuente. Las matrices $\boldsymbol{A}$ y $\mathbf{S}$ pueden tener diferentes sentidos físicos, dependiendo de la aplicación.

En NMF estándar, solo se asume la no negatividad de las matrices $\boldsymbol{A}$ y $\boldsymbol{S}$. Al contrario que en los métodos para BSS basados en ICA, en este caso no se asume la independencia de las fuentes, en cambio, se introducen otras suposiciones y restricciones para $\boldsymbol{A}$ y $\boldsymbol{S}$ cómo se mostrará en el siguiente capítulo de este trabajo. Esta simetría en las suposiciones conduce a una simetría en la factorización, por lo que podríamos simplemente escribir $\boldsymbol{Y^T \approx S^T A^T}$ propiciando así que a menudo el significado de "fuente" y "mezcla" en NMF sea algo arbitrario.

El modelo NMF también puede ser representado como una forma especial del modelo bilineal, donde los vectores son no negativos como ilustra la figura 2.3 recogida en la ecuación (2.10) [Cichocki et alt., 2009].

$$Y = \sum_{j=1}^{J} a_j \diamond b_j + E = \sum_{j=1}^{J} a_j \, b_j^T + E \tag{2.10}$$

donde el símbolo $\diamond$ representa el producto externo de dos vectores. Por lo tanto, podemos construir una representación aproximada de la matriz de datos no negativos **Y**, como una suma de matrices no negativas de rango unidad $a_j b_j^T$.

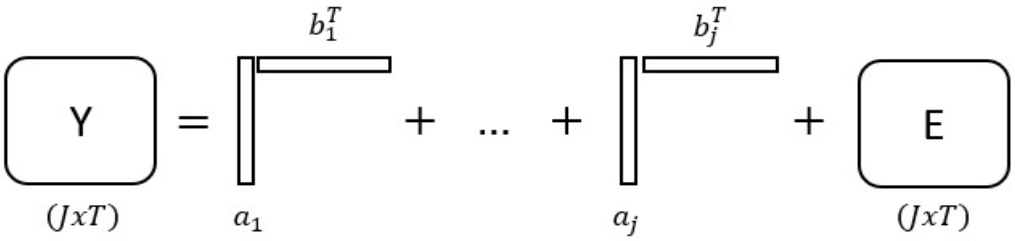



*Figura 2.3 Modelo NMF bilineal. La aproximación de la matriz de datos Y se representa como una suma de matrices no negativos de rango unidad $a_j b_j^T$*

### 2.5.1 Limitaciones de NMF

Los métodos basados en NMF poseen dos limitaciones básicas. Por un lado, son muy sensibles a las condiciones de inicialización de sus algoritmos. La solución y la convergencia dada por los algoritmos NMF normalmente dependen mucho de las condiciones iniciales, es decir, sus valores iniciales supuestos. Por ello, es importante inicializar de forma eficiente la matriz A. En otras palabras, la eficiencia de la mayoría de las estrategias NMF se ve claramente afectada por la selección de las matrices iniciales [Cichocki et alt., 2009].

La otra limitación fundamental de estos métodos reside en la no unicidad de sus soluciones. Generalmente, la estimación en NMF se realiza mediante la minimización de una o varias funciones objetivo. Sin embargo, en general, estas minimizaciones no garantizan una solución única.

## 2.6 Adaptación de los métodos al caso particular

En los puntos anteriores de este capítulo se han propuesto varios métodos de resolución del problema BSS. Este trabajo versa sobre un problema particular que posee unas características propias que hacen que no todos los esquemas sean válidos para su resolución. En este punto se propone el uso de los algoritmos NMF y la familia de algoritmos híbridos nGMCA para la resolución del problema de la separación de componentes en mezclas, ilustrando, mediante un ejemplo sencillo, las dos principales características que hacen de ellos las herramientas mejor adaptadas a este escenario.

La familia de protocolos nGMCA se compone de varios algoritmos que hacen uso de NMF y GMCA para generar separación de fuentes. Esta familia, al igual que los métodos NMF, posee dos características que la hacen adecuada para este trabajo. Por un lado, son capaces de restringir la descomposición de las fuentes al ámbito de la no-negatividad, esto es fundamental ya que tanto los componentes originales de las mezclas como las propias mezclas en sí carecen de sentido si sus magnitudes adoptan valores negativos. Por otra parte, no se ven afectados por la dependencia estadística de las fuentes originales ya que utilizan la diversidad morfológica de estas para diferenciarlas.

### 2.6.1 Restricción de no-negatividad

Para evaluar la adaptación de los métodos para la resolución de BSS expuestos en este tema al problema que plantea la separación de componentes en mezclas, vamos a analizar el comportamiento de algunos de sus principales algoritmos en un experimento simple.

Para esta evaluación utilizaremos los siguientes algoritmos:

- El algoritmo **SCA** recogido en [Yuanqing et alt., 2003], que ilustra el método fundamental del análisis de componentes dispersos.
- El algoritmo **JADE** descrito en [Cardoso and Souloumiac, 1993] que representa uno de los más empleados de los métodos ICA.
- El algoritmo **FastGMCA** detallado en [Stark et alt., 2010], que implementa la versión menos compleja del algoritmo GMCA.
- El algoritmo **NMF** basado en la factorización por mínimos cuadrados propuesto en [Kim and Park, 2008].
- El algoritmo **nGMCA**$^{\text{Standard}}$ [Rapin et alt., 2013], que combina los métodos NMF y GMCA.

El objetivo de esta prueba es comprobar el comportamiento de los diferentes algoritmos ante un supuesto en el que tanto las fuentes originales, como la matriz de mezcla tienen valores positivos. Por lo tanto, sería deseable que el resultado de la separación estuviera limitado también a valores positivos, ya que las magnitudes físicas del caso particular que nos ocupa así lo requieren. Estos resultados pueden reproducirse ejecutando el Script de Matlab '*Test_A*' que se adjunta a esta memoria.

Se parte de un conjunto $\mathbf{S} = [s_1, s_2]$ que contiene los espectros de dos componentes (Taurina y Ácido Fólico) generados artificialmente basándose en los datos recogidos en la SDBS (Spectral Database for Organic Compounds) del AIST (National Institute of Advanced Industrial Science and Technology) de Japón. Posteriormente se genera, mediante la ecuación (1.3), con un escenario en ausencia ruido $\boldsymbol{N} = 0$, un conjunto de mezclas $\boldsymbol{Y} = [y_1, y_2, y_3, y_4]^T$ a partir de una matriz de mezcla $\mathbf{A} = [a_1, a_2, a_3, a_4]$ como se muestra en la tabla 2.1. Tanto los espectros de los componentes, como de las mezclas se ilustran en la figura 2.4.

| $A = [a_1, a_2, a_3, a_4]$ | ***Taurina*** | ***Ácido Fólico*** |
|---|---|---|
| ***Mezcla 1*** | *0.15* | *0.85* |
| ***Mezcla 2*** | *0.50* | *0.50* |
| ***Mezcla 3*** | *0.80* | *0.20* |
| ***Mezcla 4*** | *0.30* | *0.70* |

*Tabla 2.1 Matriz de mezcla A.*

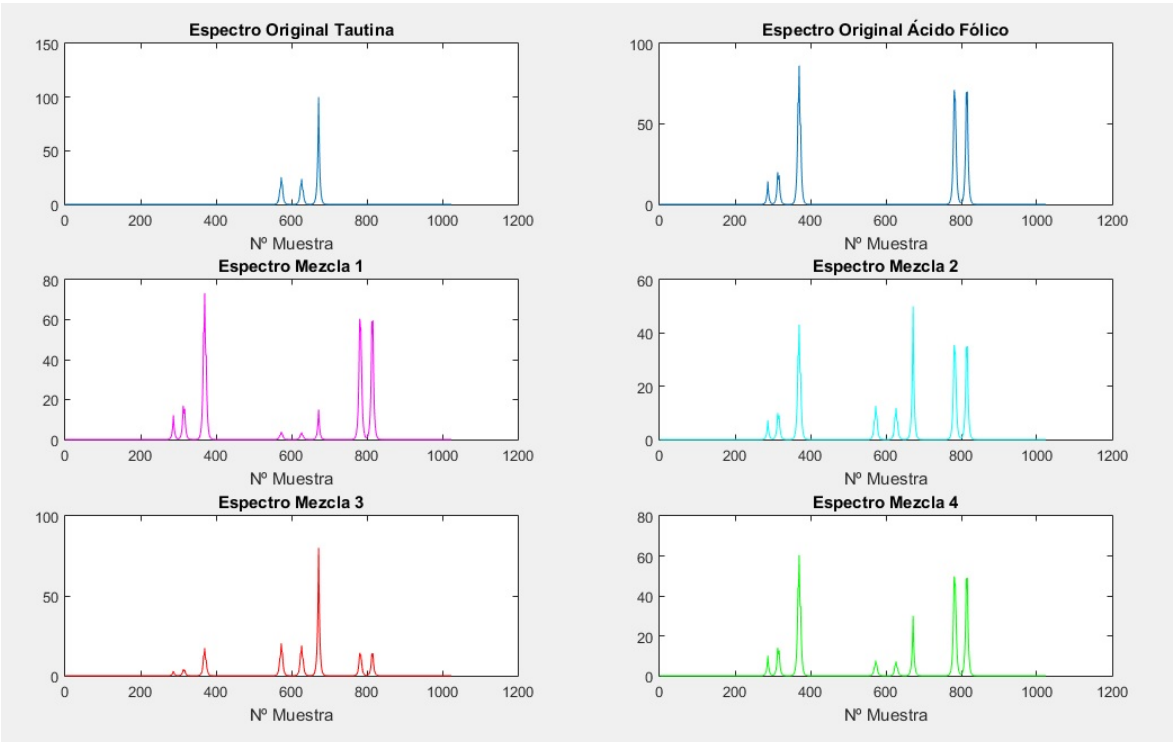



*Figura 2.4 Datos originales de espectros $\boldsymbol{S} = [s_1, s_2]$ y mezclas $\boldsymbol{Y} = [y_1, y_2, y_3, y_4]$*

Se realiza la separación ciega de fuentes empleando los algoritmos citados anteriormente con el siguiente resultado:

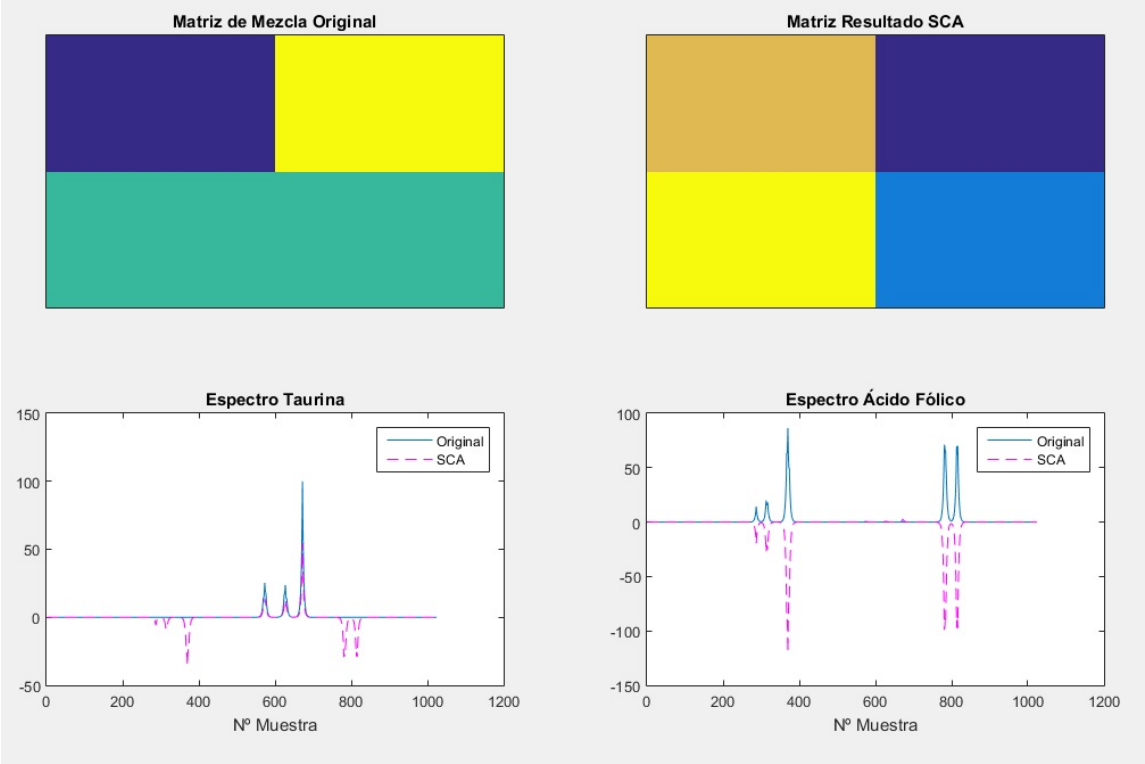



*Figura 2.5 Resultados de la separación mediante algoritmo SCA*

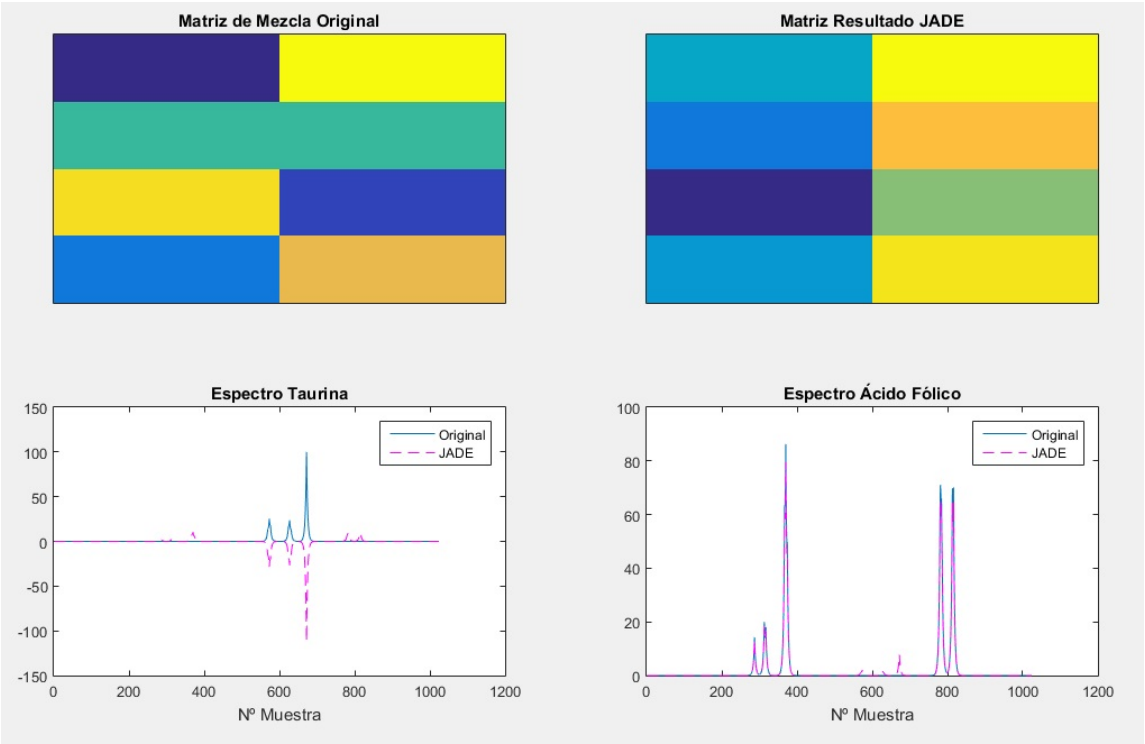



*Figura 2.6 Resultados de la separación mediante algoritmo JADE*

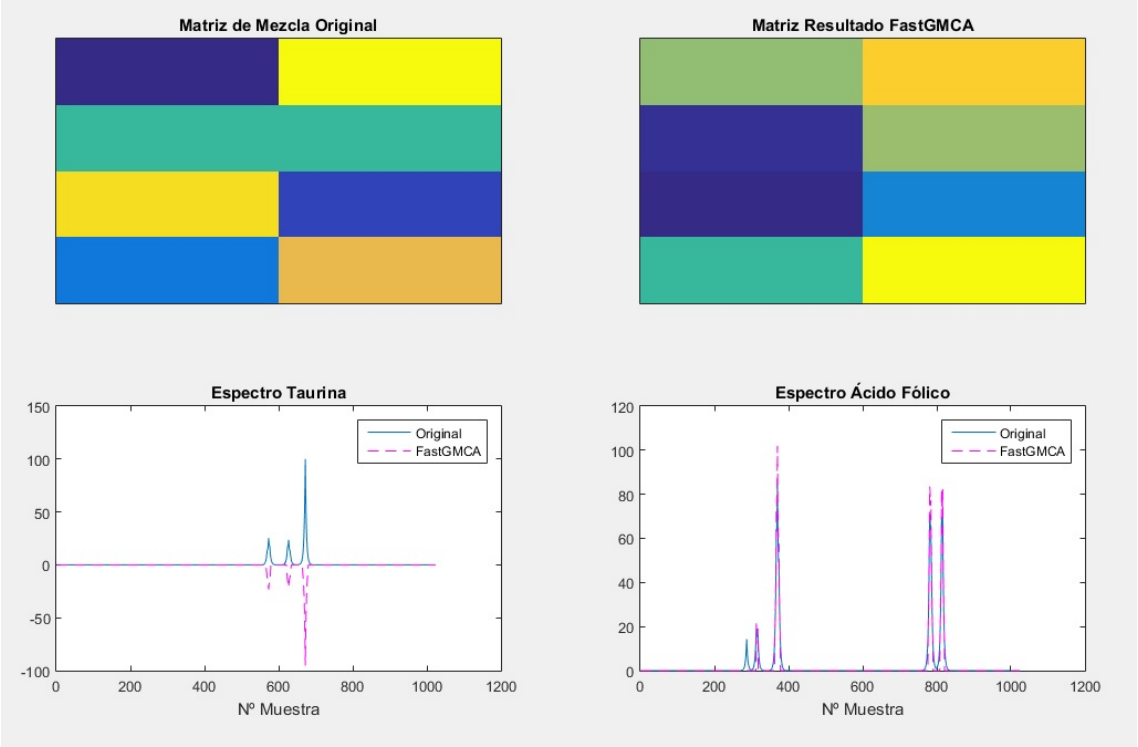



*Figura 2.7 Resultados de la separación mediante algoritmo FastGMCA*

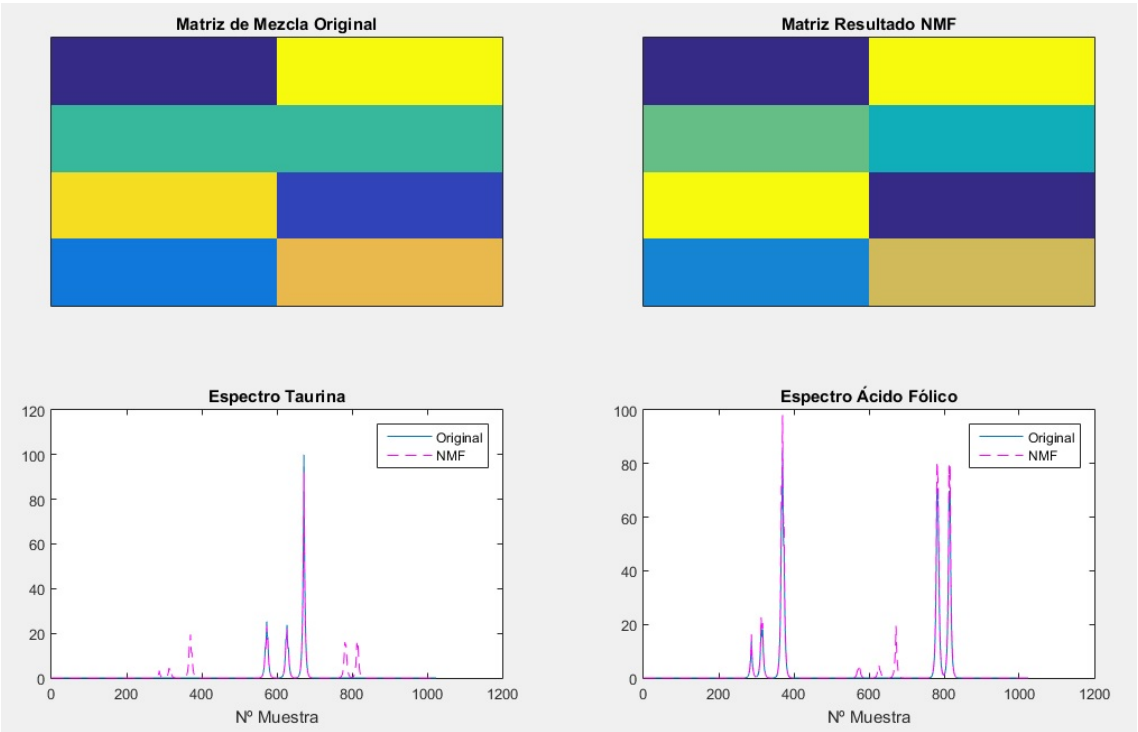



*Figura 2.8 Resultados de la separación mediante algoritmo NMF*

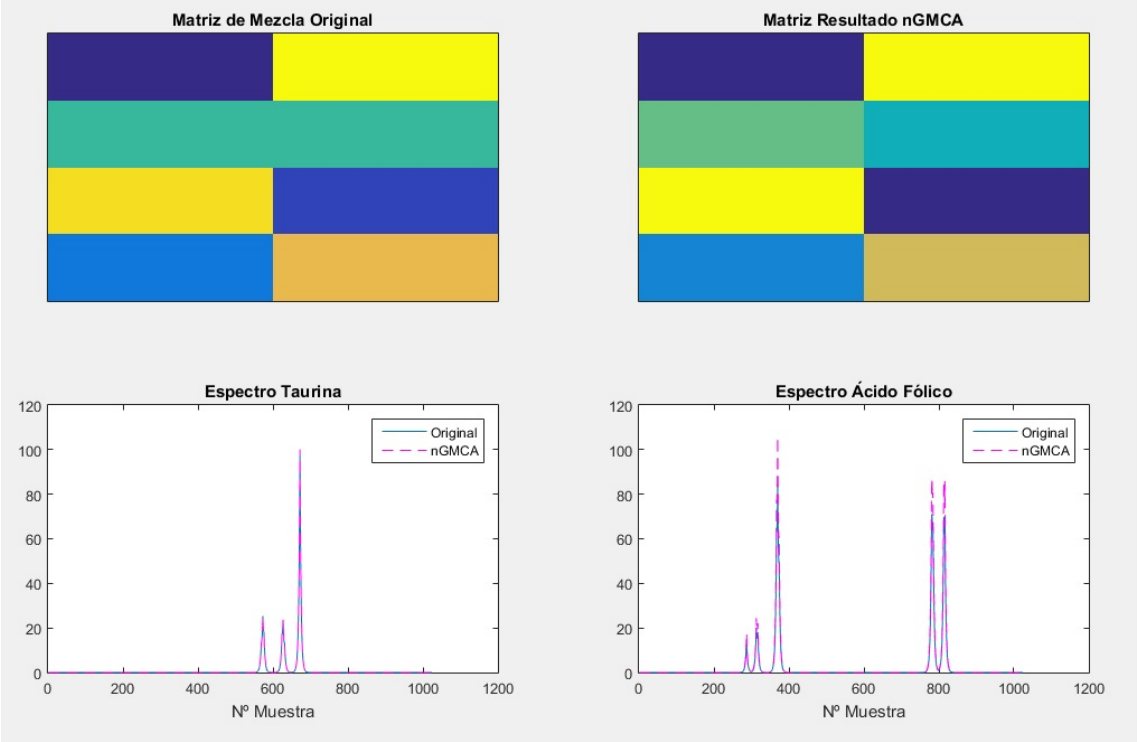



*Figura 2.9 Resultados de la separación mediante algoritmo nGMCA*

Las figuras anteriores nos muestran los resultados de la separación ciega usando diferentes técnicas. En ellas puede comprobarse la similitud entre la matriz de mezcla original y la recuperada de la separación mediante la comparación de la tonalidad de sus secciones. En estas figuras, también encontramos la comparación entre los espectros originales de los componentes de las mezclas y sus respectivos componentes recuperados. Se ha querido emplear componentes con un espectro simple para ilustrar el caso más básico de separación y así poder anticipar los resultados que podrían obtenerse con componentes y mezclas más complejas. Cabe puntualizar también, que en el caso del algoritmo SCA se ha empleado una matriz de mezcla de 2x2 correspondiente a $\mathbf{A} = [a_1, a_2]$ ya que se ha utilizado un algoritmo para separaciones con subdeterminación, aunque este hecho no desvirtúa la conclusión del análisis desarrollado.

Obviando la calidad de la separación realizada por cada algoritmo, se observa en las figuras 2.5, 2.6 y 2.7 cómo, en el caso de la Taurina, se han estimado valores negativos para su espectro. Esto se debe a que los algoritmos citados no restringen estos valores a la hora de estimar la separación, hecho que no los convierte en los más adecuados en el escenario que trata este trabajo. Por su parte, las figuras 2.8 y 2.9 muestra la separación obtenida mediante el algoritmo NMF y el método más básico de la familia de los nGMCA, que muestran resultados más coherentes con la realidad física del problema. Esto los convierte en buenos candidatos para llevar a cabo la separación de componentes en mezclas.

### 2.6.2 Independencia estadística de las fuentes

Aunque a priori hayamos descartado los métodos ICA para este trabajo, debido a sus problemas a la hora de tratar con la restricción de no negatividad, lo cierto es que en [Plumbley, 2003] y [Oja and Plumbley, 2004] se propone un conjunto de algoritmos que tienen como objetivo conseguir una separación no negativa a partir de métodos ICA, por lo que la no negatividad ya no constituiría un impedimento a la hora de emplear ICA para el escenario de este trabajo. Sin embargo, ICA obtiene el factor diferenciador de la independencia estadística de las fuentes, y este es un hecho que no se puede dar por supuesto en el escenario de las mezclas multicomponentes.

Para ilustrar este punto podemos medir la independencia estadística de algunos de los componentes que se utilizarán en la evaluación posterior de los algoritmos. Para realizar la medición emplearemos la Distancia de Correlación ($dCorr$) [Mur et alt., 2017] definida en la ecuación (2.11), que mide la dependencia estadística entre dos variables X e Y. Atendiendo a la distancia de correlación, diremos que dos variables X e Y se considerarán totalmente independientes cuando $dCorr(X,Y) = 0$.

$$dCorr^2(X,Y) = \frac{dCov^2(X,Y)}{(dVar^2(X) \cdot dVar^2(Y))^{1/2}} \tag{2.11}$$

En la figura 2.10 se muestra un conjunto $\mathbf{S} = [s_1, s_2, s_3, s_4]$ compuesto por los espectros IR de cinco disolventes que formarán parte de los conjuntos de evaluación posterior. La distancia de correlación de estos cinco componentes se muestra en la tabla 2.2. Como puede observarse en ella, existe una elevada dependencia estadística entre los espectros analizados, esto descarta de manera definitiva cualquier método ICA para la separación de estos componentes. Estas medidas pueden reproducirse ejecutando el Script de Matlab '*Test_B*' que se adjunta a esta memoria.

| $SdCorr(s_i, s_j)$ | ***Butanol*** | ***Cloroformo*** | ***Metanol*** | ***Propanol*** | ***Tolueno*** |
|---|---|---|---|---|---|
| ***1-Butanol*** | - | *0.3395* | *0.8779* | *0.8053* | *0.5124* |
| ***Cloroformo*** | - | - | *0.3153* | *0.2750* | *0.5831* |
| ***Metanol*** | - | - | - | *0.7658* | *0.4260* |
| ***2-Propanol*** | - | - | - | - | *0.3886* |
| ***Tolueno*** | - | - | - | - | - |

*Tabla 2.2 Medidas de Independencia Estadística de disolventes*

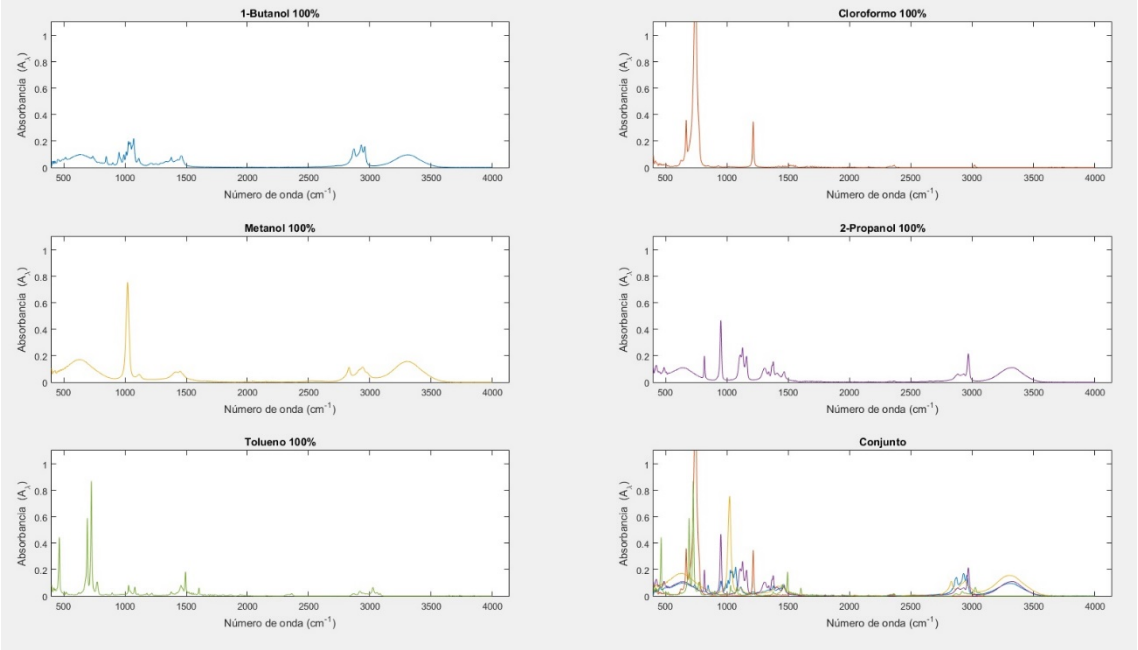



*Figura 2.10 Espectros IR de disolventes* $\boldsymbol{S} = [s_1, s_2, s_3, s_4, s_5]$

Por lo tanto, podemos concluir que los mejores candidatos para la resolución del problema BSS planteado en este trabajo son aquellos métodos que hacen uso de la factorización de matrices no negativas. Tanto los métodos originales NMF, como los nuevos métodos híbridos nGMCA que añaden la dispersión de las fuentes a las restricciones para realizar la separación, son métodos adecuados para realizar la separación de componentes en mezclas.

En el siguiente capítulo se mostrará un breve resumen de los diferentes algoritmos basados en las técnicas comentadas anteriormente, cuyo desempeño se evaluará en el capítulo 4.

# Capítulo 3

# NMF y Dispersión

## 3.1 Introducción

Las restricciones sobre la dispersión de las fuentes ya han demostrado su eficacia para resolver una amplia gama de problemas. En el contexto de BSS, se ha demostrado que la dispersión aumenta la diferenciabilidad entre las fuentes, lo que ayuda en gran medida a su separación [Bobin et alt., 2006]. Simplificando, entenderemos que una fuente es dispersa cuando la información que contiene se concentra en solo unos pocos coeficientes distintos de cero. Sin embargo, la dispersión de las fuentes depende en gran medida de la base o el diccionario en el que se representan. Por ejemplo, una onda sinusoidal será dispersa en el dominio de Fourier ya que puede codificarse con un coeficiente en este dominio, mientras que en el dominio directo la mayoría de sus coeficientes son distintos de cero. En este capítulo se mostrará la incorporación de la dispersión como elemento restrictivo a los algoritmos que hacen uso de la factorización no negativa para la resolución del problema BSS, y como, de este modo, se ha favorecido la aparición de una familia de protocolos híbridos que incorporan restricciones, tanto de no negatividad, como de dispersión en el dominio de transformación.

## 3.2 Evolución de NMF hacia algoritmos dispersos

Como hemos visto en la ecuación 1.3, en BSS, el modelo de mezcla lineal instantánea supone que las mediciones son mezclas lineales de fuentes más algo de ruido añadido. Sin embargo, es común tener alguna información previa sobre las fuentes $\boldsymbol{S}$ y la matriz de mezcla $\boldsymbol{A}$. En el contexto de NMF, tanto las entradas de los coeficientes de mezcla $\boldsymbol{A}$, como las de las fuentes $\boldsymbol{S}$ se suponen no negativas. Esta suposición surge de manera natural en muchas aplicaciones, como la minería de texto, el procesamiento de audio, las imágenes hiperespectrales o la espectrometría IR. De hecho, los espectros a menudo se miden como intensidades (espectros electromagnéticos, por ejemplo) o en términos de un número entero de elementos (moléculas en espectrometría de masas) que son necesariamente no negativos. Los coeficientes de mezcla generalmente son función de las concentraciones relativas de las entidades físicas observadas, que también son necesariamente no negativas. Por lo tanto, podemos formular el problema de la NMF como (3.1).

$$\arg \min_{A\geq 0, S\geq 0} \|Y - AS\|_2^2 \tag{3.1}$$

Pero encontrar una solución óptima para 3.1 es muy complejo ya que es un problema NP-Hard [Vavasis, 2009]. Debido a que puede haber muchos mínimos locales, y debido a que el modelo de mezcla puede ser imperfecto, agregar algunas prioridades al problema de optimización suele ser beneficioso. De hecho, pueden ayudar a priorizar los mínimos con las propiedades deseadas. Por este motivo se han propuesto diferentes métodos para aprovechar las características estructurales de los datos a la hora de aplicar NMF para su separación.

El primero de estos métodos planteó el uso de la continuidad de las fuentes. En [Zdunek and Cichoki, 2007] se propuso la recuperación de fuentes con perfiles suaves y por lo tanto continúas agregando un término de suavizado al problema de la ecuación (3.1) como muestra (3.2).

$$\arg \min_{A\geq 0, S\geq 0} \|Y - AS\|_2^2 + \alpha U_\delta(S) \tag{3.2}$$

Donde $U_\delta(S)$ es la función de Green (3.3):

$$U_\delta(S) = \delta \sum_{i=1}^{r} \sum_{j=1}^{n} log\left(cosh\left(\frac{S_{i,j} - S_{i,j-1}}{\delta}\right)\right) \tag{3.3}$$

En este caso, la minimización se lleva a cabo utilizando actualizaciones multiplicativas. La regularización penaliza las diferencias grandes entre una muestra y sus vecinos y, por lo tanto, tiende a priorizar estimaciones suaves de las fuentes. Pero, ajustar los parámetros $\delta$ y $\alpha$ puede ser engorroso y esta regularización no es apropiada para señales con perfil puntiagudo como suele ser el caso de los espectros IR, tal y como muestran los espectros de la Taurina y el ácido fólico de la figura 2.5.

Posteriormente se han desarrollado otros métodos más adecuados para el escenario que plantea este trabajo, estos son los que hacen uso de la dispersión de las fuentes. En algunos de estos métodos, la dispersión se prioriza restringiendo aún más la norma $\ell_1$ de las fuentes de la manera muestra (3.4).

$$\arg \min_{A\geq 0, S\geq 0} \|Y - AS\|_2^2 + \lambda \|S\|_1 \tag{3.4}$$

En [Hoyer, 2002] se usa un descenso de gradiente para $\boldsymbol{A}$ y un procedimiento de actualización multiplicativa para $\boldsymbol{S}$. En [Zdunek abd CiChoki, 2007] se desarrolla el algoritmo HALS, donde las columnas de $\boldsymbol{A}$ y las filas de $\boldsymbol{S}$ se actualizan una por una, lo que conduce a actualizaciones simples y eficientes. Mas recientemente, se ha introducido una implementación acelerada de HALS [Kimura et alt., 2015], en esta versión el parámetro $\lambda$ se genera automáticamente para obtener una tasa de dispersión definida por el usuario.

También se han explorado otras formas de regularizaciones para imponer la dispersión de las fuentes en el dominio directo. En [Zdunek abd CiChoki, 2007] también se propone el uso de la penalización $\sum_{t=1}^{n}\left\|s_{.,t}\right\|_1^2$. Esta penalización tiende a favorecer soluciones donde una sola fuente domina en cada muestra. En [Hoyer, 2004] se propuso restringir el nivel de dispersión para cada fila de $\boldsymbol{S}$. Para un vector $x$, el valor de dispersión va desde 1, cuando $x$ es perfectamente disperso (1 coeficiente activo), a 0 cuando todos los coeficientes están activos (con el mismo valor). Este valor de dispersión se define en (3.5) como:

$$Dispersión(x) = \frac{\sqrt{n} - \frac{\|x\|_1}{\|x\|_2}}{\sqrt{n} - 1} \tag{3.5}$$

Sin embargo, aunque estos métodos pueden procesar datos con un perfil más o menos puntiagudos, no tienen en cuenta la continuidad de la fuente. Por lo tanto, ninguno de los algoritmos NMF mencionados anteriormente puede utilizar toda la información previa sobre las fuentes. Para modelar mejor este tipo de datos se puede expresar la dispersión en un dominio diferente. De hecho, como se ha comentado en la introducción de este capítulo, la dispersión de una señal depende de la base o el diccionario en las que se representa. En pocas palabras, las bases que mejor capturan la estructura geométrica de una fuente producirán representaciones más dispersas de esta fuente. Los antecedentes de aplicación de la dispersión en dominios transformados se han utilizado con éxito para resolver una gama muy amplia de problemas inversos. Sin embargo, en NMF, se han propuesto pocos algoritmos para imponer la dispersión en una base o diccionario diferente debido a la dificultad de tratar con dos apriorismos distintos en dos dominios diferentes. Es por esto que se han desarrollado algoritmos híbridos que permiten tratar estos escenarios, como es el caso del algoritmo nGMCA y sus variantes.

## 3.3 Análisis Generalizado de Componentes Morfológicos no Negativos (nGMCA)

### 3.3.1 Una primera versión "naive"

En la última década, el uso de la dispersión en el campo de BSS ha sido ampliamente explorado. En [Bobin et alt., 2006] se propuso un algoritmo llamado análisis generalizado de componentes morfológicos que introdujo la dispersión y que ha demostrado ser eficaz para separar las señales dispersas de los datos con ruido añadido. La diversidad morfológica se toma como medio para caracterizar fuentes separables en función de sus estructuras geométricas o morfológicas, "las fuentes separables con diferentes morfologías no

comparten los mismos coeficientes significativos en una representación dispersa dada" [Bobin et alt., 2006]. Cuando la dispersión se mantiene en el dominio directo, esto significa que las entradas de cada fuente con las amplitudes más significativas deberían ser diferentes. Posteriormente [Bobin et alt., 2007] extiende este algoritmo para tratar con mezclas no negativas. El GMCA con una restricción no negativa adicional estima la matriz de mezcla y las fuentes minimizando el problema de optimización que aparece en (3.6).

$$\arg \min_{A \geq 0, S \geq 0} \frac{1}{2} \|Y - AS\|_2^2 + \lambda \|S\|_0 \tag{3.6}$$

La pseudo-norma $\ell_0$ cuenta coeficientes no nulos en $\boldsymbol{S}$ y, por lo tanto, limita su número, forzando así la dispersión de $\boldsymbol{S}$. GMCA estima alternativa e iterativamente la solución de mínimos cuadrados sin restricciones, y restringe la no negatividad mediante un paso de umbral adicional para las fuentes con el fin de mantener solo los coeficientes más significativos. Estas actualizaciones se muestran en las líneas 6 y 7 del Algoritmo 1:

---

Algoritmo 1: nGMCA[naive]

---

*Entrada*: $\boldsymbol{Y}, K$

1: *Inicializar* $\boldsymbol{A}$ y $\boldsymbol{S}$
2: *para* $k = 1$ *hasta* $K$ *hacer*
3: *Normalizar columnas de* $\boldsymbol{A_{k-1}}$
4: $\boldsymbol{S}_{total} = \left(\boldsymbol{A_{k-1}^T A_{k-1}}\right)^{-1} \boldsymbol{A_{k-1}^T Y}$
5: *Seleccionar umbral* $\lambda_k$ *considerando* $\boldsymbol{S}_{total}$
6: $\boldsymbol{S}_k \leftarrow [Hard_\lambda(\boldsymbol{S}_{total})]_+$
7: $\boldsymbol{A}_k \leftarrow \left[\boldsymbol{Y S_k^T} \left(\boldsymbol{S_k S_k^T}\right)^{-1}\right]_+$
8: *fin para*
9: *Devuelve* $\boldsymbol{A_k}, \boldsymbol{S_k}$

Siendo $Y$ el conjunto de mezclas objeto de separación y $K$ el número máximo de iteraciones que permitiremos al algoritmo. El operador de umbral $Hard_\lambda$ se define en la ecuación (3.7).

$$Hard_\lambda : x \rightarrow \begin{cases} 0 \; si \; x < \lambda \\ x \; en \; otro \; caso \end{cases} \tag{3.7}$$

En [Bobin et alt., 2007] se pone de manifiesto que una característica crucial de nGMCA es el uso de un umbral decreciente $\lambda$. Al principio, este parámetro se establece en un valor alto y luego disminuye a lo largo de las iteraciones hasta un valor final que depende del nivel de ruido. La motivación detrás del umbral decreciente es, por una parte, estimar la matriz de mezcla a partir de las entradas de las fuentes que tienen la amplitud más alta y, por lo tanto, es probable que pertenezcan a una sola fuente y, por otro lado, ayudar a

eliminar los coeficientes más pequeños que son más sensibles a la contaminación acústica. Los valores finales de este umbral son un compromiso entre la eliminación de una cantidad de ruido suficiente y una separación correcta, ya que un umbral demasiado pequeño no eliminará suficiente ruido y uno demasiado grande puede generar que se pierda alguna fuente.

Es importante resaltar que este algoritmo generalmente no converge a una pareja estable $(\boldsymbol{A}, \boldsymbol{S})$, de ahí la denominación "naive". Esto significa que la solución dada por este tipo de algoritmo puede no ser estable y no ser óptima, por lo que la solución puede no proporcionar las fuentes no negativas más dispersas. Es por esto por lo que se desarrolló una nueva versión de este algoritmo que permite obtener soluciones más robustas y estables.

### 3.3.2 nGMCA standard

En la misma línea que el nGMCA$^{naive}$ mostrado en la sección anterior, mediante este nuevo algoritmo nGMCA$^{standard}$ presentado en [Rapin et alt., 2013], $\boldsymbol{A}$ y $\boldsymbol{S}$ se actualizan alternativamente con la excepción de que, ahora, cada actualización se resuelve exactamente, lo que garantiza que la solución $(\boldsymbol{A}, \boldsymbol{S})$ es estable y tiene la estructura buscada. Los pasos principales de este nuevo algoritmo se muestran en el Algoritmo 2.

Algoritmo 2: nGMCA$^{standard}$

*Entrada*: $\boldsymbol{Y}, K$
1: *Inicializar* $\boldsymbol{A_0}$, $\boldsymbol{S_0}$ y $\lambda_1$
2: *para* $k = 1$ *hasta* $K$ *hacer*
3: *Normalizar columnas de* $\boldsymbol{A_{k-1}}$
4: $\boldsymbol{S}_k \leftarrow \arg\min_{S \geq 0} \frac{1}{2} \|Y - A_{k-1}S\|_2^2 + \lambda_k \|S\|_1$
5: $\boldsymbol{A}_k \leftarrow \arg\min_{A \geq 0} \frac{1}{2} \|Y - AS_k\|_2^2$
6: *Seleccionar umbral* $\lambda_{k+1} \leq \lambda_k$
7: *fin para*
8: *Devuelve* $\boldsymbol{A_k}$, $\boldsymbol{S_k}$

Al igual que en el Algoritmo 1, $Y$ representa el conjunto de mezclas objeto de separación y $K$ el número máximo de iteraciones que permitiremos al algoritmo. Siguiendo la estrategia general de umbralización utilizada en GMCA y sus extensiones, el umbral $\lambda$ disminuye en cada iteración. Sin embargo, la estrategia utilizada en esta versión nGMCA$^{standard}$ difiere de la utilizada en el enfoque nGMCA$^{naive}$. En el primero, el umbral se aplica a las fuentes según lo definido mediante la estimación de sus mínimos cuadrados. Por el contrario, el umbral en nGMCA$^{standard}$ se aplica en cada paso descendente del gradiente. La regla de actualización de $\boldsymbol{S}$ en las iteraciones secundarias de nGMCA$^{standard}$ se puede escribir de la forma que muestra la ecuación (3.8).

$$S_{k+1} \leftarrow \left[S_k + \frac{1}{L}(A^T(AS_k - Y) - \lambda 1_{rn})\right] \tag{3.8}$$

siendo $1_{rn}$ una matriz de dimensiones $r\ x\ n$ que contiene solo unos. Por lo tanto, el umbral iterativo opera en el gradiente y no directamente en los valores de origen como en nGMCA$^{naive}$. Además, la variación de $\lambda$ afecta a todos los coeficientes activos. Esta estrategia consiste en comenzar con un parámetro grande $\lambda_0 = \|A_0^T(A_0S_0 - Y)\|_\infty$ que obliga a que los coeficientes de $\boldsymbol{S}$ no aumenten en la primera iteración, y luego ir reduciendo linealmente el umbral para refinar la solución mientras se preserva la continuidad.

## 3.4 nGMCA en el dominio transformado.

En el contexto de BSS, se ha demostrado que la dispersión proporciona más diversidad o contraste entre las fuentes, lo que ayuda en gran medida a mejorar su separación. Imponer la dispersión en un dominio transformado hace posible la separación de fuentes con estructuras geométricas complejas. En este punto se muestran dos extensiones del algoritmo nGMCA para abordar los problemas de NMF con la dispersión impuesta en un dominio transformado.

### 3.4.1 Formulaciones de síntesis y análisis

En un dominio transformado, la dispersión puede imponerse de dos maneras diferentes, estas son las formulaciones de síntesis y análisis [Rapin et alt., 2014].

Siendo $\boldsymbol{W}$ una matriz de dimensiones $p\ x\ n$, minimizar una función $f$ mediante una regularización de síntesis se expresa en la ecuación (3.9) de la siguiente manera:

$$\arg\min_{x_w \in \mathbb{R}p} f(W^T x_w) + \lambda \|x_w\|_1 \tag{3.9}$$

En esta formulación, lo desconocido de la minimización no es directamente la fuente sino coeficientes dispersos en el espacio transformado $\hat{x}_w \in \mathbb{R}p$. Siendo $D = W^T$, el objetivo es reconstruir la fuente buscada $\hat{x}$ como una combinación lineal dispersa de columnas de $\boldsymbol{D}$, es decir usando la menor cantidad de columnas de $\boldsymbol{D}$ posible como muestra la ecuación (3.10).

$$\hat{x} = W^T \hat{x}_w = \sum_{j=1}^{p} \hat{x}_{w|j} D_{.,j} \tag{3.10}$$

Por consiguiente, $\boldsymbol{D}$ se llama diccionario y sus columnas se llaman átomos. Este es un modelo generativo, de ahí el nombre de "síntesis" (se construye la fuente buscada usando componentes del espacio de la fuente, los átomos).

Aun así, es importante resaltar que el hecho de que $\hat{x}_w$ sea disperso no significa que $W\hat{x}_w = WW^T\hat{x}_w$ sea necesariamente disperso. En la práctica, esto significa que la formulación de síntesis encuentra una solución que tiene una representación dispersa en el dominio transformado, pero no que la transformación de la solución sea dispersa.

Por otro lado, en la formulación del análisis, la minimización se lleva a cabo directamente en el dominio directo de la fuente para encontrar una solución que sea dispersa cuando se multiplica por $\boldsymbol{W}$. Esto se expresa en la ecuación (3.11) de la siguiente manera:

$$\arg \min_{x_w \in \mathbb{R}n} f(x) + \lambda \|W_x\|_1 \tag{3.11}$$

El término $\lambda\|W_x\|_1$ penaliza las correlaciones entre $x$ y los átomos de $D = W^T$. En otras palabras, mientras que en la formulación de síntesis la señal se expresa como la suma de un número limitado de átomos, el objetivo en la formulación de análisis es obtener una fuente que esté fuertemente correlacionada solo con unos pocos átomos de D y que esté débilmente correlacionada con los otros átomos.

Cuando $\boldsymbol{W}$ es ortonormal, las formulaciones de síntesis y análisis son estrictamente equivalentes. De hecho, el cambio de variables $x_w = W_x$ nos permite obtener una formulación a partir de la otra, ya que $\boldsymbol{W}$ es invertible en ese caso. Aun así, es preferible usar diccionarios redundantes, con $p > n$, ya que se ha mostrado que mejoran la recuperación de fuentes. La ventaja de los diccionarios redundantes proviene de la mayor cantidad de átomos disponibles para representar dispersamente las fuentes.

#### 3.4.1.1 nGMCA Síntesis

Para ampliar el algoritmo nGMCA$^{\text{standard}}$ para usar una regularización de síntesis dispersa, las líneas 4 y 5 del Algoritmo 2 deben actualizarse de la siguiente manera:

Algoritmo 3: nGMCA$^{\text{Síntesis}}$

4: $\boldsymbol{S}_w^{(k)} \leftarrow \arg\min_{S \geq 0} \frac{1}{2}\|Y - A_{k-1}S_wW\|_2^2 + \lambda_k\|S_w\|_1$

5: $\boldsymbol{A}^{(k)} \leftarrow \arg\min_{A \geq 0} \frac{1}{2}\left\|Y - AS_w^{(k)}W\right\|_2^2$

Donde $\boldsymbol{S}_w^{(k)}$ y $\boldsymbol{A}^{(k)}$ representan respectivamente las reconstrucciones provisionales de las matrices de fuentes y mezclas en la k-esima iteración del algoritmo.

#### 3.4.1.2 nGMCA Análisis

Para ampliar el algoritmo nGMCA$^{\text{standard}}$ para usar una regularización de análisis dispersa, la línea 4 del Algoritmo 2 debe actualizarse de la siguiente manera:

---

Algoritmo 4: nGMCA$^{\text{Análisis}}$

---

4: $\boldsymbol{S}^{(k)} \leftarrow \arg\min_{S\geq 0} \frac{1}{2}\|Y - A_{k-1}S\|_2^2 + \lambda_k \|SW^T\|_1$

### 3.4.2 Formulación Convolutiva

En esta formulación, las fuentes se modelan como trenes de potenciales dispersos mediante un kernel laplaciano. Por lo tanto, las fuentes se pueden descomponer como una combinación lineal no negativa de átomos no negativos. En este caso, el problema de la actualización de $\boldsymbol{S}$ puede simplificarse enormemente ya que la no negatividad y la dispersión se pueden expresar en el mismo dominio:

---

Algoritmo 5: nGMCA$^{\text{Convolutiva}}$

---

4: $\boldsymbol{S}^{(k)} \leftarrow \arg\min_{S\geq 0} \frac{1}{2}\|Y - A_{k-1}S_w W\|_2^2 + \lambda_k \|S_w\|_1 + i_{\cdot\geq 0}(S_w)$

Esta versión de nGMCA se denomina nGMCA$^{\text{Convolutiva}}$. El kernel de convolución de nGMCA$^{\text{Convolutiva}}$ se establece como un kernel laplaciano de ancho 4. Este enfoque debería conducir a los mejores rendimientos de separación cuando las fuentes están compuestas de picos no negativos.

# Capítulo 4

# Evaluación de Algoritmos

## 4.1 Introducción

En el capítulo anterior se han expuesto diferentes algoritmos y variaciones que hacen uso de la dispersión y el NMF para resolver problemas BSS. Todos ellos han sido diseñados con la finalidad de abarcar un amplio grupo de escenarios y tipos de datos, desde diferentes dimensiones (audio, imágenes, etc.) hasta diferentes campos de aplicación. Es por esto, que la tarea de seleccionar uno de ellos para aplicar en el escenario de las mezclas multicomponente no es, a priori, una tarea trivial.

El objetivo de este capítulo es exponer estos algoritmos a un conjunto de pruebas para poder evaluar, de un modo objetivo, cuál de ellos se adapta mejor al escenario de este trabajo. Estas pruebas se realizarán desde diferentes puntos de vista para obtener una visión general del desempeño de los algoritmos en la separación de mezclas.

## 4.2 Algoritmos.

Los algoritmos que evaluaremos serán los siguientes:

- Algoritmo NMF (Método de mínimos cuadrados no negativos) [Kim and Park, 2008]. Algoritmo que hace uso de la factorización de matrices no negativas. Se incluye en la selección para contrastar los posibles beneficios de incluir la dispersión en las restricciones de la separación, ya que el NMF solo incluye restricciones de no negatividad.
- Algoritmo nGMCA$^{\text{Standard}}$ [Rapin et alt., 2013]. Algoritmo original que aúna la no negatividad y la dispersión como restricciones de la separación.
- Algoritmo nGMCA$^{\text{Síntesis}}$ [Rapin et alt., 2014]. Algoritmo que recoge el enfoque sintético de la separación de fuentes en el espacio de transformación. Principalmente desarrollado para separar componentes de imágenes con características diferentes como texturas. Se incluye para contraste.
- Algoritmo nGMCA$^{\text{Análisis}}$ [Rapin et alt., 2014]. Algoritmo que recoge el enfoque analítico de la separación de fuentes en el espacio de transformación. Al igual que el anterior, se incluye para contraste.
- Algoritmo nGMCA$^{\text{Convolutivo}}$ [Rapin et alt., 2014]. Algoritmo que explota los beneficios de representar las restricciones de separación en un espacio de transformación que pueda aumentar la dispersión de las mezclas.

## 4.3 Conjuntos de datos

Para desarrollar diferentes pruebas que abarquen múltiples puntos de vista, se han seleccionados varios conjuntos de datos con diferentes características. Concretamente se han seleccionado 3 conjuntos de datos que van aumentando la complejidad morfológica de los componentes, lo que nos permitirá comprobar la calidad de la separación de cada uno de los algoritmos en diferentes escenarios.

Los conjuntos están compuestos por 4 matrices:

- **Matriz S**. Contiene los componentes originales de las mezclas. Se diferencian 3 conjuntos de componentes distintos que escalan en complejidad morfológica. Estos se detallan en puntos subsiguientes.
- **Matriz A**. Contiene la matriz de mezcla de los componentes. Se genera aleatoriamente en cada una de las iteraciones de las pruebas para generar diferentes mezclas de los mismos componentes. Cada fila de esta matriz representa una mezcla diferente, y cada columna la concentración de un componente particular en la mezcla. Para la generación de esta matriz se establece como restricción que los valores de cada una de sus filas sumen 1.
- **Matriz N**. Contiene ruido blanco generado con distribución gaussiana, que se añadirla a las mezclas para evaluar la calidad de la separación con ruido agregado.
- **Matriz Y**. Esta matriz contiene las mezclas generadas a partir de las matrices anteriores y la ecuación (1.3).

Estas matrices nos permitirán proponer diferentes escenarios en los que evaluar los algoritmos presentados anteriormente.

### 4.3.1 Conjunto I. Conjunto de componentes sintéticos.

No se trata de un conjunto de componentes propiamente dicho ya que se genera de manera aleatoria en cada prueba. Se trata de componentes sintéticos simples, que permiten modificar sus características de dispersión mediante parametrización.

Estos componentes sintéticos se crean generando un conjunto aleatorio de valores de muestra con distribución normal dentro del rango [0,1]. Este conjunto de valores se filtra mediante una máscara lógica que consigue que un porcentaje seleccionado de ellos sea 0, generando de este modo un perfil parecido al que podría tener el espectro de un componente. El enmascaramiento se consigue multiplicando el conjunto original de muestras por otro conjunto similar generado aleatoriamente, pero con valores de 0 o 1 (máscara lógica). Esta máscara lógica se genera atendiendo a un parámetro llamado "coeficiente de activación" cuyos valores son del rango [0,1], el valor de este parámetro indicará el porcentaje de valores 1 que tendrá el conjunto. De este modo se consigue

controlar el número de muestras con valor 0 que tiene el componente sintético y, por lo tanto, modificar su dispersión, cuanto mayor sea el valor de activación menor será la dispersión del componente. Una muestra de ello se puede observar en la figura 4.1.

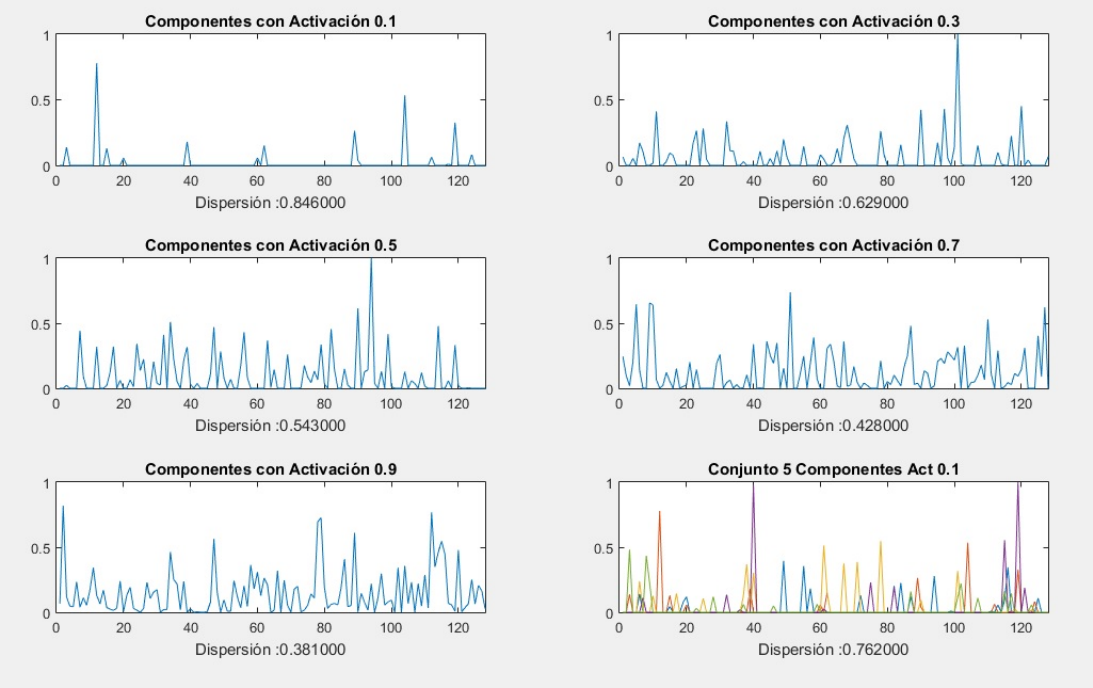
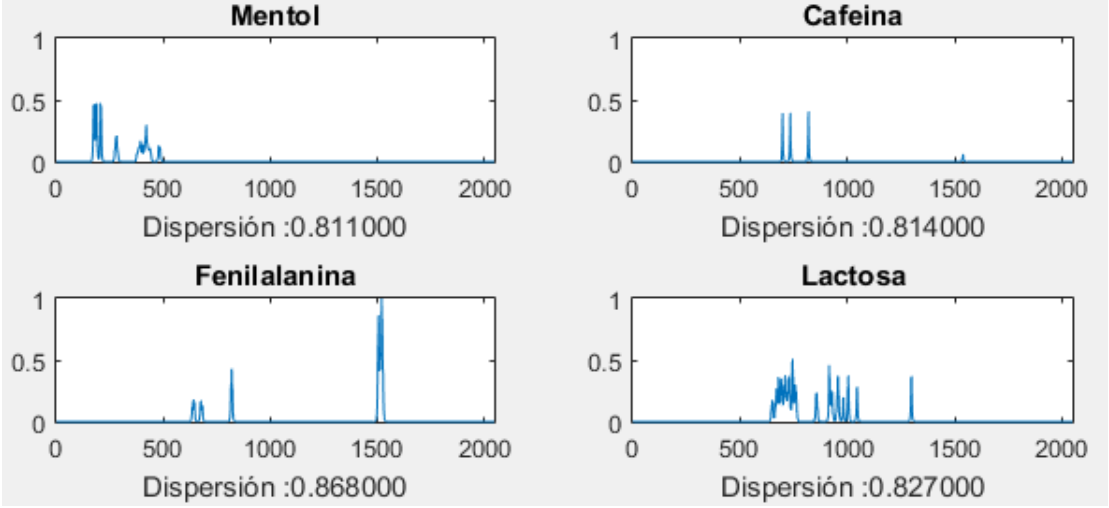



*Figura 4.1 Ejemplo de diferentes componentes sintéticos*

### 4.3.2 Conjunto II. Conjunto de componentes realista.

Se trata de un conjunto de espectros componentes generados artificialmente basándose en los datos recogidos en la SDBS (Spectral Database for Organic Compounds) del AIST (National Institute of Advanced Industrial Science and Technology) de Japón [Web]. Se compone de 15 espectros de diferentes compuestos químicos que aumentan la complejidad morfológica del conjunto anterior. La figura 4.2 ilustra una muestra de estos componentes.



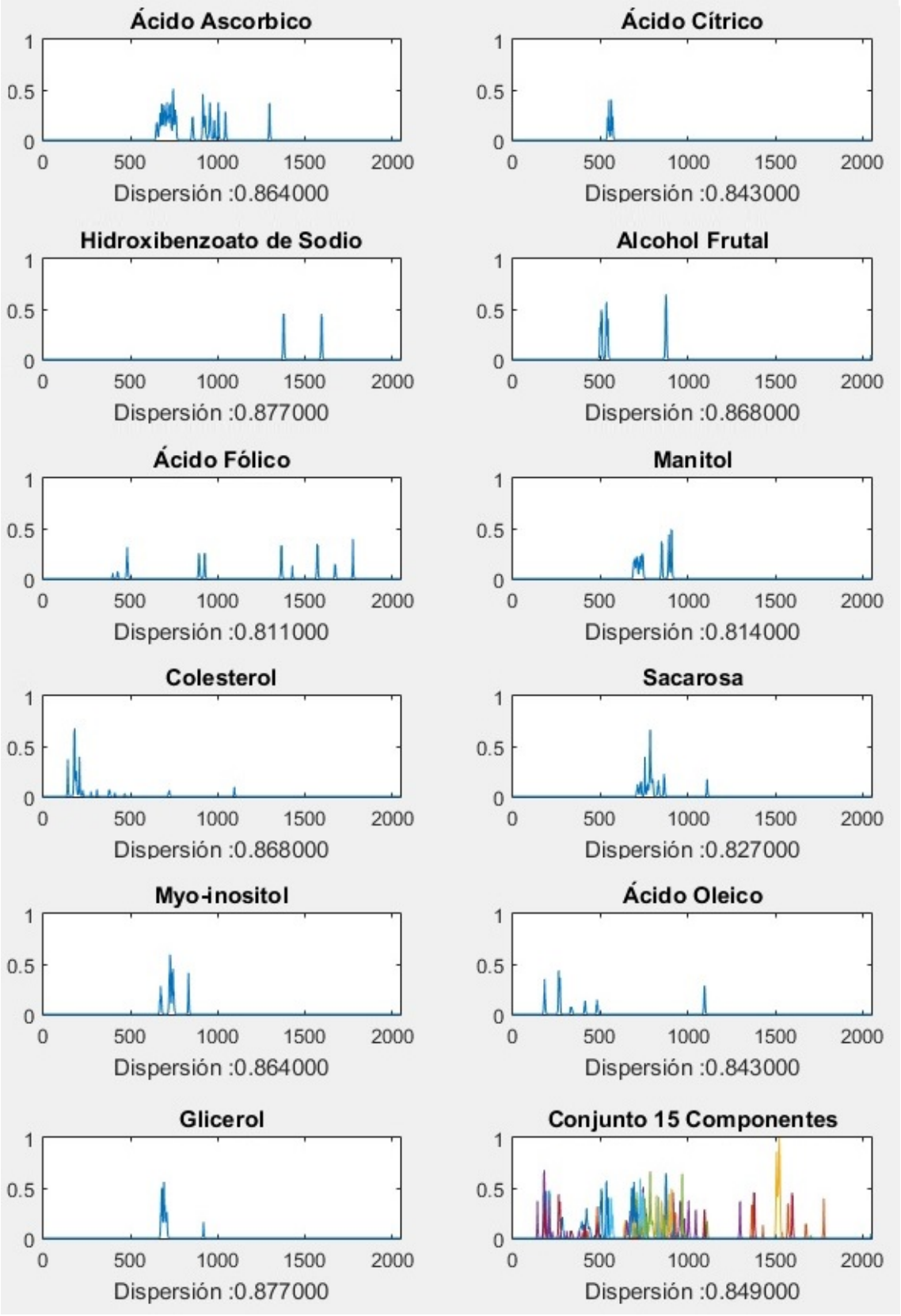



*Figura 4.2 Conjunto de 15 componentes realistas.*

### 4.3.3 Conjunto III. Conjunto de componentes reales.

Se trata de un conjunto de espectros IR de componentes reales y sus mezclas recogidos en laboratorio mediante espectrometría IR. Corresponden a 5 disolventes (1-Butanol, Cloroformo, Metanol, 2-Propanol y Tolueno) que forman parte de un caso real que se analizará en el capítulo 5 de este trabajo. Estos componentes tienen una complejidad morfológica muy elevada, lo que supone un reto para la separación de mezclas generadas a partir de ellos. Sus espectros se recogen en la figura 2.10.

La dispersión de cada una de las fuentes que aparece en las figuras anteriores está calculada mediante el índice de Hoyer [Hoyer, 2004], referenciado en la ecuación (3.5). Este índice indica la dispersión de una fuente con valores acotados [0,1] siendo 1 el valor de mayor y 0 el de menor dispersión.

## 4.4 Evaluación de los resultados.

Para comparar el desempeño de la separación de componentes que realizan los diferentes algoritmos en cada escenario, es necesario un criterio de calidad que sea invariante en su escala. Este criterio debe estar bien adaptado para medir el rendimiento de la reconstrucción.

En [Vincent et al., 2006] se proponen diferentes criterios para evaluar el desempeño de las técnicas de separación ciega de fuentes. En escenarios con presencia de ruido, proponen separar cada fuente estimada en la suma de varios componentes, como muestras (4.1).

$$s^{est} = s_{obj} + s_{int} + s_{noi} + s_{art} \tag{4.1}$$

Siendo $s_{obj}$ la proyección de $s^{est}$ sobre la fuente de referencia, y $s_{int} + s_{noi} + s_{art}$ representando respectivamente las interferencias con otras fuentes, la contaminación por ruido y contaminación por artefactos de algoritmos. Creando un criterio de relación de energía de tipo SNR llamado Source Distortion Ratio (SDR), que se ilustra en la ecuación (4.2).

$$SDR(s^{est}) = 10\log_{10}\left(\frac{\|s_{obj}\|_2^2}{\|s_{int} + s_{noi} + s_{art}\|_2^2}\right) \tag{4.2}$$

Como se indica en [Vincent et al., 2006], este criterio es una medida de rendimiento global que tiene en cuenta todos los elementos de la reconstrucción, es decir, una separación correcta, eliminación de ruido eficiente y pequeños artefactos que deja el algoritmo.

Además, este criterio tiene la ventaja de ser invariante en escala. Por lo tanto, el SDR será el criterio empleado en este trabajo para medir la calidad de separación de los componentes de una mezcla. La figura 4.3 ilustra diferentes calidades de separación según su SDR, en cada uno de sus apartados podemos ver en color azul el espectro del componente objetivo de la separación, y en color naranja punteado el componente resultado de la separación. Se puede observar que un mayor valor del SDR corresponde a un mejor ajuste del espectro resultado con el objetivo y, por tanto, una identificación del componente y su concentración en las mezclas más precisa.

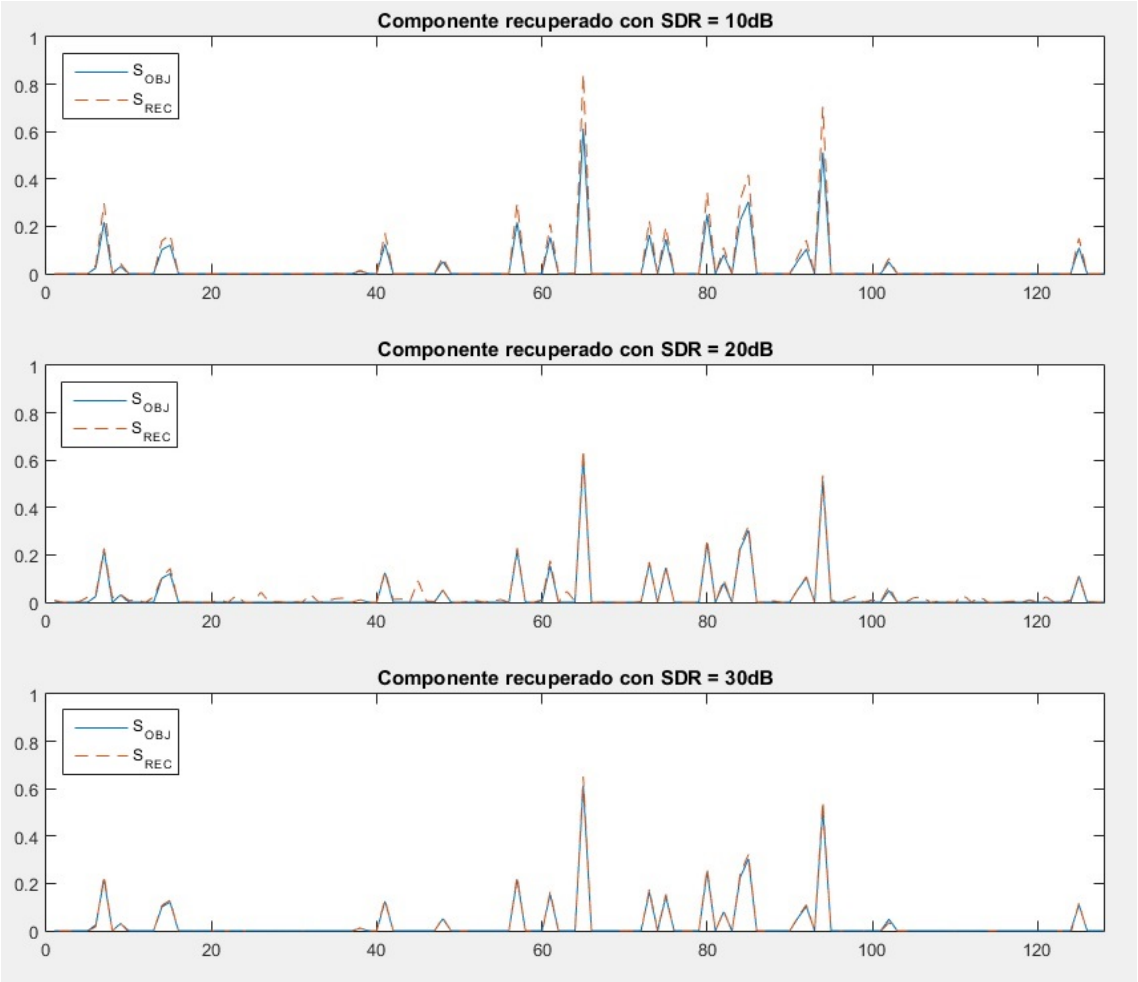



*Figura 4.3 Ejemplo de Componente recuperado con diferentes SDR*

Como norma general se puede afirmar que un componente recuperado con un SDR superior a 15dB posee un espectro bastante similar al original, y podemos considerar que una recuperación con valores por encima de 30dB de SDR es una recuperación óptima del componente que conducirá a una estimación precisa de su concentración en la matriz de mezcla.

## 4.5 Separación de componentes sobre el Conjunto_I.

### 4.5.1 Evaluación respecto al número de componentes.

En esta primera prueba vamos a evaluar la capacidad de separación de los diferentes algoritmos con respecto al número de componentes que componen las mezclas generadas. Para ello se crearán, en un escenario carente de ruido, unos conjuntos de prueba compuestos por 50 mezclas, generadas mediante matriz de mezcla aleatoria y un número de componentes variable entre 4 y 32, de 128 muestras. Se emplea un coeficiente de activación de 0.3.

Se ejecuta cada algoritmo por separado utilizando el mismo conjunto de datos, y se repite el proceso en 30 iteraciones generando un conjunto de pruebas distinto en cada una de ellas. Los resultados de esta prueba pueden reproducirse ejecutando el script de Matlab '*Test1_I*' que se adjunta a esta memoria y son los que se muestran en la figura 4.4.

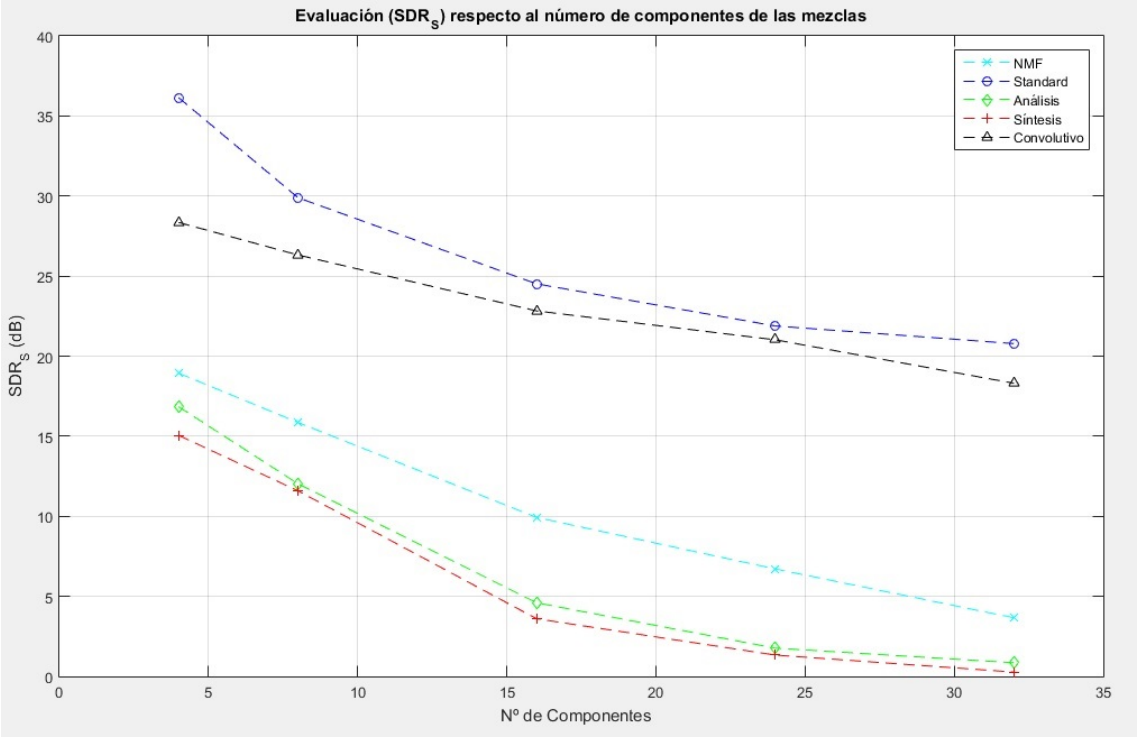



*Figura 4.4 Resultados de la evaluación respecto al número de componentes de las mezclas 'Test1_I'*

Se puede observar el efecto adverso que ejerce el aumento del número de componentes a separar en todos los algoritmos, siendo las variantes nGMCA$^{\text{Standard}}$ y nGMCA$^{\text{Convolutivo}}$ las que mejor calidad de separación generan en este caso.

### 4.5.2 Evaluación respecto al nivel de ruido.

En esta prueba se evalúa la capacidad de separación de los diferentes algoritmos con respecto al nivel de ruido que presentan las diferentes mezclas. De este modo, también se evalúa la capacidad de cada algoritmo para filtrar el ruido de las fuentes. Para ello se emplearán unos conjuntos de prueba compuestos por 50 mezclas generadas mediante matriz de mezcla aleatoria y 10 componentes de 128 muestras. Se emplea un coeficiente de activación de 0.3.

Para cada conjunto de mezclas **Y** se genera una matriz de ruido blanco con distribución gaussiana **N** que se añade a la mezcla siguiendo la ecuación (1.3). Se ejecuta cada algoritmo por separado utilizando el mismo conjunto de datos, y se repite el proceso en 30 iteraciones generando un conjunto de pruebas distinto en cada una de ellas, variando en cada ocasión el nivel de ruido entre -100dB y -10dB. Los resultados de esta prueba pueden reproducirse ejecutando el script de Matlab '*Test2_I*' que se adjunta a esta memoria y son los que se muestran en la figura 4.5.

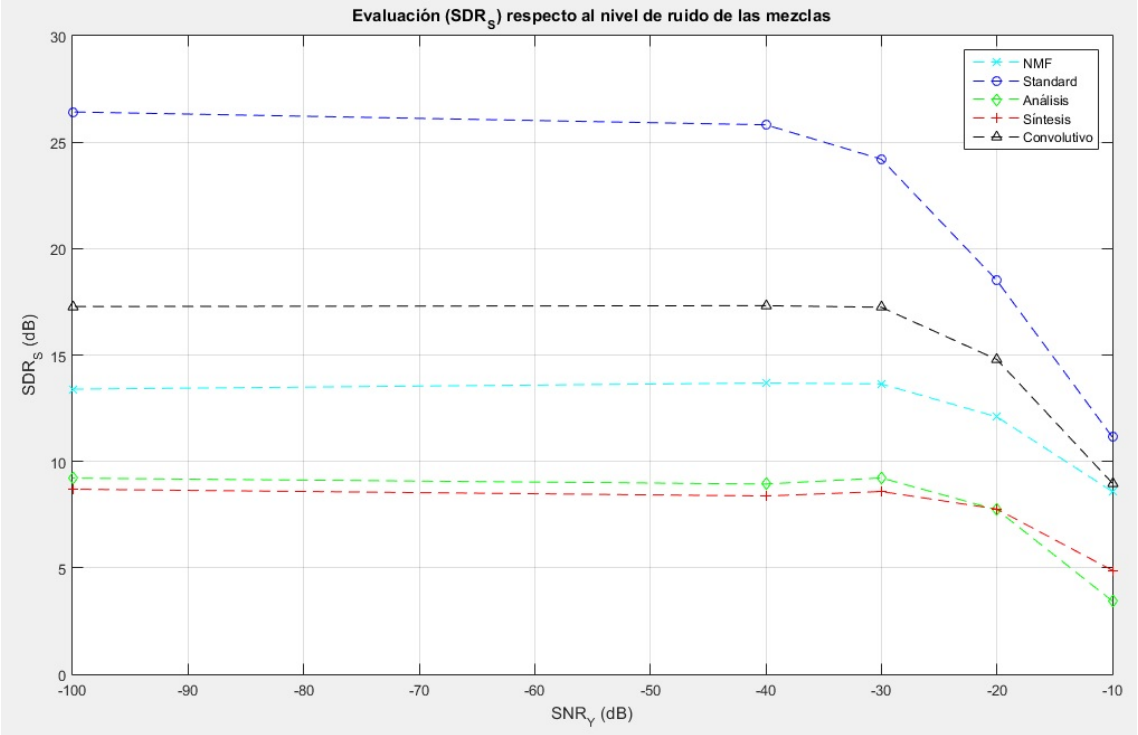



*Figura 4.5 Resultados de la evaluación respecto al nivel de ruido de las mezclas 'Test2_I'*

Se observa que los algoritmos admiten aceptablemente unos niveles de ruido entorno a los -20dB, tras lo cual la calidad de la separación empieza a verse afectada. Se observa también que, con mayores niveles de ruido, el desempeño de los algoritmos se iguala notablemente.

Para ilustrar la diferencia de calidad en las mezclas dependiendo del nivel de ruido de cada una de ellas, se muestra en la figura 4.6 un conjunto de gráficos que representan el espectro

una mezcla al azar compuesta por 5 componentes sintéticos de 512 muestras cada uno, generados con un coeficiente de activación de 0.1. Concretamente se compara el espectro original de la mezcla, representado en naranja punteado, con el espectro de ella misma con diferentes niveles de ruido añadido, desde -40 dB hasta -10 dB.

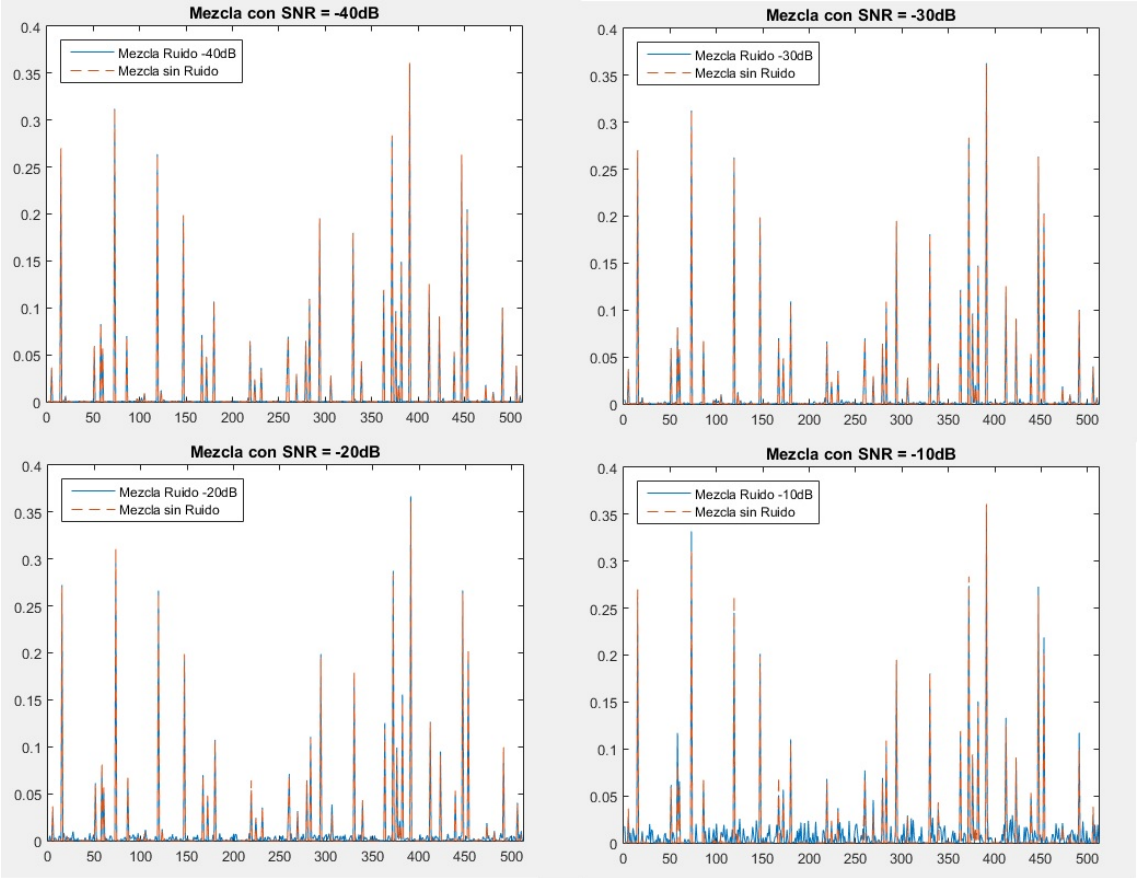



*Figura 4.6 Comparación de espectros con diferentes niveles de ruido añadido.*

### 4.5.3 Evaluación respecto al número de mezclas del conjunto.

Sin duda la cantidad de información que poseen los algoritmos para la separación debe afectar a su desempeño. En esta prueba se evalúa la capacidad de separación de los diferentes algoritmos con respecto al número de mezclas que contiene el conjunto de datos aportado. Para ello se emplearán, en un escenario sin ruido añadido, unos conjuntos de prueba compuestos por un número variable de mezclas (entre 10 y 200) generadas mediante matriz de mezcla aleatoria y 5 componentes de 128 muestras. Se emplea un coeficiente de activación de 0.3.

Los algoritmos utilizados en este trabajo no están preparados para soportar problemas de separación subdeterminados (con menos mezclas que componentes), sin embargo, es interesante observar cómo se comportan ante niveles de sobredeterminación elevados. Para esta prueba se ejecuta cada algoritmo por separado utilizando el mismo conjunto de datos,

y se repite el proceso en 30 iteraciones generando un conjunto de pruebas distinto en cada una de ellas. Los resultados de esta prueba pueden reproducirse ejecutando el script de Matlab '*Test3_I*' que se adjunta a esta memoria y son los que se muestran en la figura 4.7.

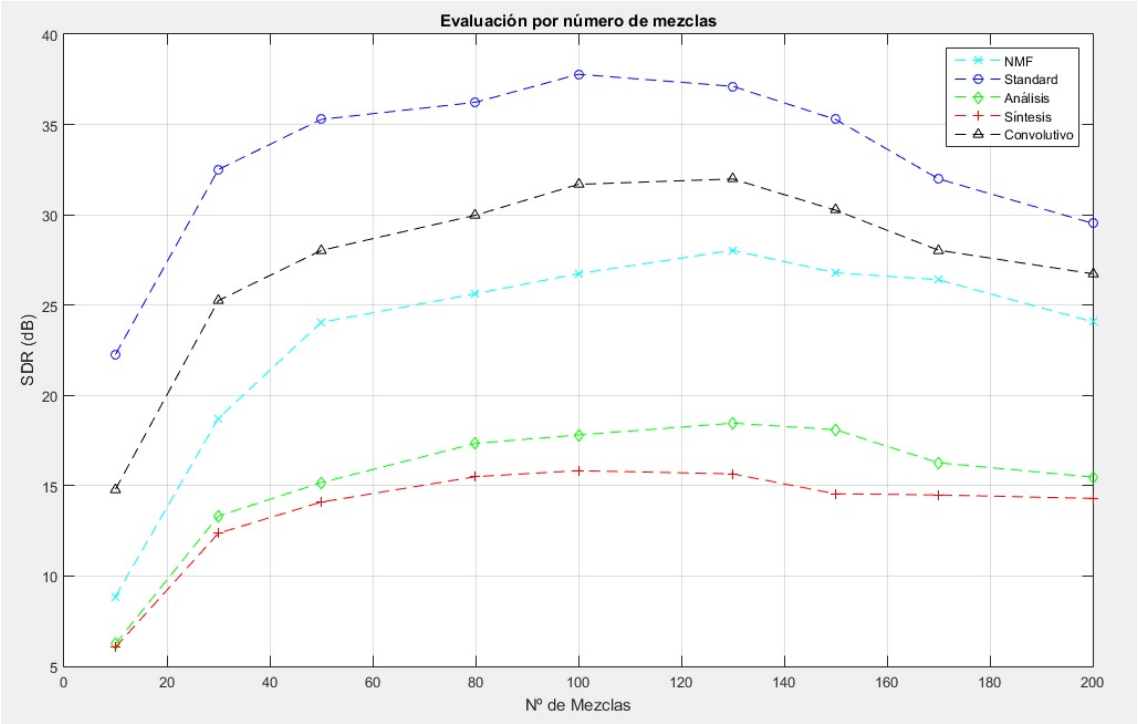



*Figura 4.7 Resultados de la evaluación respecto al número de mezclas 'Test3_I'*

Al contrario de lo que cabría esperar, la figura 4.7 muestra un límite en la mejora de la separación, tras el cual la mayoría de los algoritmos, no solo dejan de mejorar su rendimiento, sino que empeoran la calidad de recuperación de componentes llegada cierta cantidad de mezclas. Como veremos a continuación, en el comportamiento observado en la figura 4.7 influyen fundamentalmente dos factores.

Por una parte, existe un factor que condiciona el punto en el que la calidad de la separación deja de mejorar, esto viene determinado por la proporción Componentes/Mezcla, se puede decir que existe una relación ideal Componentes/Mezclas en la que se observan mejores resultados de separación. La figura 4.8 muestra la separación, mediante algoritmo $nGMCA^{Standard}$, de varios conjuntos de mezclas compuestos por un número variable de componentes y con un índice de dispersión media de 0.85. En ella se puede observar como a medida que aumenta el número de componentes, el punto en el que la separación alcanza su máxima calidad se encuentra en un número mayor de mezclas.

El segundo factor determinante en el comportamiento observado en la figura 4.7 es la dispersión de las mezclas, esta influye en el grado de empeoramiento de la calidad de la separación una vez alcanzado el punto óptimo de proporción Componentes/Mezclas. La figura 4.9 ilustra el resultado de la separación de varios conjuntos de mezclas con diferentes

grados de dispersión y diversa cantidad de componentes. Se puede observar cómo una mayor dispersión de las mezclas conlleva un empeoramiento mayor, mientras que una dispersión baja mantiene la calidad de la separación en niveles máximos.

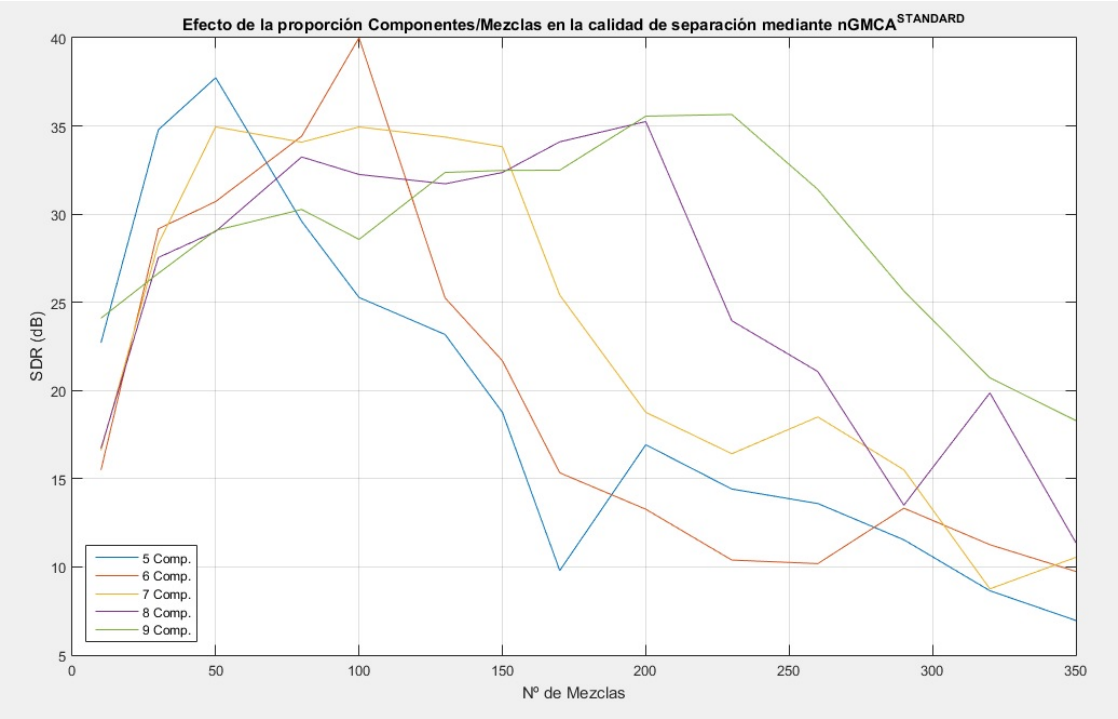



*Figura 4.8 Efecto de la proporción Componentes/Mezclas sobre Conjunto_I de datos*

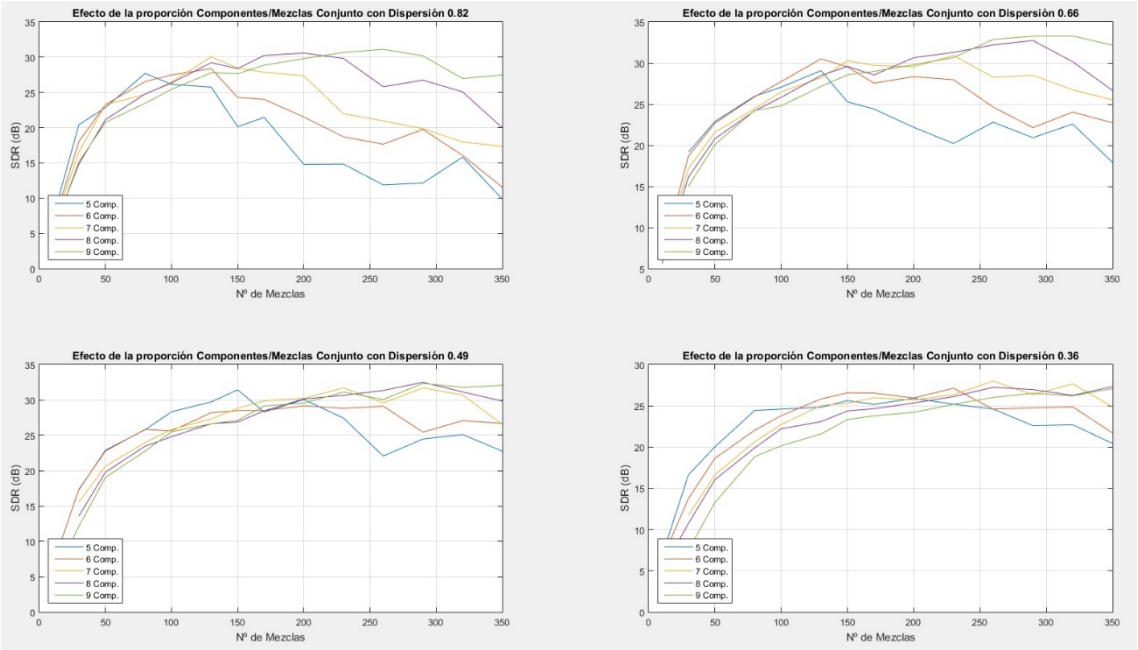



*Figura 4.9 Efecto de la dispersión en la separación tras alcanzar la proporción óptima Componentes/Mezclas*

La causa de este fenómeno se detalla más adelante en el punto 4.7.1 de esta memoria.

### 4.5.4 Evaluación respecto a la dimensionalidad de los componentes.

Al igual que en el caso anterior, la dimensionalidad o longitud de los componentes de las mezclas es un factor a tener en cuenta a la hora de ejecutar la separación. Un número mayor de muestras supone más información, y debería mejorar la separación. En esta prueba se evalúa la capacidad de separación de los diferentes algoritmos con respecto a la dimensión de los componentes. Para ello se emplearán, en un escenario sin ruido añadido, unos conjuntos de prueba compuestos por 50 mezclas generadas mediante matriz de mezcla aleatoria y 10 componentes de dimensionalidad variable. Se emplea un coeficiente de activación de 0.3.

Para esta prueba se ejecuta cada algoritmo por separado utilizando el mismo conjunto de datos, y se repite el proceso en 30 iteraciones generando un conjunto de pruebas distinto en cada una de ellas, variando en cada ocasión la longitud de los componentes de 16 a 512 muestras. Los resultados de esta prueba pueden reproducirse ejecutando el script de Matlab '*Test4_I*' que se adjunta a esta memoria y son los que se muestran en la figura 4.10.

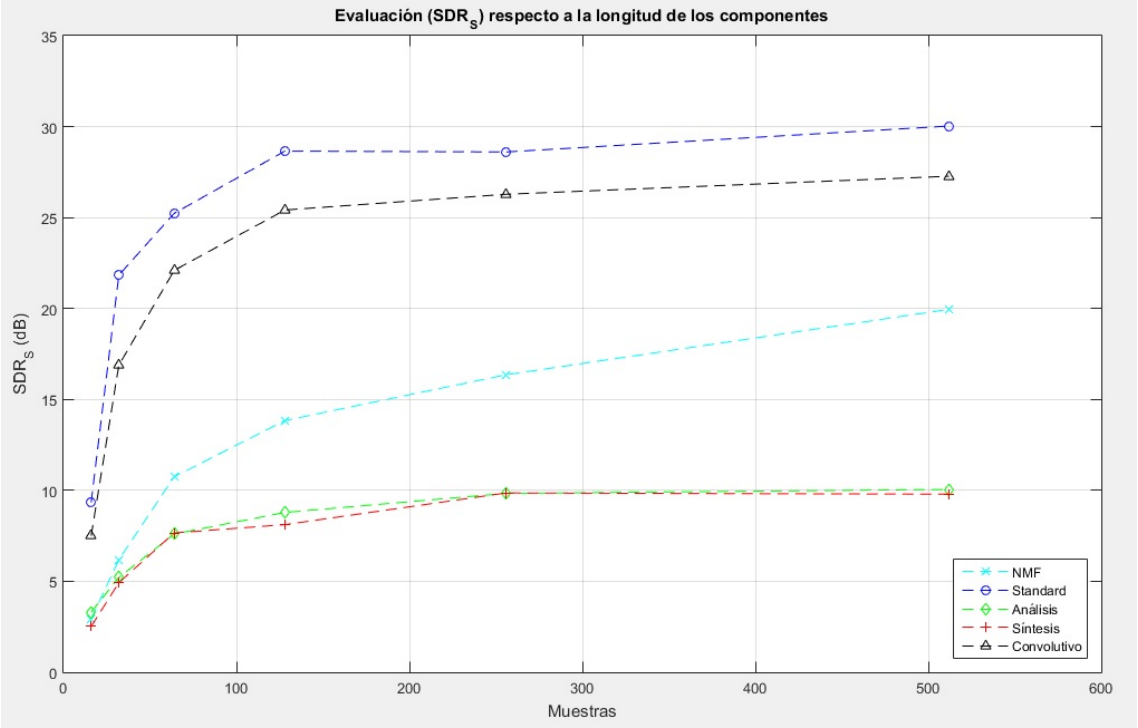



*Figura 4.10 Resultados de la evaluación respecto a la longitud de los componentes 'Test4_I'*

La figura anterior ilustra, al igual que en el caso de la sobredeterminación de mezclas, que una mayor información influye positivamente en la calidad de separación de los algoritmos. Aunque cabe destacar que, llegado cierto punto (128 muestras por componente en este caso), la mejoría en la calidad no es demasiado determinante. Como excepción, el algoritmo NMF sigue mejorando su desempeño cuanto mayor es el tamaño de los componentes de las mezclas.

### 4.5.5 Evaluación respecto a la dispersión de los componentes.

La última prueba sobre este conjunto de componentes pretende medir la relación que existe entre la calidad de separación de cada algoritmo con la dispersión media de los componentes que forman las mezclas. Para ello se emplearán, en un escenario sin ruido añadido, unos conjuntos de prueba compuestos por 50 mezclas generadas mediante matriz de mezcla aleatoria y 10 componentes de 128 muestras.

Para modificar la dispersión media se ha modificado el coeficiente de activación en la generación de los conjuntos de prueba, a mayor coeficiente de activación menor dispersión. Para esta prueba se ejecuta cada algoritmo por separado utilizando el mismo conjunto de datos, y se repite el proceso en 30 iteraciones generando un conjunto de pruebas distinto en cada una de ellas, variando en cada ocasión el coeficiente de activación de 0.1 a 1. Los resultados de esta prueba pueden reproducirse ejecutando el script de Matlab '*Test5_I*' que se adjunta a esta memoria y son los que se muestran en la figura 4.11.

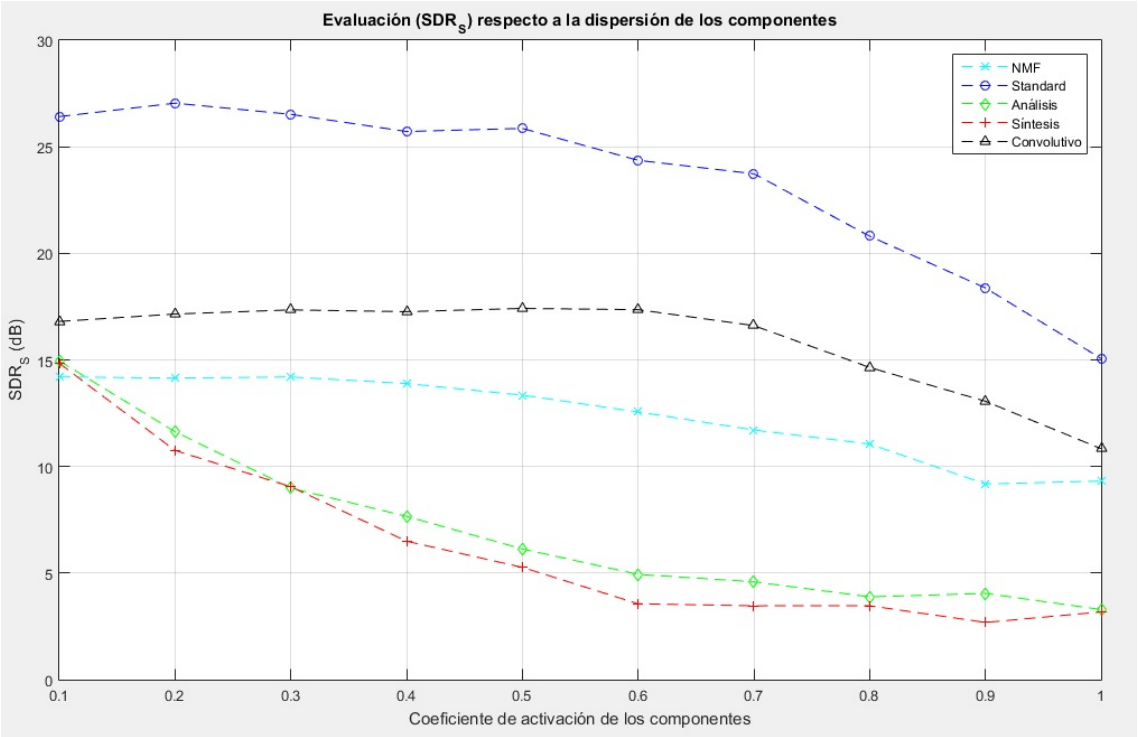



*Figura 4.11 Resultados de la evaluación respecto a la dispersión de los componentes 'Test5_I'*

Comprobamos como los algoritmos más afectados por la reducción de la dispersión de las fuentes son todos los de la familia nGMCA, que son aquellos que explotan esta característica. Se puede observar como en el caso de los enfoques nGMCA$^{\text{Analítico}}$ y nGMCA$^{\text{Sintético}}$ se ven afectados con anterioridad al caso de los algoritmos nGMCA$^{\text{Sstandard}}$ y nGMCA$^{\text{Convolutivo}}$ los cuales siguen obteniendo unos buenos niveles de separación con dispersiones más bajas.

### 4.5.6 Análisis temporal de pruebas sobre Conjunto_I de componentes.

Además de la calidad en la separación de los componentes, se ha querido evaluar, de una manera empírica, el coste temporal que ha tenido cada uno de los algoritmos en el desarrollo de cada prueba realizada a este grupo de componentes.

Los resultados que se muestran a continuación son el producto de cada uno de los test anteriores, los cuales, junto a los resultados ya expuestos, generan una tabla de tiempos con los valores medios de las 30 iteraciones completadas por cada algoritmo. Estos tiempos ha sido medidos durante la ejecución de los diferentes algoritmos mediante un sistema con procesador Intel Core i5-4460 de cuatro núcleos y frecuencia de reloj de 3.20Ghz. La figura 4.10 ilustra el coste temporal de cada una de las separaciones realizadas.

De estos resultados se extraen dos conclusiones. Por una parte, podemos concluir que las dos únicas variables que influyen claramente en los tiempos de ejecución de las separaciones son tanto el número de componentes que componen las mezclas, como el propio número de mezclas. En el primer caso se muestra un incremento exponencial y en el segundo uno lineal.

Por otro lado, también se observa en la figura 4.12, como las variaciones $nGMCA^{Analítico}$ y $nGMCA^{Sintético}$ son claramente ineficientes desde el punto de vista temporal, mientras que el resto de algoritmos se mantienen en unos tiempos de ejecución aceptables.

### 4.5.7 Conclusiones de las pruebas sobre Conjunto_I de componentes.

A partir de los resultados obtenidos de esta primera batería de pruebas sobre el Conjunto_I de componentes se pueden extraer varias conclusiones interesantes.

Por una parte, podemos determinar la clara ineficiencia de los algoritmos $nGMCA^{Analítico}$ y $nGMCA^{Sintético}$ sobre este conjunto de datos. Tanto desde el punto de vista temporal, como se vio en el punto anterior, como desde lo relativo a la calidad de la separación.

Otro aspecto a destacar es la estabilidad en el comportamiento del algoritmo NMF. Este algoritmo se ve menos afectado que los otros por el empeoramiento de las condiciones del conjunto de componentes, ya sea por el aumento de ruido en las mezclas o por el deterioro en la dispersión de los propios componentes. También cabe destacar la gran eficiencia temporal que presenta, siendo el algoritmo que realiza la separación de modo más rápido, con diferencia.

Por último, se observa como el algoritmo $nGMCA^{Standard}$ es el que mejor comportamiento tiene a la hora de realizar la separación sobre este conjunto de componentes. Si bien su eficiencia temporal es peor que en el caso del algoritmo NMF, su compromiso Tiempo/Calidad de separación es netamente superior al del resto.

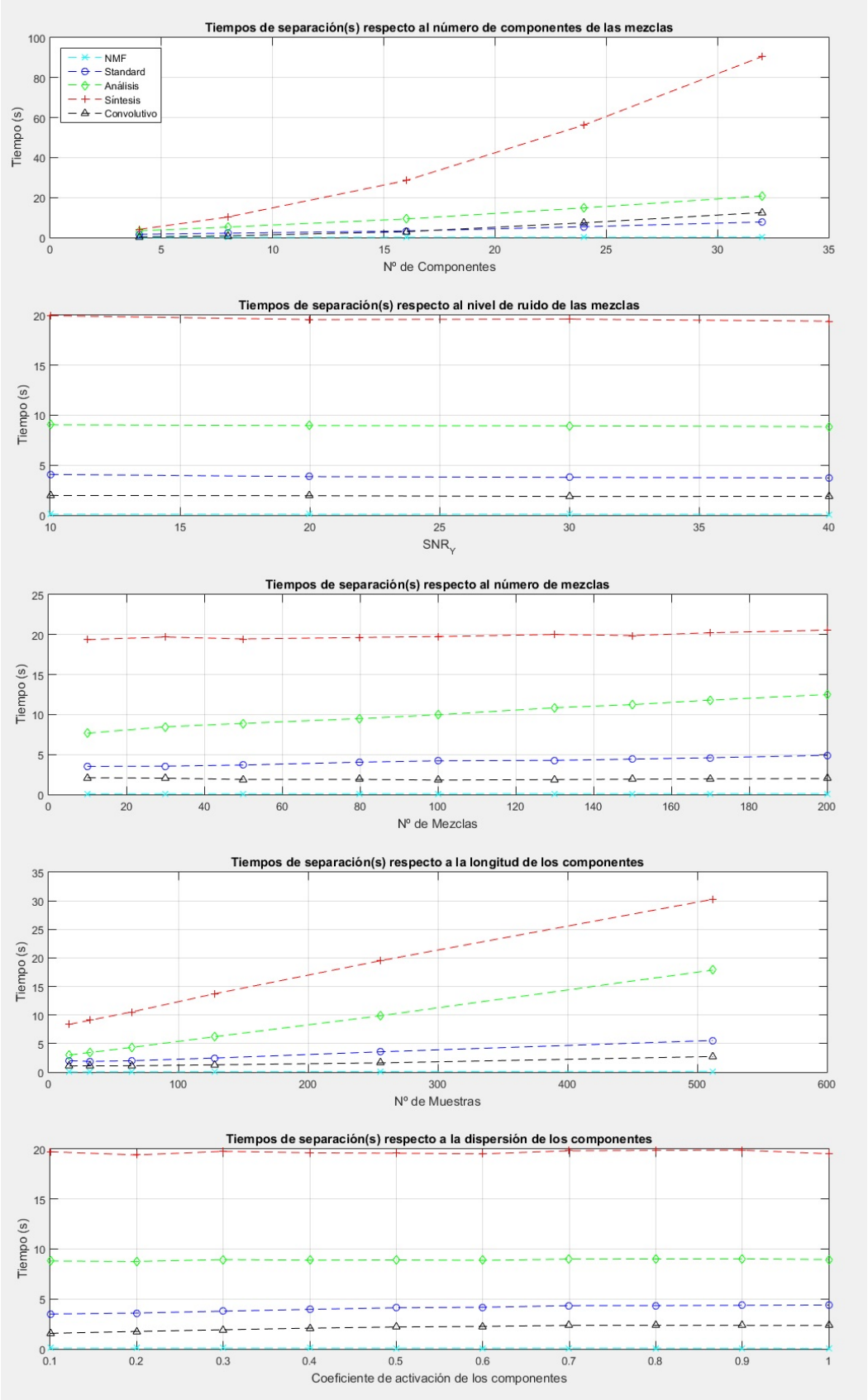



*Figura 4.12 Coste temporal de ejecución de los algoritmos en pruebas de evaluación sobre Conjunto_I. Estimación realizada mediante procesador Intel Core 5i-4460 (3.2Ghz).*

La tabla 4.1 recoge la comparativa de los resultados de los diferentes test de esta sección. Se muestran los valores medios de reconstrucción conseguidos por cada uno de los algoritmos en las diferentes pruebas realizadas sobre el Conjunto_I de componentes.

| Algoritmo / Test | *Componentes* | *$SNR_Y$* | *Mezclas* | *Muestras* | *Dispersión* |
|---|---|---|---|---|---|
| ***NMF*** | *11.01 dB* | *12.27 dB* | *17.20 dB* | *11.67 dB* | *12.36 dB* |
| ***nGMCA$^{Standard}$*** | ***26.64 dB*** | ***21.21 dB*** | ***28.36 dB*** | ***23.95 dB*** | ***23.39 dB*** |
| ***nGMCA$^{Analítico}$*** | *7.21 dB* | *7.70 dB* | *11.06 dB* | *7.46 dB* | *7.01 dB* |
| ***nGMCA$^{Sintético}$*** | *6.36 dB* | *7.66 dB* | *10.67 dB* | *7.14 dB* | *6.27 dB* |
| ***nGMCA$^{Convolutivo}$*** | *23.36 dB* | *15.12 dB* | *16.79 dB* | *20.90 dB* | *15.84 dB* |

*Tabla 4.1 Comparativa de resultados SDR medios (dB) de reconstrucción por algoritmo y test, Conjunto_I.*

La tabla 4.1 muestra de manera condensada lo apuntado anteriormente. El algoritmo nGMCA$^{Sstandard}$ es, con diferencia, el que mejor desempeño muestra en las separaciones realizadas sobre el Conjunto_I de datos.

## 4.6 Separación de componentes sobre el Conjunto_II.

Con la repetición de algunas de las pruebas de la batería anterior sobre otro conjunto de componentes se ha pretendido comprobar si existe diferencia en el comportamiento de los algoritmos al variar la diversidad morfológica de las mezclas. Este segundo conjunto se caracteriza por tener similares niveles de dispersión que el Conjunto_I, pero una diversidad morfológica de sus componentes mayor.

### 4.6.1 Evaluación respecto al número de componentes.

Se evalúa la capacidad de separación de los diferentes algoritmos con respecto al número de componentes que componen las mezclas generadas. Para ello se generarán, en un escenario carente de ruido, unos conjuntos de prueba compuestos por 50 mezclas generadas mediante matriz de mezcla aleatoria y un número de componentes variable entre 4 y 14 de 512 muestras.

Se ejecuta cada algoritmo por separado utilizando el mismo conjunto de datos, y se repite el proceso en 30 iteraciones generando un conjunto de pruebas distinto en cada una de ellas. Los resultados de esta prueba pueden reproducirse ejecutando el script de Matlab '*Test1_II*' que se adjunta a esta memoria y son los que se muestran en la figura 4.13. Esta figura muestra un deterioro de la calidad de separación con el aumento del número de fuentes a separar, como sucedía con el conjunto de componentes sintéticos.

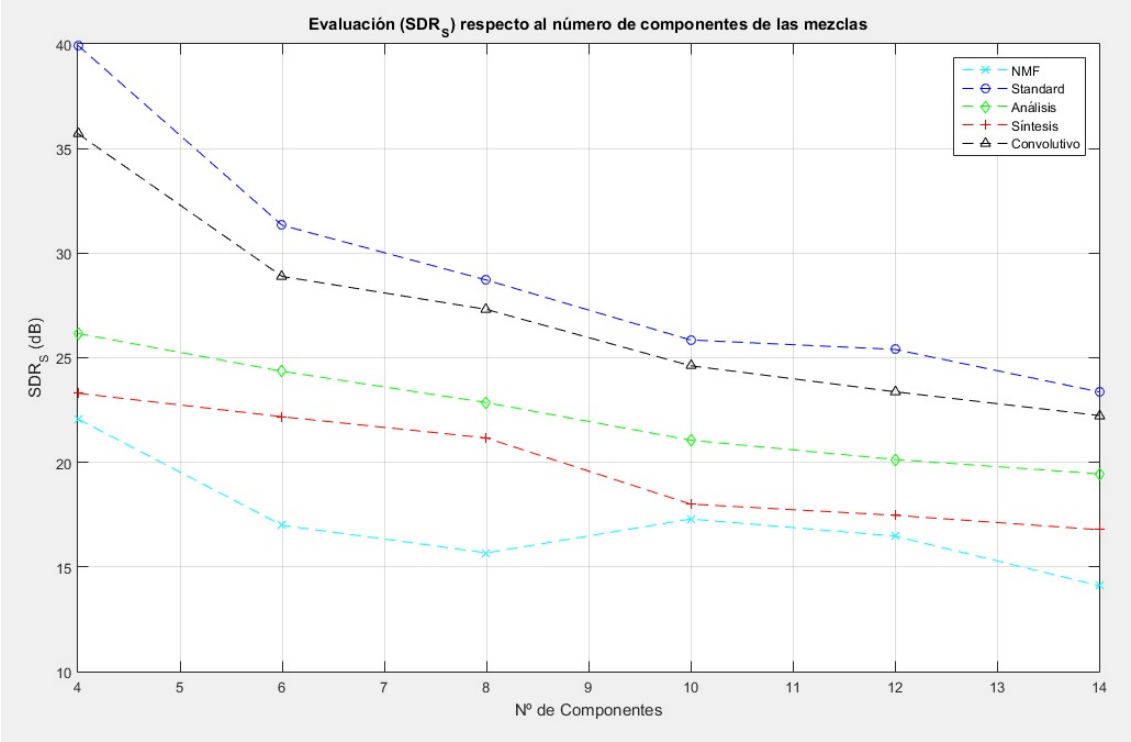



*Figura 4.13 Resultados de la evaluación respecto al número de componentes de las mezclas 'Test1_II'*

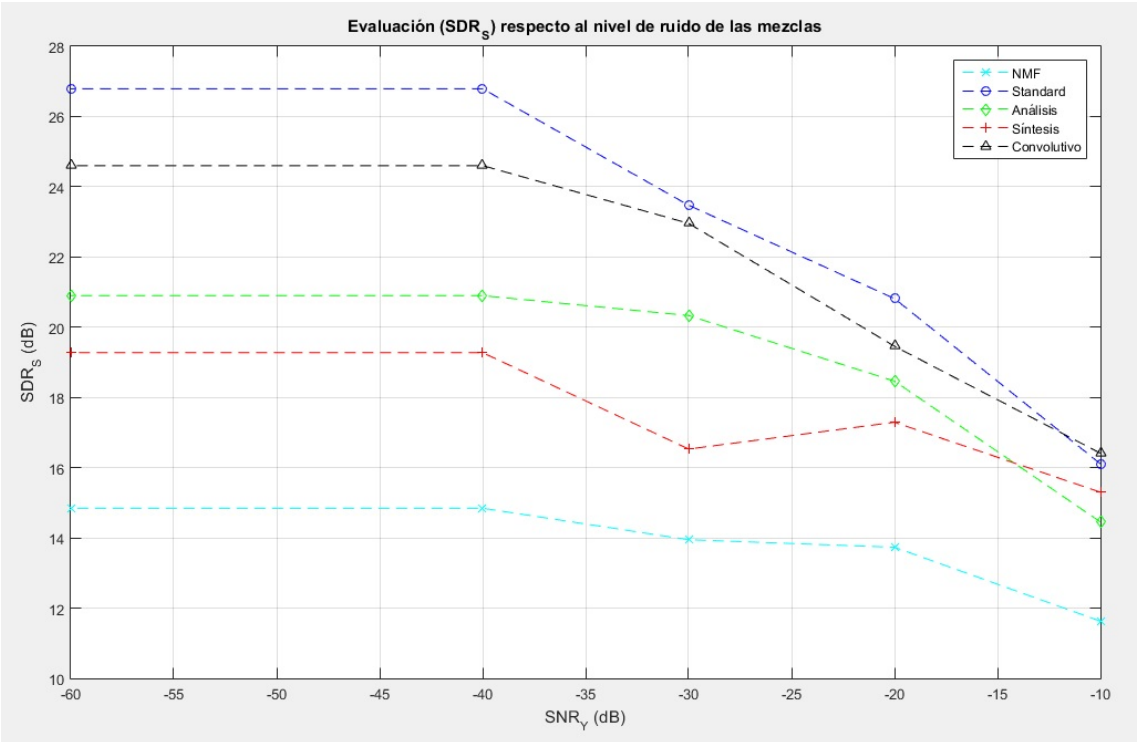



*Figura 4.14 Resultados de la evaluación respecto al nivel de ruido de las mezclas 'Test2_II'*

### 4.6.2 Evaluación respecto al nivel de ruido.

Se repite la prueba del punto 4.5.2. Para ello se emplearán unos conjuntos de prueba compuestos por 25 mezclas generadas mediante matriz de mezcla aleatoria y 5 componentes de 512 muestras del Conjunto_II.

Como en el caso anterior, para cada conjunto de mezclas **Y** se genera una matriz de ruido blanco con distribución gaussiana **N** que se añade a la mezcla siguiendo la ecuación (1.3). Se ejecuta cada algoritmo por separado utilizando el mismo conjunto de datos, y se repite el proceso en 30 iteraciones generando un conjunto de pruebas distinto en cada una de ellas, variando en cada ocasión el nivel de ruido entre -60dB y -10dB. Los resultados de esta prueba pueden reproducirse ejecutando el script de Matlab '*Test2_II*' que se adjunta a esta memoria y son los que se muestran en la figura 4.14.

Se observa cómo, en este caso, el deterioro en la calidad de la separación empieza con niveles de ruido inferiores, ya que con -30dB de ruido la calidad desciende notablemente en todos los algoritmos, salvo el caso del NMF que no se ve tan afectado por esta característica de las mezclas. El hecho de que el ruido afecte más a la separación de componentes en las mezclas generadas mediante el Conjunto_II que en las mezclas generadas mediante el Conjunto_I se debe a que el ruido añadido aumenta la complejidad morfológica de las mezclas, si este ruido se añade a mezclas ya de por sí más complejas, su efecto negativo será mayor.

### 4.6.3 Evaluación respecto al número de mezclas del conjunto.

En este caso se repiten las pruebas realizadas en el punto 4.5.3. Para ello se emplearán, en un escenario sin ruido añadido, unos conjuntos de prueba compuestos por un número variable de mezclas (entre 15 y 160) generadas mediante matriz de mezcla aleatoria y 5 componentes de 512 muestras.

Para esta prueba se ejecuta cada algoritmo por separado utilizando el mismo conjunto de datos, y se repite el proceso en 30 iteraciones generando un conjunto de pruebas distinto en cada una de ellas. Los resultados de esta prueba pueden reproducirse ejecutando el script de Matlab '*Test3_II*' que se adjunta a esta memoria y son los que se muestran en la figura 4.15.

Podemos comprobar en este caso un comportamiento similar al observado en el punto 4.5.3 en el que existe un punto de inflexión en el número de mezclas que los algoritmos pueden manejar con mejora sobre este conjunto de componentes. Podemos ver también como la degradación en la calidad de la separación se asemeja a la encontrada en la separación de mezclas de 5 componentes con un grado de dispersión similar.

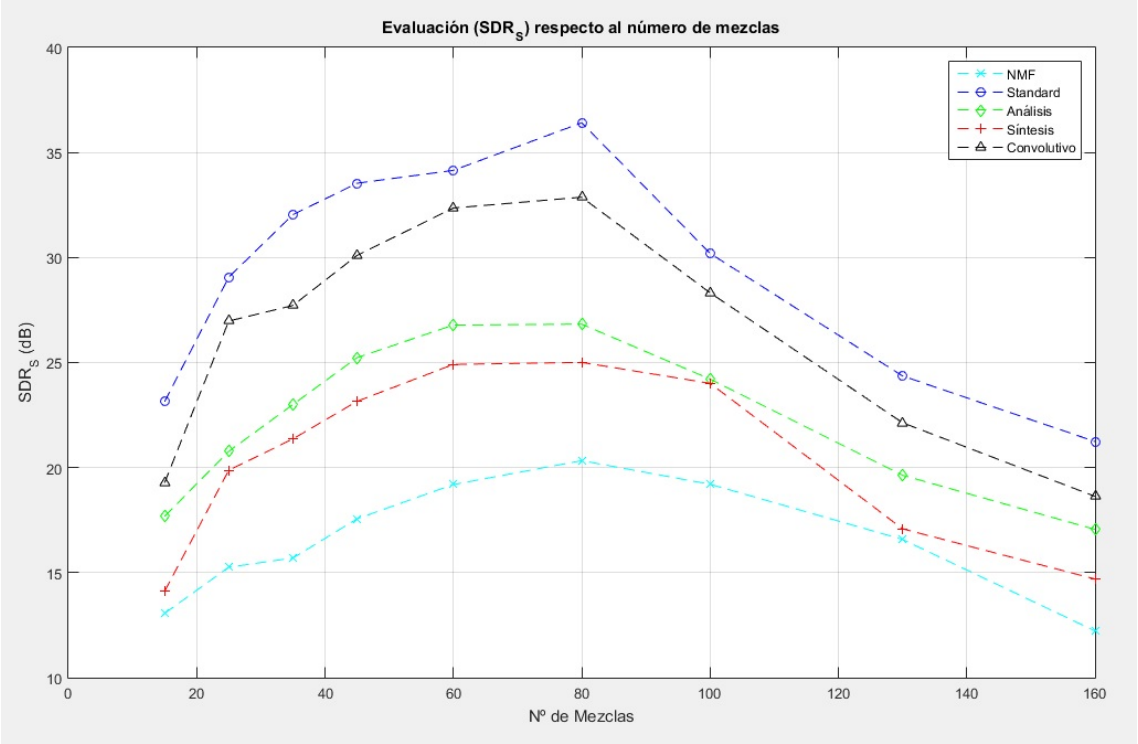



*Figura 4.15 Resultados de la evaluación respecto al número de mezclas 'Test3_II'*

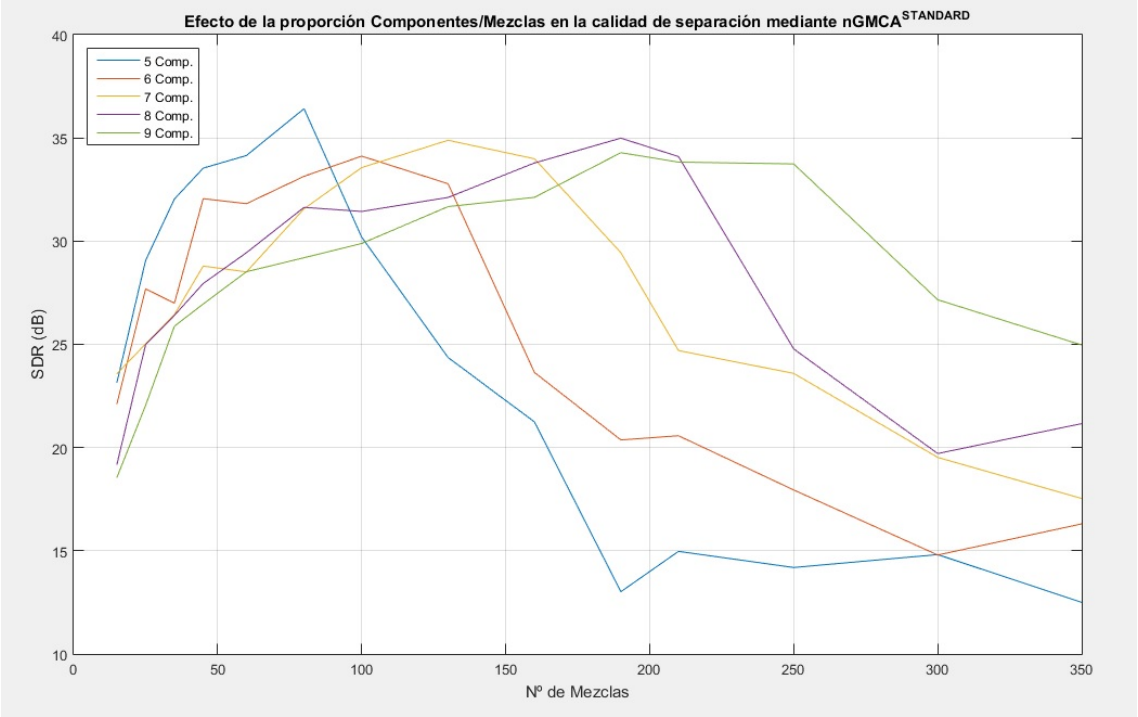



*Figura 4.16 Efecto de la proporción Componentes/Mezclas en la calidad de la separación sobre el Conjunto_II*

La figura 4.16 ilustra el efecto de la proporción Componentes/Mezclas sobre separaciones con número variable de componentes del Conjunto_II de datos, estas mezclas tienen un índice medio de dispersión de 0,86. Llama atención la similitud de esta figura con la figura 4.8 que pertenece a mezclas con un índice de dispersión parecido, esto nos lleva a pensar que el efecto de esta proporción es similar sea cual sea el conjunto de datos analizado y, por lo tanto, que no guarda relación con la diversidad morfológica de los componentes.

### 4.6.4 Evaluación respecto a la dimensionalidad de los componentes.

En esta última evaluación se repite la prueba realizada en el punto 4.5.4. Para ello se emplearán, en un escenario sin ruido añadido, unos conjuntos de prueba compuestos por 25 mezclas generadas mediante matriz de mezcla aleatoria y 5 componentes de dimensionalidad variable.

Para esta prueba se ejecuta cada algoritmo por separado utilizando el mismo conjunto de datos, y se repite el proceso en 30 iteraciones generando un conjunto de pruebas distinto en cada una de ellas, variando en cada ocasión la longitud de los componentes de 64 a 2048 muestras. Los resultados de esta prueba pueden reproducirse ejecutando el script de Matlab '*Test4_II*' que se adjunta a esta memoria y son los que se muestran en la figura 4.17.

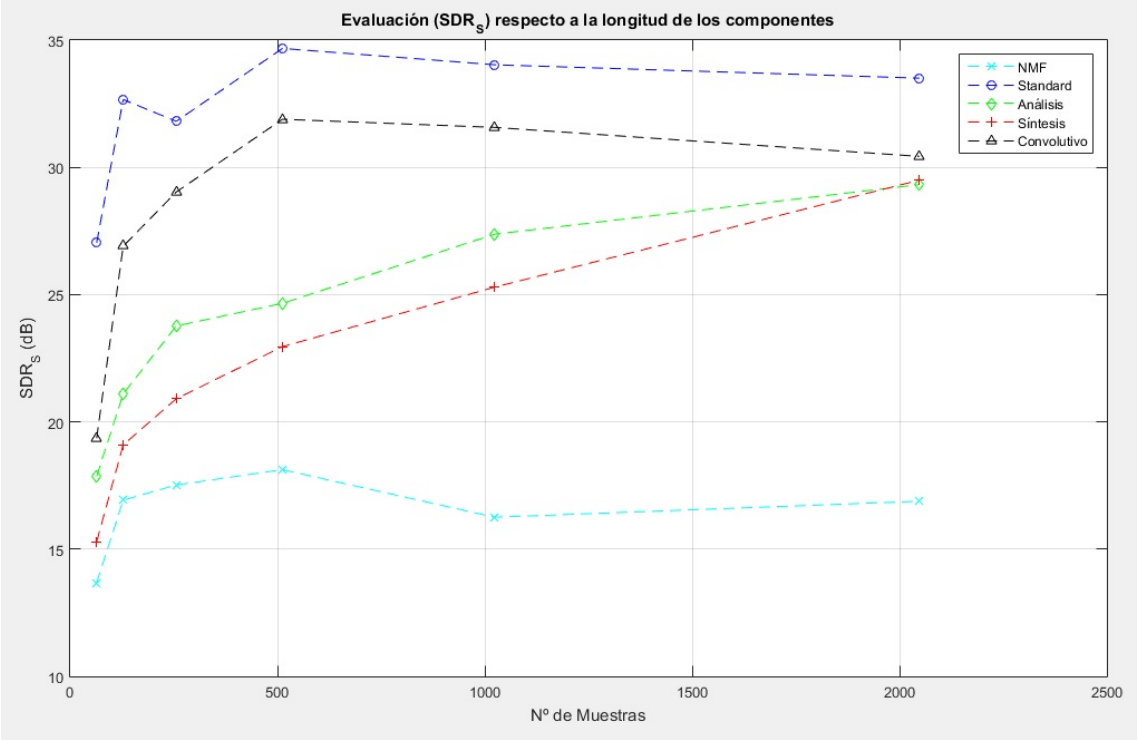



*Figura 4.17 Resultados de la evaluación respecto a la longitud de los componentes 'Test4_II'*

Destaca en esta prueba la sustancial mejora en el rendimiento que se produce en los algoritmos nGMCA$^{\text{Analítico}}$ y nGMCA$^{\text{Sintético}}$ cuando la longitud de los componentes aumenta,

llegando a equipararse a la calidad de separación de los algoritmos que destacaban por su precisión con el conjunto anterior de componentes.

### 4.6.5 Análisis temporal de pruebas sobre el Conjunto_II de componentes.

Como ilustra la figura 4.18, podemos encontrar poca diferencia en el rendimiento de los algoritmos evaluados en el punto 4.5.6 con respecto a su tiempo de ejecución. Se sigue constatando el aumento exponencial del tiempo de ejecución respecto al número de componentes a separar y el aumento lineal con respecto a la longitud de las mezclas.

Cabe destacar, sin embargo, la mejora que se produce en los tiempos de separación de los algoritmos $nGMCA^{Analítico}$ y $nGMCA^{Sintético}$ con este conjunto de componentes, y el ligero empeoramiento del $nGMCA^{Standard}$ en este escenario. Por otra parte, el algoritmo NMF sigue mostrando una velocidad muy superior al resto.

Al igual que en el punto 4.5.6, para realizar las medidas de tiempo se han ejecutado los algoritmos mediante un sistema con procesador Intel Core i5-4460 de cuatro núcleos y frecuencia de reloj de 3.20Ghz.

### 4.6.6 Conclusiones de las pruebas sobre Conjunto_II de componentes.

Como se ha podido apreciar mediante esta segunda batería de pruebas, existe una variación en el comportamiento de los algoritmos dependiendo de la diversidad morfológica de las fuentes. El cambio más importante se ha podido observar en el comportamiento con respecto al número de mezclas, en el que se aprecia un cambio de tendencia en la calidad llegado cierto número de mezclas.

En lo que respecta al rendimiento de los algoritmos, por un lado, el rendimiento de los algoritmos $nGMCA^{Analítico}$ y $nGMCA^{Sintético}$ ha mejorado notablemente, llegando a equipararse al algoritmo $nGMCA^{Standard}$ en algunos casos. También se ha notado una mejora de estos algoritmos en su rendimiento temporal, aunque sigue siendo notablemente peor que el del resto.

También ha mejorado el desempeño en la calidad de separación del $nGMCA^{Convolutivo}$, el cual obtiene unos resultados similares a los del algoritmo $nGMCA^{Standard}$ pero con un menor tiempo de ejecución, igualando así el compromiso Tiempo/Calidad en el cual destacaba el primero con el Conjunto_I.

Por otro lado, el algoritmo NMF sigue manteniendo su velocidad de ejecución y su calidad de separación, pero en este escenario de prueba su rendimiento queda a bastante distancia del resto. Sin duda, este es un algoritmo que puede aplicarse a un gran número de escenarios con unos resultados aceptables, pero no es el mejor en ninguno de ellos.

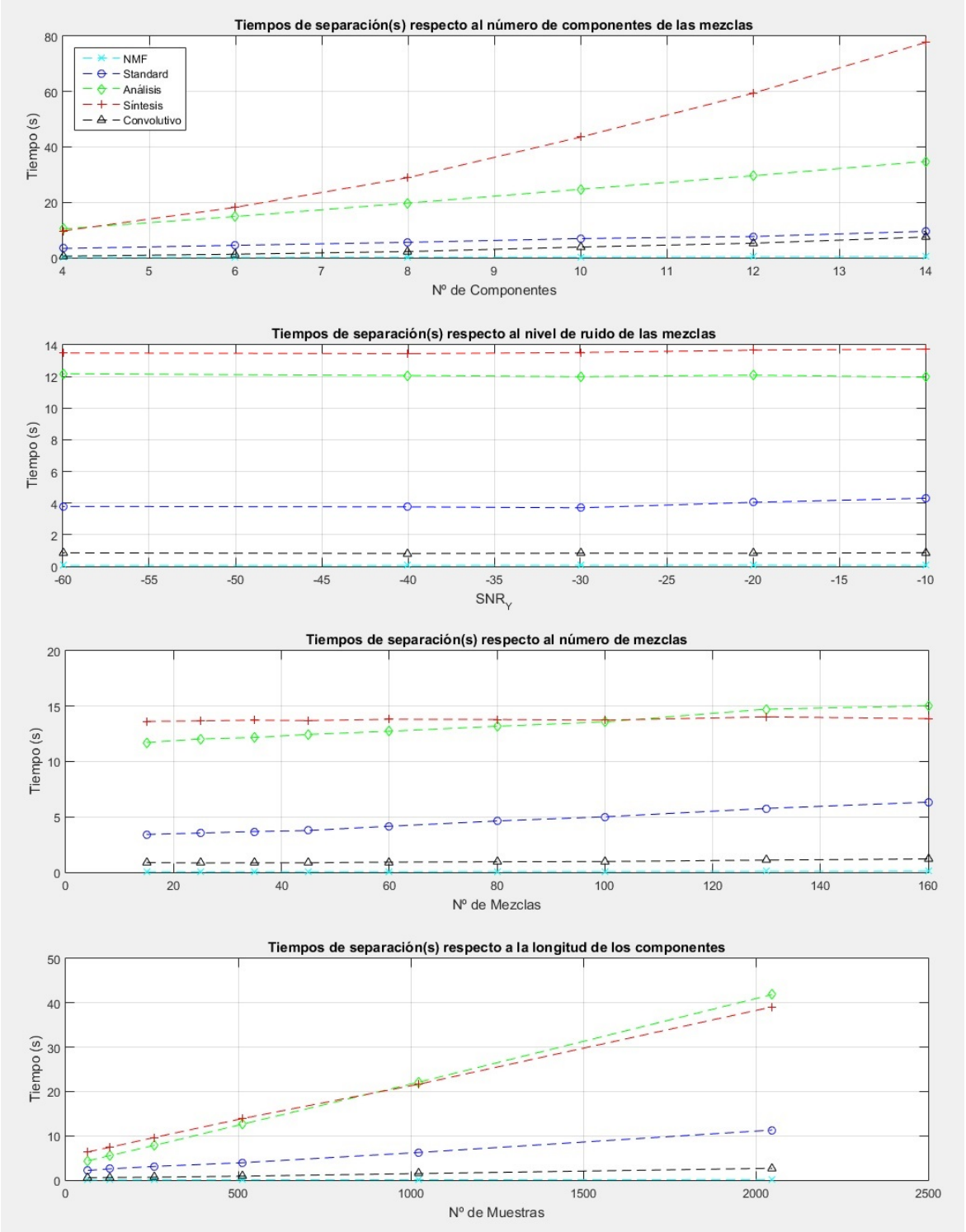



*Figura 4.18 Coste temporal de ejecución de los algoritmos en pruebas de evaluación sobre Conjunto_II*

La tabla 4.2 recoge la comparativa de los resultados de los diferentes test de esta sección. Se muestran los valores medios de reconstrucción conseguidos por cada uno de los algoritmos en las diferentes pruebas realizadas sobre el Conjunto_II de componentes.

| Algoritmo / Test | *Componentes* | *$SNR_Y$* | *Mezclas* | *Muestras* |
|---|---|---|---|---|
| ***NMF*** | *17.10 dB* | *13.81 dB* | *16.57 dB* | *16.56 dB* |
| ***$nGMCA^{Standard}$*** | ***29.10 dB*** | ***22.80 dB*** | ***29.34 dB*** | ***32.28 dB*** |
| ***$nGMCA^{Analítico}$*** | *22.33 dB* | *19.00 dB* | *22.35 dB* | *24.01 dB* |
| ***$nGMCA^{Sintético}$*** | *19.81 dB* | *17.60 dB* | *20.46 dB* | *22.17 dB* |
| ***$nGMCA^{Convolutivo}$*** | *27.02 dB* | *21.61 dB* | *26.48 dB* | *28.19 dB* |

*Tabla 4.2 Comparativa de resultados SDR medios de reconstrucción por algoritmo y test, Conjunto_II.*

Por último, cabe destacar que el curioso efecto que tiene la proporción Componentes/Mezclas en la calidad de la separación no guarda relación con la morfología de los componentes o las mezclas, siendo este un efecto que se comporta de forma similar en los dos conjuntos de datos analizados hasta el momento. Esto nos lleva a pensar que se trate de una consecuencia del uso de estos algoritmos.

## 4.7 Separación sobre el conjunto de Componentes_III.

El objetivo principal de este trabajo es encontrar la mejor manera de separar los diferentes compuestos químicos que integran una mezcla. En los puntos anteriores se ha pretendido evaluar, en diferentes escenarios, el comportamiento de los algoritmos candidatos para esta tarea. Los conjuntos de componentes Conjunto_I y Conjunto_II se han empleado para poder modificar diferentes condiciones y, de este modo, poder crear diferentes escenarios de evaluación, pero estos conjuntos de componentes no dejan de ser espectros simulados que carecen de la complejidad morfológica de los espectros FTIR reales.

El Conjunto_III, es un grupo de 5 espectros de componentes reales con los que evaluaremos el comportamiento de los candidatos ante mezclas de mayor complejidad. Estos espectros se han recogido mediante un espectrofotómetro JASCO FT/IR-4100 typeA con una resolución de 4 $cm^{-1}$ y serán los empleados en el caso real que analizaremos en el siguiente capítulo.

En este punto evaluaremos los algoritmos en dos supuestos. Comprobaremos cómo se comportan frente a mezclas de diversas dimensiones y también se realizarán pruebas para comprobar si se puede seleccionar solo una fracción del espectro para mejorar la separación. En este punto no se realizará análisis temporal, ya que su comportamiento se ha evidenciado en las secciones anteriores.

### 4.7.1 Método de reconstrucción de los componentes.

Antes de comenzar con las pruebas se debe hacer una puntualización con respecto al modo en el que se recuperan los espectros de los componentes en esta sección.

En los apartados 4.5.3 y 4.6.3 se ha puesto de manifiesto el efecto que tiene la proporción Componentes/Mezclas sobre la calidad de la separación de los componentes. Se ha visto como existe una proporción determinada en la que la separación tiene una calidad máxima mientras que en proporciones mayores o menores la calidad de separación se deteriora de manera ostensible. Este deterioro no es debido a una identificación errada de los diferentes componentes, sino que se trata de un problema de escala en la reconstrucción de los espectros. Como se observa en la figura 4.19, la reconstrucción de los componentes en proporciones Componentes/Mezclas menores a la proporción optima genera espectros con un perfil adecuado, pero con una escala menor a la original, mientras que si la separación se realiza con proporciones superiores a la óptima se generan espectros con una escala mayor a la original.

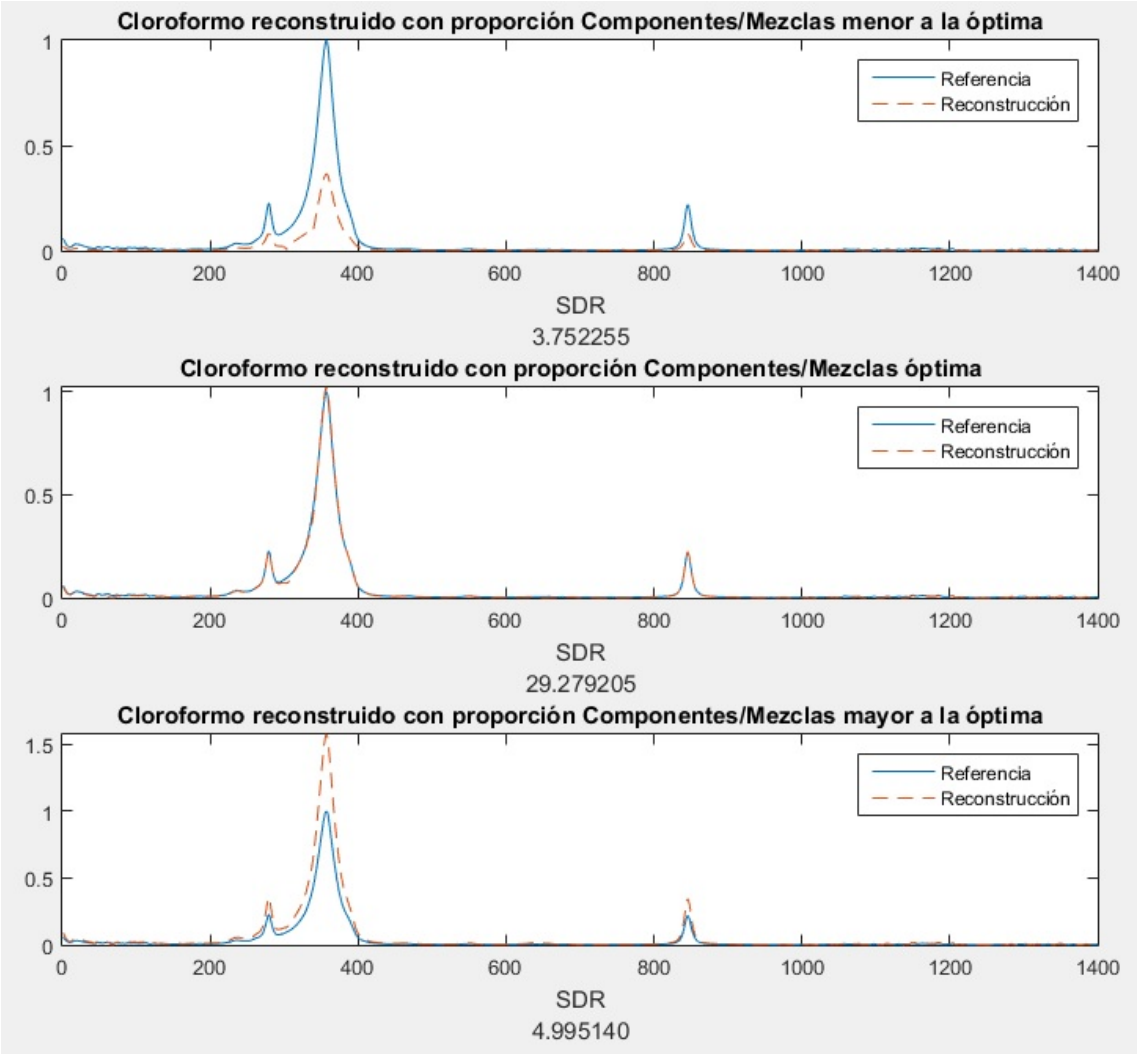



*Figura 4.19 Efecto de la proporción Componentes/Mezcla en la escala de la reconstrucción*

Como también vimos en el punto 4.5.3, este efecto es más agudo en mezclas con mayor dispersión, esto se debe a que el cambio de escala es más pronunciado en los puntos más energéticos de los espectros, los cuales son característicos en componentes y mezclas más

dispersos. Esto genera que el cálculo del SDR de la reconstrucción de un componente con un índice de dispersión mayor se vea más afectado, ya que la diferencia en el perfil de su espectro con respecto al del componente original es más pronunciada que en el caso de componentes con espectros que tengan perfiles con puntos menos energéticos.

El efecto generado por la proporción componentes/Mezclas sobre la calidad de la separación tiene su origen en el modo que emplean los algoritmos basados en NMF, como los estudiados en este trabajo, para minimizar el efecto de una de sus limitaciones. Como vimos en el punto 2.5.1 una de las limitaciones del algoritmo NMF es la no unicidad de las soluciones que genera, en la práctica, esta limitación se corrige mediante la estandarización de las columnas de la matriz **A**, recuperada en cada paso del algoritmo, mediante la norma $\ell_2$. Esta estandarización introduce un cambio de escala que depende del tamaño de la matriz **A** con respecto a la matriz **S**, también recuperada en cada iteración del algoritmo [Aoulass and Chakkour, 2020], o lo que es lo mismo la proporción Componentes/Mezclas. Otra forma de evitar la no unicidad de las soluciones de NMF es la estandarización, mediante la norma $\ell_2$, de las filas de la matriz S, pero se introduce el mismo efecto en la escala que en el caso anterior.

Las pruebas que se van a realizar sobre el Conjunto_III de datos tienen como objetivo asemejarse a un escenario real como el que se evaluará en el capítulo 5, por lo que se realizarán siempre con una proporción Componentes/Mezcla subóptima, ya que el número de mezclas del que se dispone es limitado. Por lo tanto, para evitar la pérdida de calidad con respecto a la escala introducidas por el efecto mencionado anteriormente, en las separaciones que se realicen sobre el Conjunto_III, se reconstruirán los componentes a partir de una matriz de mezcla ponderada $\boldsymbol{A_p}$ mediante el método clásico de mínimos cuadrados inverso tal y como indica la ecuación (4.3), como el que se emplea en el Algoritmo 1 para generar **S** a partir de **A** e **Y**.

$$S = \left(A_p^T A_p\right)^{-1} A_p^T Y \tag{4.3}$$

La matriz ponderada $\boldsymbol{A_p}$ no es más que la matriz recuperada por el algoritmo, sometida a un procesado mediante el cual, manteniendo la proporcionalidad de las concentraciones obtenidas como resultado de la separación, se consigue que la suma de concentraciones de los componentes de cada mezcla sea 1, o lo que es lo mismo, que la suma total de cada fila de la matriz $\boldsymbol{A}$ sea la unidad. Una muestra de los beneficios de este método se ilustra en la figura 4.20, donde podemos ver que la reconstrucción sobredimensionada del cloroformo afecta enormemente a su medida de calidad, corrigiéndose, en gran medida, mediante el uso del procesado de la matriz de mezcla ponderada $\boldsymbol{A_p}$ .

La ponderación de la matriz de mezcla la lleva a cabo la función de Matlab '*ProcesaM*' que se adjunta a este trabajo.

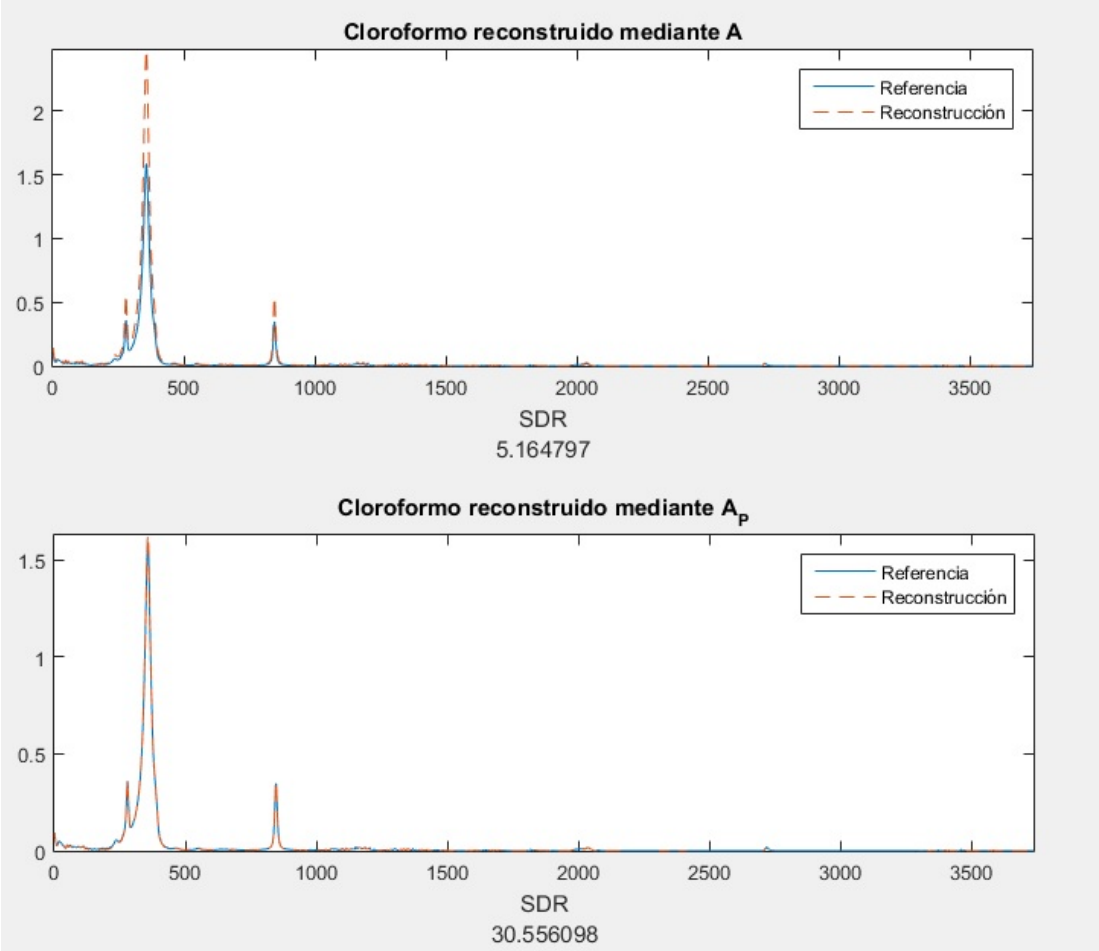



*Figura 4.20 Diferencia de calidad en reconstrucción de componentes mediante A y $A_p$*

## 4.7.2 Evaluación respecto a la dimensionalidad de los componentes.

En este punto comprobaremos si la reducción de la dimensión original de los componentes (3736 Muestras) afecta positiva o negativamente a la calidad de la reconstrucción de estos. Para ello se emplearán unos conjuntos de prueba compuestos por 15 mezclas generadas mediante matriz de mezcla $\boldsymbol{A}$ (4.4) y los 5 componentes del Conjunto_III. Variando su longitud de 512 a 3736 muestras.

Se realizan 30 iteraciones de los algoritmos para calcular los valores medios de salida ya que, aunque los datos de entrada son los mismos, la separación sufre pequeñas variaciones con cada ejecución. Los resultados de esta prueba pueden reproducirse ejecutando el script de Matlab '*Test1_III*' que se adjunta a esta memoria y son los que se muestran en la figura 4.21.

$$\boldsymbol{A} = \begin{pmatrix} 0.1 & 0.4 & 0.5 & 0 & 0 \\ 0 & 0.8 & 0.2 & 0 & 0 \\ 0.4 & 0.4 & 0.2 & 0 & 0 \\ 0.3 & 0.3 & 0.4 & 0 & 0 \\ 0.1 & 0.1 & 0.3 & 0.1 & 0.2 \\ 1 & 0.1 & 0.2 & 0.3 & 0.3 \\ 0 & 0.2 & 0 & 0.3 & 0.5 \\ 0.2 & 0.2 & 0.2 & 0.2 & 0.2 \\ 0.2 & 0.3 & 0.1 & 0.1 & 0.3 \\ 0.4 & 0.1 & 0.2 & 0.2 & 0.1 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix} \tag{4.4}$$

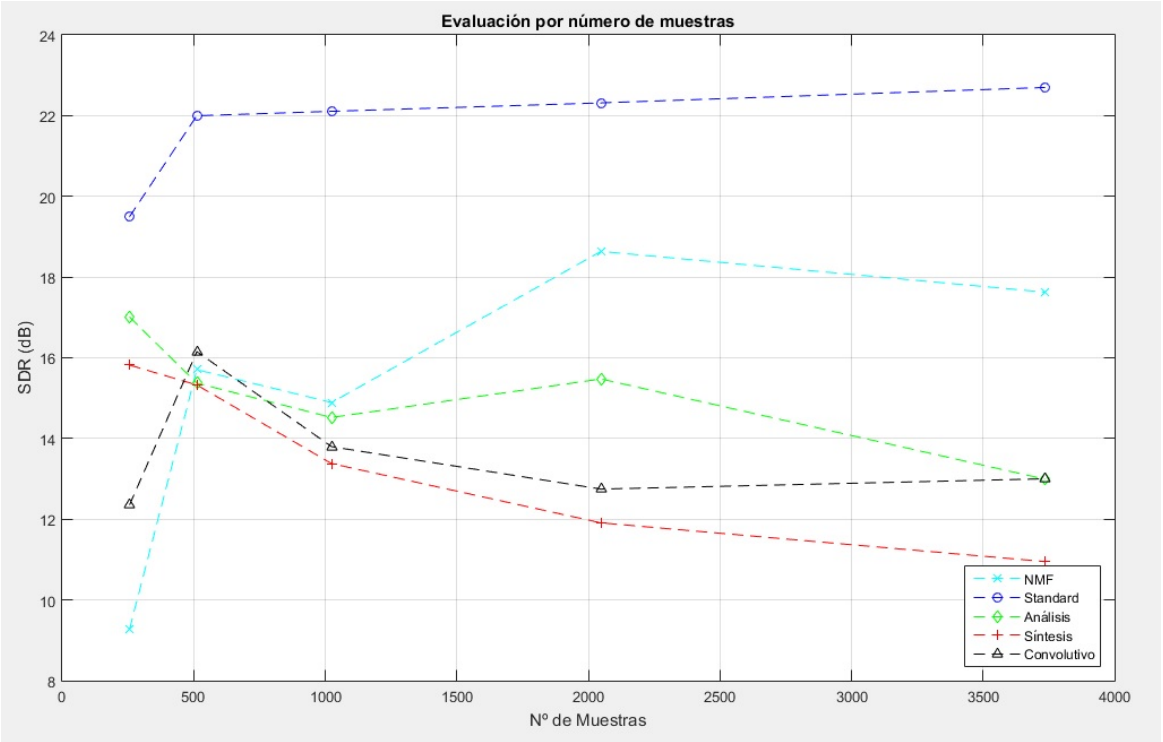



*Figura 4.21 Resultados de la evaluación respecto a la longitud de los componentes 'Test1_III'*

Se observa como el aumento en el número de muestras afecta de manera dispar a los algoritmos. Si bien los algoritmos NMF y nGMCA$^{\text{Standard}}$ parecen beneficiarse en cierto modo del aumento en el número de muestras, el resto de algoritmos sufren una pérdida de calidad en la separación según aumenta la longitud de estas.

### 4.7.3 Fracciones del espectro.

El espectro FTIR de un compuesto abarca longitudes de onda entre 25000 y 2500 nm (4000 a 400 $cm^{-1}$). Como ilustra la figura 4.21, el espectro de cada componente está dividido en dos zonas principalmente. Por un lado, posee una región llamada "zona de grupos funcionales" [4000-1400 $cm^{-1}$], la cual nos indica a que grupo funcional pertenece el compuesto (Alcoholes, Aldehídos, Cetonas, Alquenos, …), y por otro posee una región conocida como "zona de huella dactilar" [1400 - 400 $cm^{-1}$], la cual identifica al compuesto dentro de cada grupo funcional, y que es la región morfológicamente más compleja.

La figura 4.22 muestra el ancho del espectro IR donde se pueden distinguir las dos zonas especificadas anteriormente.

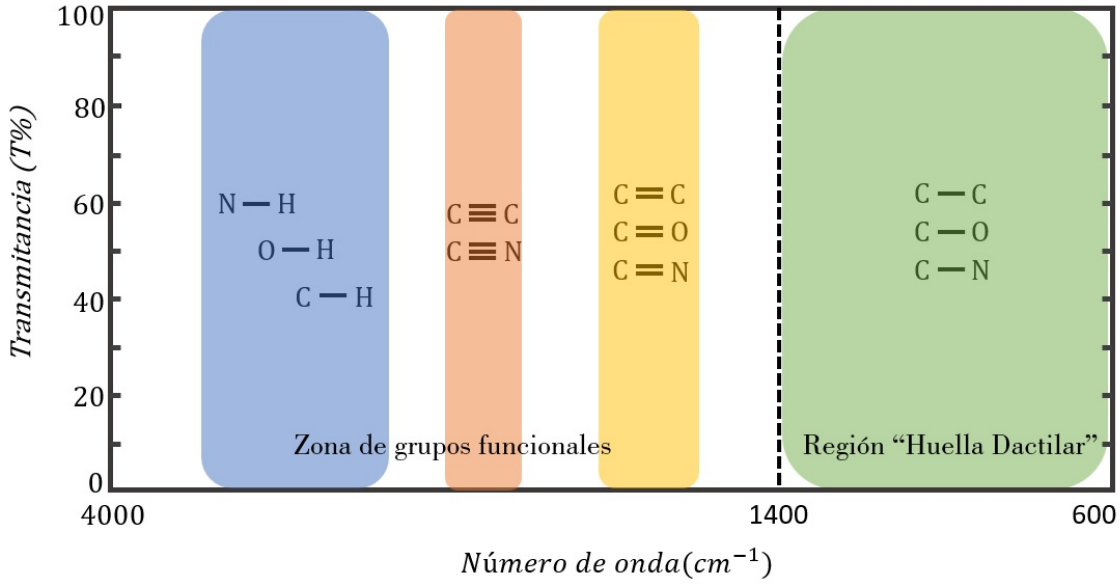



*Figura 4.22 Gráfico de regiones del espectro FTIR de un compuesto.*

Lo que se pretende con esta prueba es comprobar si los algoritmos separan con mayor calidad los componentes de una mezcla utilizando solamente la zona de la huella dactilar o si, por el contrario, las separaciones de componentes son mejores empleando el espectro completo. Para ello se empleará un conjunto de prueba compuestos por 15 mezclas generadas mediante matriz de mezcla $\boldsymbol{A}$ (4.3) y los 5 componentes del Conjunto_III en su longitud original de 3736 puntos de muestra.

Se realizan 30 iteraciones de los algoritmos para calcular los valores medios de salida ya que, aunque los datos de entrada son los mismos, la separación sufre pequeñas variaciones con cada ejecución. Los resultados de esta prueba pueden reproducirse ejecutando el script de Matlab '*Test2_III*' que se adjunta a esta memoria y son los que se muestran en la figura 4.23.

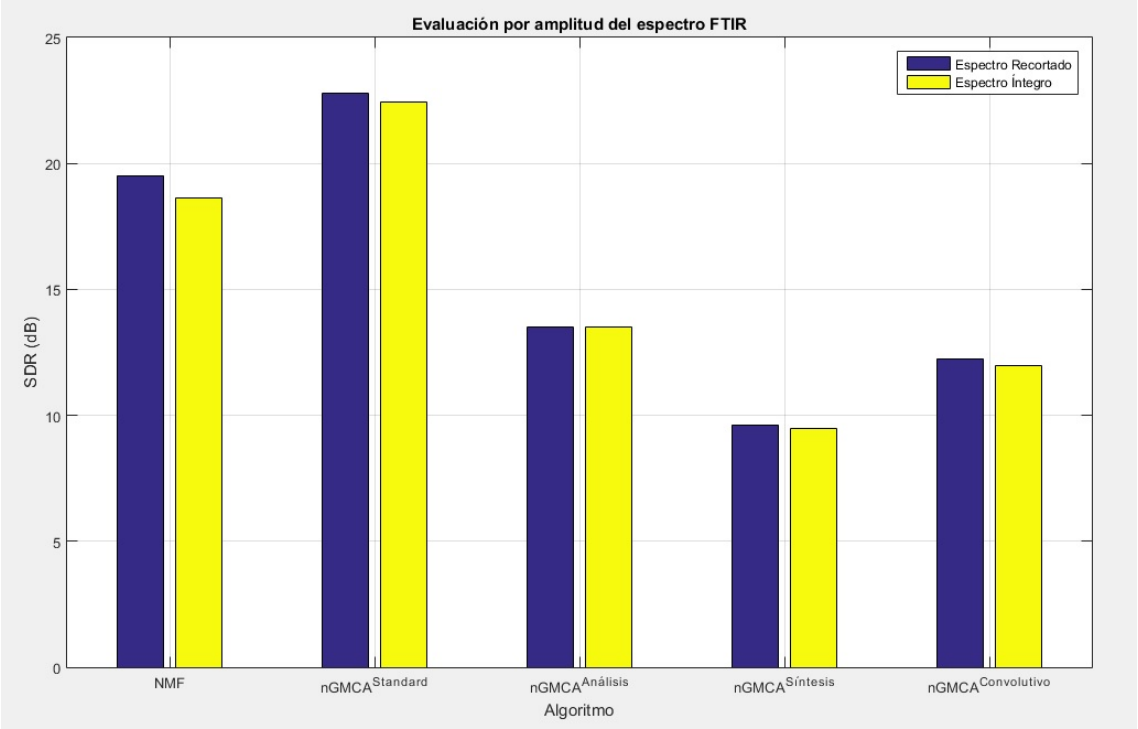



*Figura 4.23 Comparativa de calidad de reconstrucción mediante diferentes secciones del espectro FTIR 'Test2_III'*

Como se observa en la figura 4.23, no existe una diferencia determinante entre la separación de componentes mediante el uso del espectro recortado o mediante el espectro íntegro, aunque sí parece haber una ligera ventaja en usar el espectro recortado cuando se utilizan los métodos NMF y nGMCA$^{Standard}$.

## 4.8 Selección final de los candidatos.

El objetivo principal de las diferentes pruebas realizadas sobre los cinco algoritmos evaluados anteriormente era seleccionar uno o varios candidatos para llevar a cabo la separación de componentes en una prueba práctica real. Debe hacerse notar que la parametrización de estos algoritmos es una tarea bastante delicada, por este motivo, se ha decidido realizar las evaluaciones con los parámetros por defecto que recomiendan sus autores y que se recogen en los artículos que los describen, citados en la sección 4.2.

Hemos podido ver a lo largo de este capítulo que existe un claro candidato para procesar el caso práctico que se propone en el capítulo 5. El algoritmo nGMCA$^{Standard}$ ha destacado por la calidad de sus reconstrucciones en los diferentes conjuntos de pruebas y en todos y cada uno de los escenarios propuestos. Es por eso que, en el siguiente capítulo, se empleará este algoritmo.

Por otro lado, un algoritmo que se ha mantenido muy estable en las separaciones de las secciones anteriores ha sido el NMF. Este dato, junto a la velocidad a la que es capaz de procesar una mezcla, le convierten en un algoritmo muy robusto, válido para emplearse en multitud de escenarios.

El algoritmo nGMCA$^{\text{Convolutivo}}$, si bien parecía un buen candidato debido a su compromiso entre velocidad de ejecución y calidad de reconstrucción, no parece adaptarse bien a las mezclas producidas mediante espectros reales, por lo que debemos descartarlos para realizar separaciones en este entorno. La parametrización de este algoritmo es realmente compleja, ya que a los parámetros comunes a la familia nGMCA se unen los necesarios para transformar el dominio de representación de las mezclas, lo que supone un problema que, por su dimensión, no se ha podido abordar en este trabajo y se propondrá en la sección 6.2 como un trabajo futuro a desarrollar en esta área. Es posible que mediante el uso de una transformación adecuada se pueda mejorar el desempeño de este algoritmo para este problema particular.

Para terminar, debemos descartar los algoritmos nGMCA$^{\text{Analítico}}$ y nGMCA$^{\text{Sintético}}$. No solo no han destacado en la calidad de reconstrucción en ninguno de los casos de prueba, sino que, además su velocidad de reconstrucción es bastante pobre en el caso del nGMCA$^{\text{Analítico}}$, y crítica en el caso del nGMCA$^{\text{Sintético}}$.

# Capítulo 5

# Aplicación en mezclas reales

## 5.1 Conjunto de datos.

Para llevar a cabo la tarea que se aborda en este capítulo disponemos de un conjunto de espectros reales. Estos espectros han sido recogidos mediante un espectrofotómetro JASCO FT/IR-4100 typeA con accesorio ATR PRO ONE, una fuente de luz estándar con inclinación de 45º y un sensor de tipo TGS. Estos espectros emplean como unidad de medida la transmitancia óptica, poseen una resolución de 4 $cm^{-1}$ y constan de 6847 puntos de medida entre los números de onda 399 $cm^{-1}$ y 7000 $cm^{-1}$.

El conjunto espectros se divide en dos grupos. Por un lado, disponemos de 5 componentes (1-Butanol, Cloroformo, Metanol, 2-Propanol y Tolueno) y un espectro llamado Background que representa una medición en vacío. Estos espectros se ilustran en la figura 5.1.

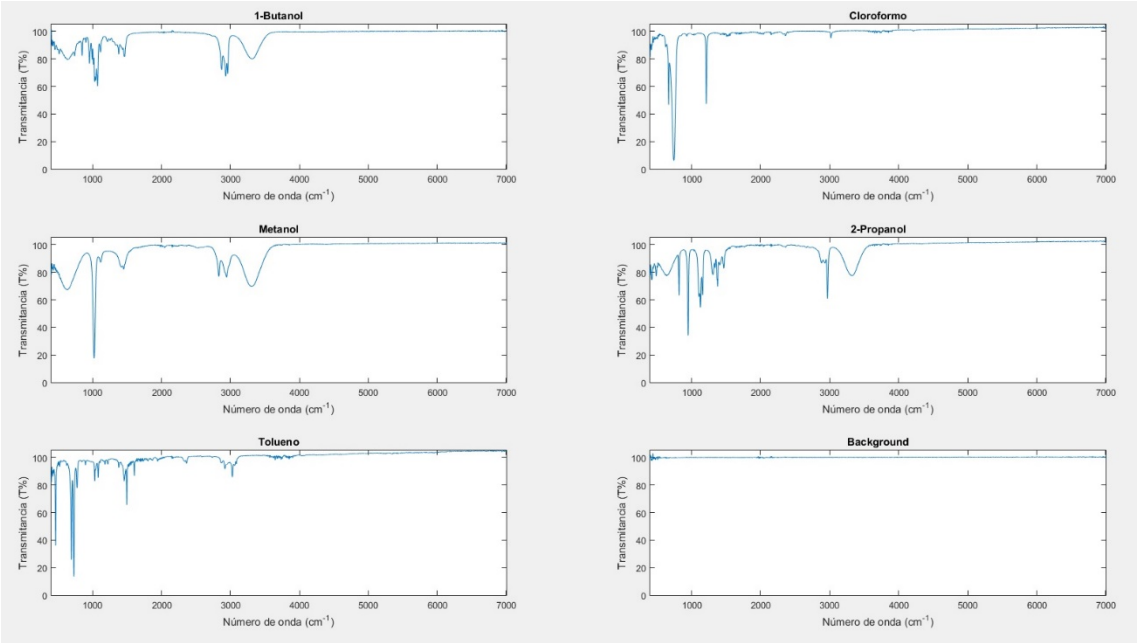



*Figura 5.1 Espectros originales de componentes reales*

Por otra parte, se dispone de 10 mezclas compuestas por diferentes concentraciones de los componentes. La concentración teórica de cada componente en cada una de las mezclas es la recogida en la tabla 5.1. Los espectros de estas mezclas se representan en la figura 5.2.

Cada uno de los espectros originales se ha obtenido del cálculo de la media aritmética de, al menos, tres medidas diferentes.

| | *1-Butanol* | *Cloroformo* | *Metanol* | *2-Propanol* | *Tolueno* |
|---|---|---|---|---|---|
| ***Mezcla 1*** | *10%* | *40%* | *50%* | *0%* | *0%* |
| ***Mezcla 2*** | *0%* | *80%* | *20%* | *0%* | *0%* |
| ***Mezcla 3*** | *40%* | *40%* | *20%* | *0%* | *0%* |
| ***Mezcla 4*** | *30%* | *30%* | *40%* | *0%* | *0%* |
| ***Mezcla 5*** | *30%* | *10%* | *30%* | *10%* | *20%* |
| ***Mezcla 6*** | *10%* | *10%* | *20%* | *30%* | *30%* |
| ***Mezcla 7*** | *0%* | *20%* | *0%* | *30%* | *50%* |
| ***Mezcla 8*** | *20%* | *20%* | *20%* | *20%* | *20%* |
| ***Mezcla 9*** | *20%* | *30%* | *10%* | *10%* | *30%* |
| ***Mezcla 10*** | *40%* | *10%* | *20%* | *20%* | *10%* |

*Tabla 5.1 Concentración teórica de componentes en cada mezcla*

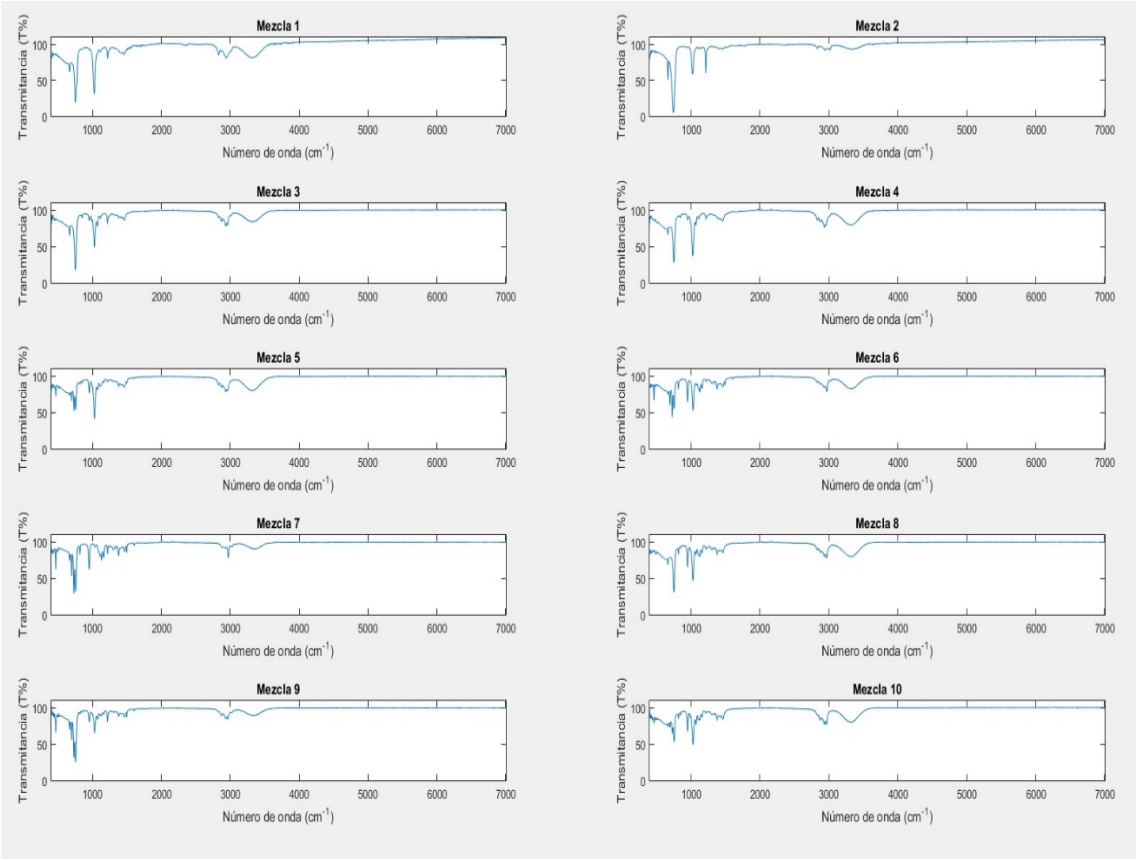



*Figura 5.2 Espectros originales de mezclas reales*

## 5.2 Procesado de los espectros.

Para facilitar la separación en este caso práctico, debemos emplear las conclusiones extraídas de las pruebas del capítulo 4 para procesar los espectros y acondicionarlos del mejor modo posible al algoritmo de separación.

Se ha concluido que una mayor dispersión de los componentes mejora la calidad de la separación del algoritmo nGMCA$^{Standard}$ por lo que debemos intentar representar los componentes y las mezclas de la manera más dispersa posible sin alterar la información que contienen. Como se ha indicado anteriormente los espectros originales vienen expresados en una unidad de medida llamada "transmitancia" ($T\%$), pero existe la posibilidad de emplear otra unidad de medida derivada de esta llamada "absorbancia" o "densidad óptica" ($A_\lambda$). La absorbancia de un compuesto se obtiene de la transmitancia según se expresa en la fórmula (5.1).

$$A_\lambda = -10 \log_{10}(T\%) \tag{5.1}$$

Se puede observar la enorme mejora que se consiguen mediante la representación de la absorbancia, tanto de los componentes como de las mezclas en la tabla 5.2.

| | *Dispersión* (T%) | *Dispersión* ($A_\lambda$) |
|---|---|---|
| ***1-Butanol*** | *0.0299* | *0.7773* |
| ***Cloroformo*** | *0.0165* | *0.9377* |
| ***Metanol*** | *0.0417* | *0.7994* |
| ***2-Propanol*** | *0.0320* | *0.8109* |
| ***Tolueno*** | *0.0153* | *0.8879* |
| ***Mezcla 1*** | *0.0350* | *0.8351* |
| ***Mezcla 2*** | *0.0248* | *0.8852* |
| ***Mezcla 3*** | *0.0308* | *0.8079* |
| ***Mezcla 4*** | *0.0341* | *0.8038* |
| ***Mezcla 5*** | *0.0349* | *0.7475* |
| ***Mezcla 6*** | *0.0297* | *0.7675* |
| ***Mezcla 7*** | *0.0238* | *0.7984* |
| ***Mezcla 8*** | *0.0334* | *0.7840* |
| ***Mezcla 9*** | *0.0254* | *0.7943* |
| ***Mezcla 10*** | *0.0317* | *0.7890* |

*Tabla 5.2 Dispersión según representación de componentes y mezclas*

Por otra parte, se ha visto en el capítulo 4 como mediante el uso de la zona del espectro llamada "región de huella dactilar" se consigue una ligera mejora en la calidad de las separaciones, al menos con este conjunto de componentes. Es por esto que vamos a tratar el caso práctico solamente con la banda 400-1400 $cm^{-1}$ de los espectros.

Por último, para filtrar el posible ruido generado por el aparato de medida, se ha eliminado el impacto que pudiera tener el componente de Background de cada una de las mezclas. Con todo lo anterior se consiguen unos espectros como se muestran en las figuras 5.3 y 5.4.

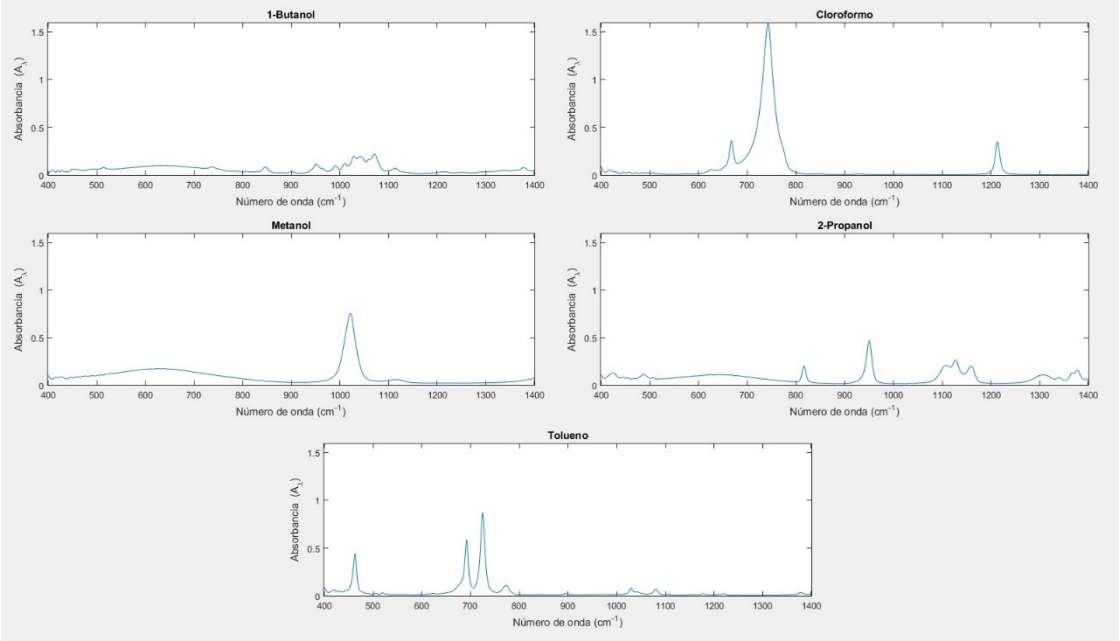



*Figura 5.3 Espectros refinados de componentes originales*

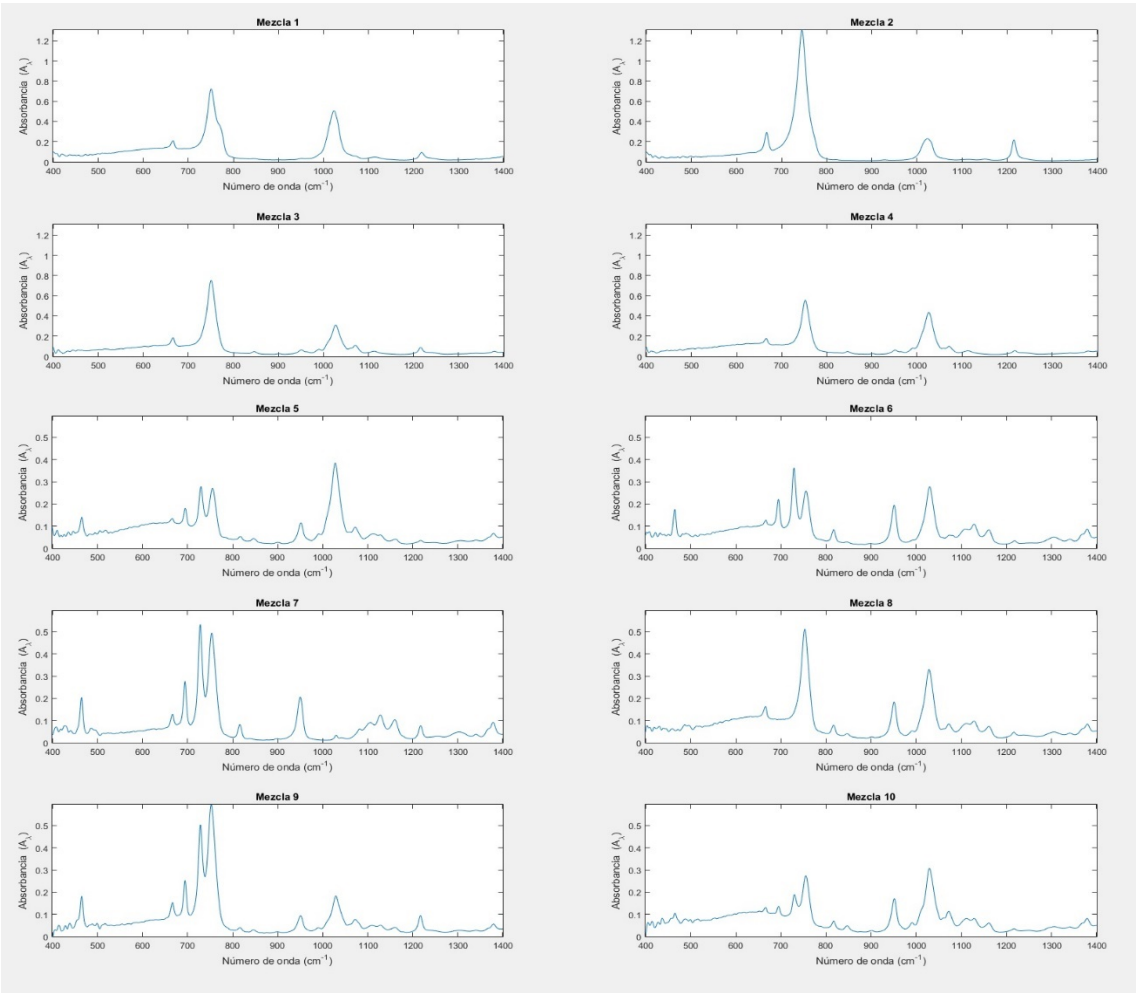



*Figura 5.4 Espectros refinados de mezclas reales*

## 5.3 Separación de componentes.

Para poder valorar en su justa medida la calidad de la separación realizada por el algoritmo nGMCA$^{Standard}$ en este entorno, debemos disponer de una referencia que nos permita comparar el correcto desempeño de este. Para ello se ha realizado la separación con un conjunto generado matemáticamente mediante los espectros de los componentes refinados mostrados en la figura 5.3 y la matriz de mezclas (4.3), empleando la fórmula (1.3). El resultado de la separación con este conjunto de datos se muestra en la figura 5.5. Esta separación muestra una aceptable calidad de separación con un SDR medio de 23,74 dB.

Posteriormente se realiza la separación mediante el mismo algoritmo utilizando el conjunto de mezclas reales mostradas en el punto anterior. Los resultados de esta separación se muestran en la figura 5.6 y pueden reproducirse, junto a los anteriores, mediante la ejecución del script de Matlab *'Separación_1'* que se adjunta a esta memoria.

Observando los resultados de la separación con muestras reales, llama la atención la pérdida de calidad en la separación de los componentes y especialmente en el caso del 1-Butanol, el cual representa un espectro muy distinto al que debería. Debemos, por tanto, analizar las posibles causas de unos resultados tan pobres en este caso.

Se debe tener en cuenta que la única diferencia entre la separación real y la de contraste son las mezclas, de ello se deduce que el problema se encuentra en estas. Podemos realizar una comparación a simple vista de las mismas donde se observa un claro desajuste entre las mezclas ideales y las reales. Esta comparación se ilustra en la figura 5.7.

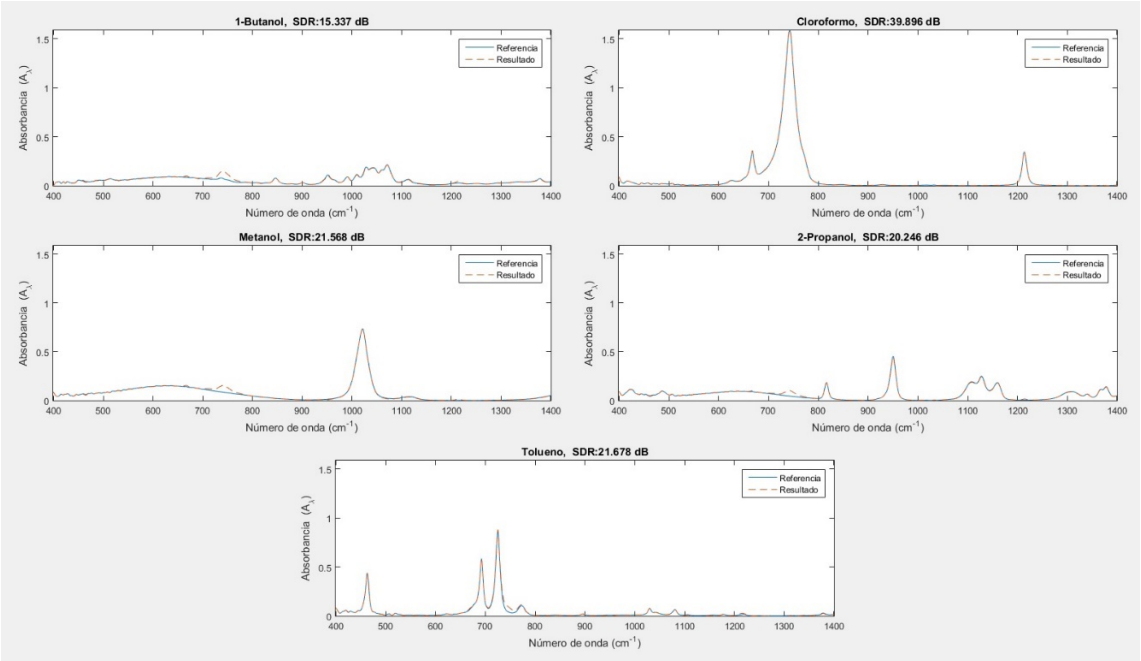



*Figura 5.5 Resultados de separación con mezclas ideales*

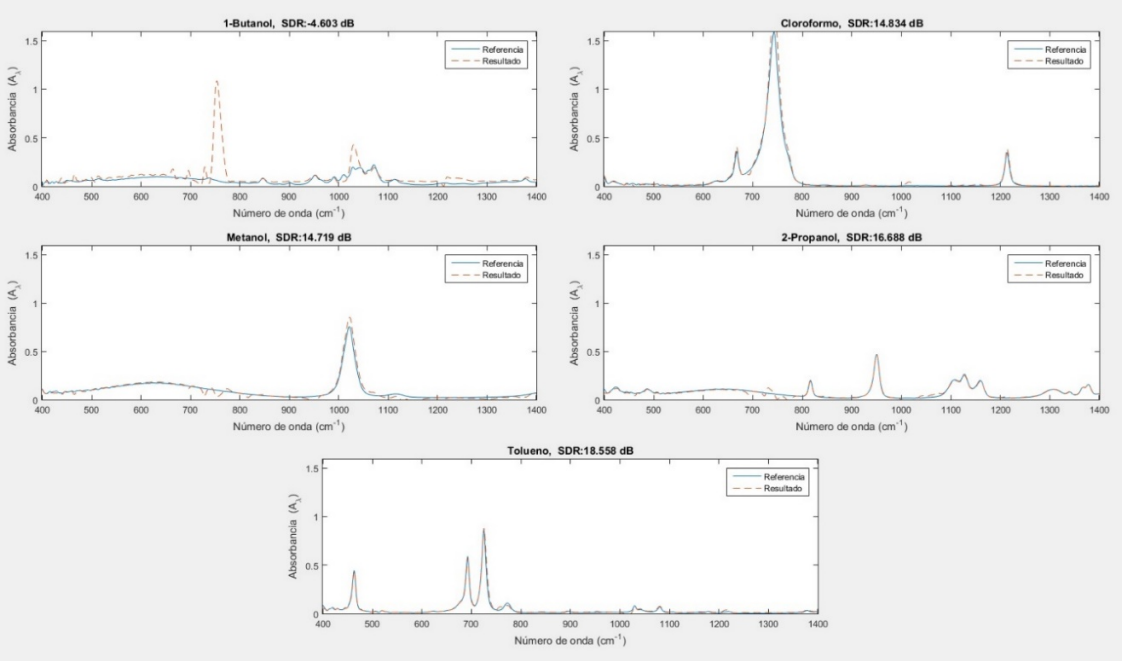



*Figura 5.6 Resultados de separación con mezclas reales*

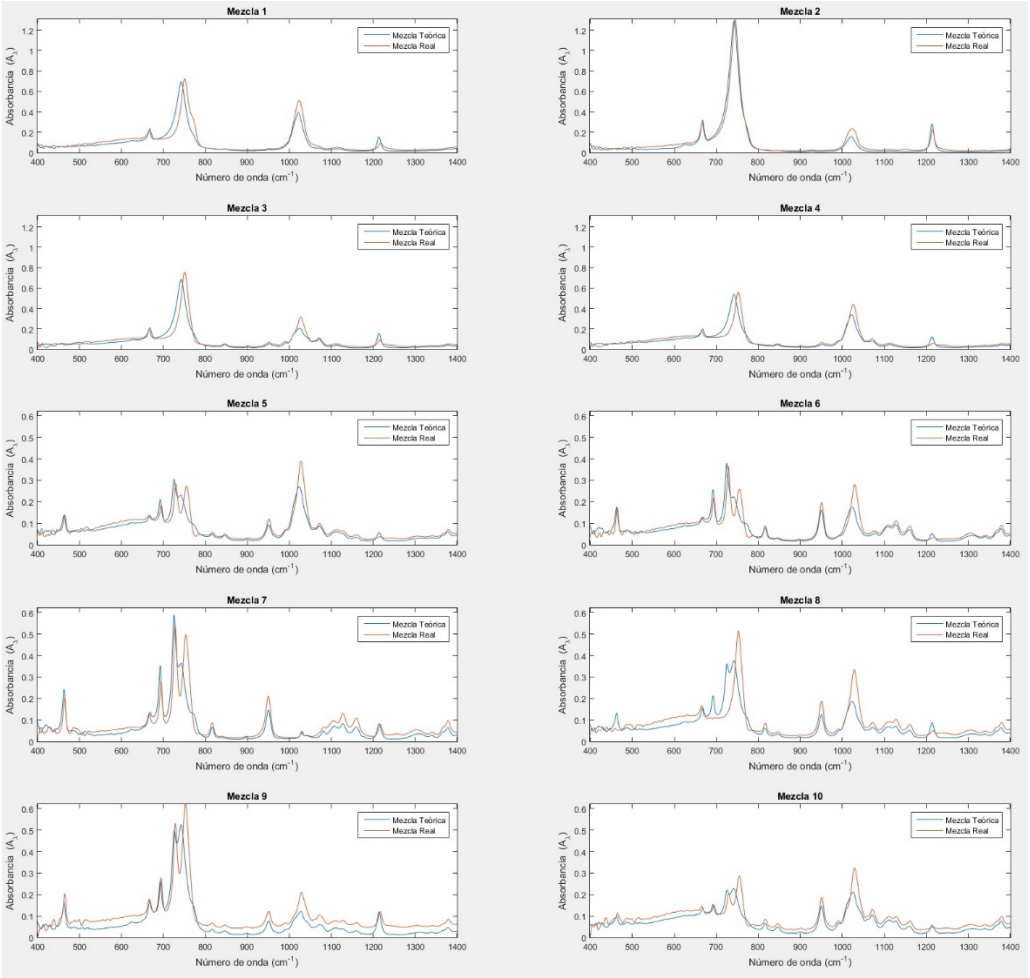



*Figura 5.7 Comparación de mezclas ideales con mezclas reales*

Por otro lado, conociendo los compuestos originales y las mezclas reales, se puede deducir la matriz de mezcla de forma directa por medio del método clásico de mínimos cuadrados tal y como indica la ecuación (5.2)

$$A = YS(S^T S)^{-1} \tag{5.2}$$

De este modo obtenemos la matriz de mezcla que se muestra a continuación en la tabla 5.3.

| | *1-Butanol* | *Cloroformo* | *Metanol* | *2-Propanol* | *Tolueno* |
|---|---|---|---|---|---|
| ***Mezcla 1*** | *22%* | *35%* | *60%* | *0%* | *-16%* |
| ***Mezcla 2*** | *10%* | *78%* | *30%* | *0%* | *-15%* |
| ***Mezcla 3*** | *51%* | *37%* | *28%* | *0%* | *-18%* |
| ***Mezcla 4*** | *46%* | *23%* | *45%* | *0%* | *-13%* |
| ***Mezcla 5*** | *45%* | *10%* | *33%* | *10%* | *10%* |
| ***Mezcla 6*** | *27%* | *10%* | *20%* | *30%* | *19%* |
| ***Mezcla 7*** | *7%* | *22%* | *0%* | *45%* | *28%* |
| ***Mezcla 8*** | *40%* | *20%* | *28%* | *27%* | *-12%* |
| ***Mezcla 9*** | *50%* | *28%* | *12%* | *21%* | *20%* |
| ***Mezcla 10*** | *64%* | *10%* | *21%* | *24%* | *0%* |

*Tabla 5.3 Concentración teórica de componentes en cada mezcla*

La tabla superior muestra unas concentraciones que nada tienen que ver con las teóricas mostradas en la tabla 5.1, encontrando incluso valores negativos lo que no sería posible en ningún caso debido a la naturaleza de las mezclas. Por lo tanto, debemos concluir que, o bien las mezclas reales no están formadas por los compuestos relatados, o bien sus concentraciones no corresponden a las informadas, o bien existe algún error en los datos de las mezclas que impiden una aceptable correlación entre los datos teóricos que poseemos y las medias reales. Tomando como ciertos los datos teóricos suministrados, centraremos el análisis en los datos de las mezclas.

Existen varios factores que pueden introducir errores en los datos de muestra. En primer lugar, se puede haber cometido algún error a la hora de transformar y refinar los datos originales. En segundo lugar, es posible que existan defectos en la toma de datos por parte del aparato de medida. Por último, es posible que no exista una relación exacta entre los datos teóricos y los recogidos en las muestras reales, por ejemplo, que no coincidan las concentraciones de las mezclas con las informadas.

El primer aspecto ha sido descartado por el autor de este trabajo después de una reconstrucción y tratamiento minucioso de los datos originales. Se ha realizado el proceso en múltiples ocasiones, por medio de diferentes métodos, generando resultados idénticos.

Contrastar un error de medida es una cuestión complicada ya que se poseen un número muy limitado de muestras y se desconocen las condiciones exactas en que están recogidas, aunque se han detectado comportamientos de las mezclas reales, como el que se ilustra en la figura 5.8, que podría achacarse al aparato de medida.

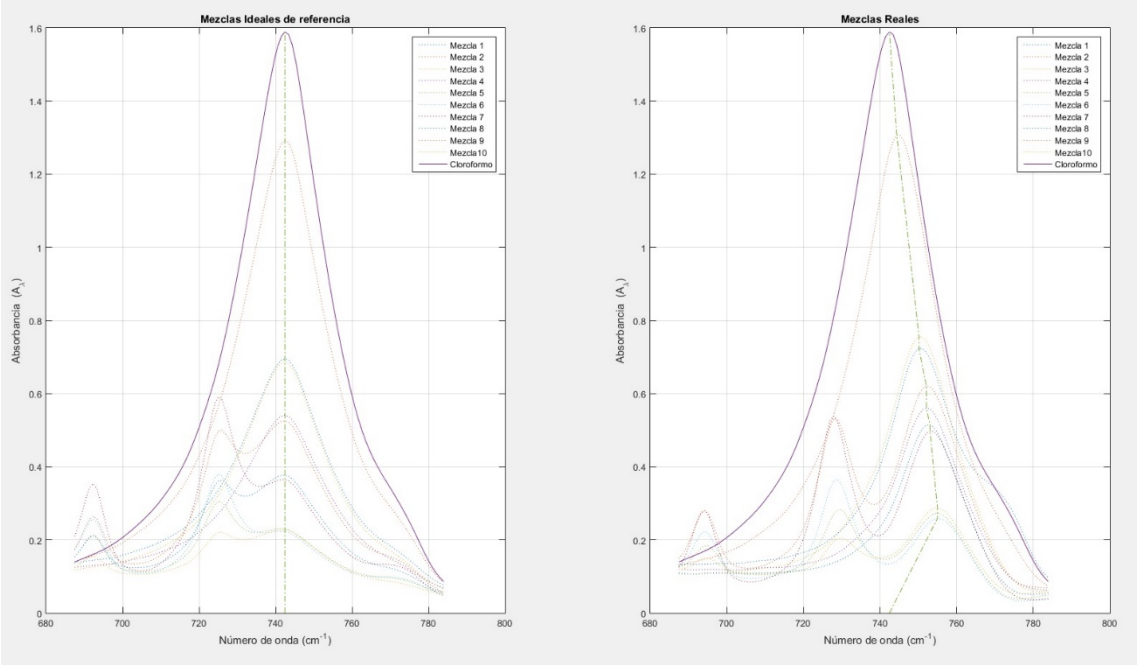



*Figura 5.8 Desplazamiento del espectro en banda 700-780 $cm^{-1}$*

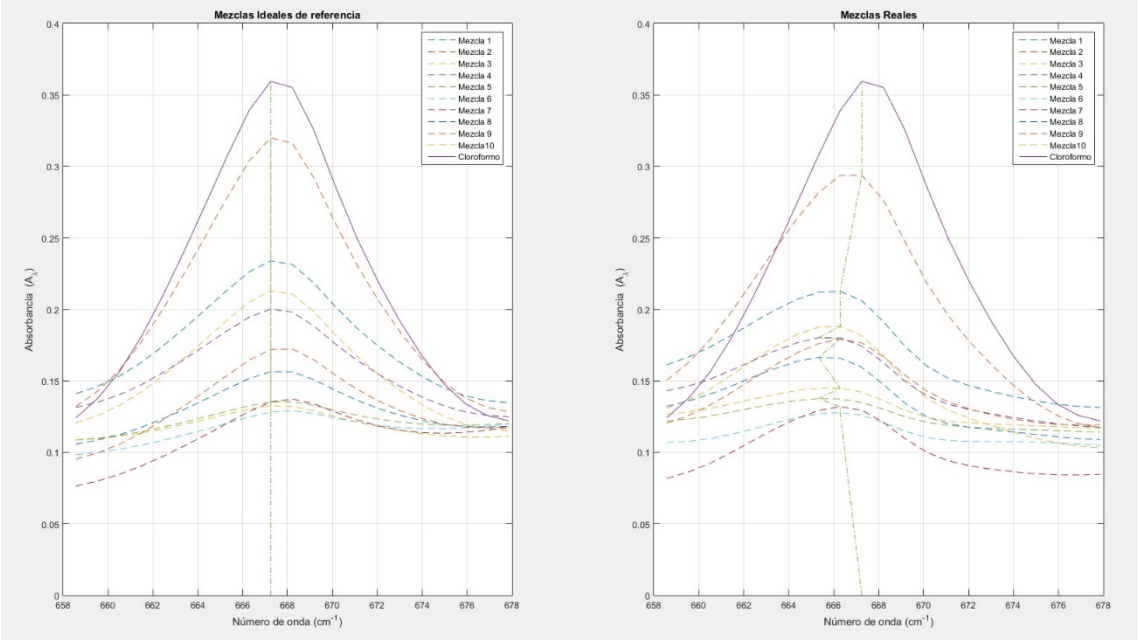



*Figura 5.9 Desplazamiento del espectro en banda 650-690 $cm^{-1}$*

En la figura 5.8 se muestra la banda de espectro comprendida entre los números de onda 700 y 780cm$^{-1}$ tanto del conjunto de mezclas ideales como del conjunto de mezclas reales. Se puede observar un claro desplazamiento del centro de los máximos de cada una de las mezclas en lo que respecta a la influencia del cloroformo en ellas. Se podría pensar que se trata de un comportamiento lineal con una tendencia de desplazamiento dependiente de la energía de los máximos. El problema es que esta tendencia no se mantiene, ya no en la influencia de otros componentes en las mezclas, sino del mismo cloroformo. En la figura 5.9 se ilustra la banda de espectro comprendida entre los números de onda 650 y 690cm$^{-1}$. En ella se puede observar como el espectro tiene la tendencia opuesta al ejemplo anterior. El caso del cloroformo es ilustrativo, pero no único, el desplazamiento de los máximos en todas las muestras con respecto a la influencia de todos los componentes sigue un patrón aparentemente aleatorio que lo hace difícil de corregir.

Otro error detectado en las mezclas reales se refiere a su homogeneidad. Cabría esperar que, aunque las concentraciones de los componentes en cada mezcla no fueran exactas, al menos si serian proporcionales, que cuando una mezcla tuviera una concentración de un componente muy superior a otra, su impacto en esta fuera mayor. Por desgracia esto no sucede en todos los casos. Para ilustrarlo veamos la figura 5.10.

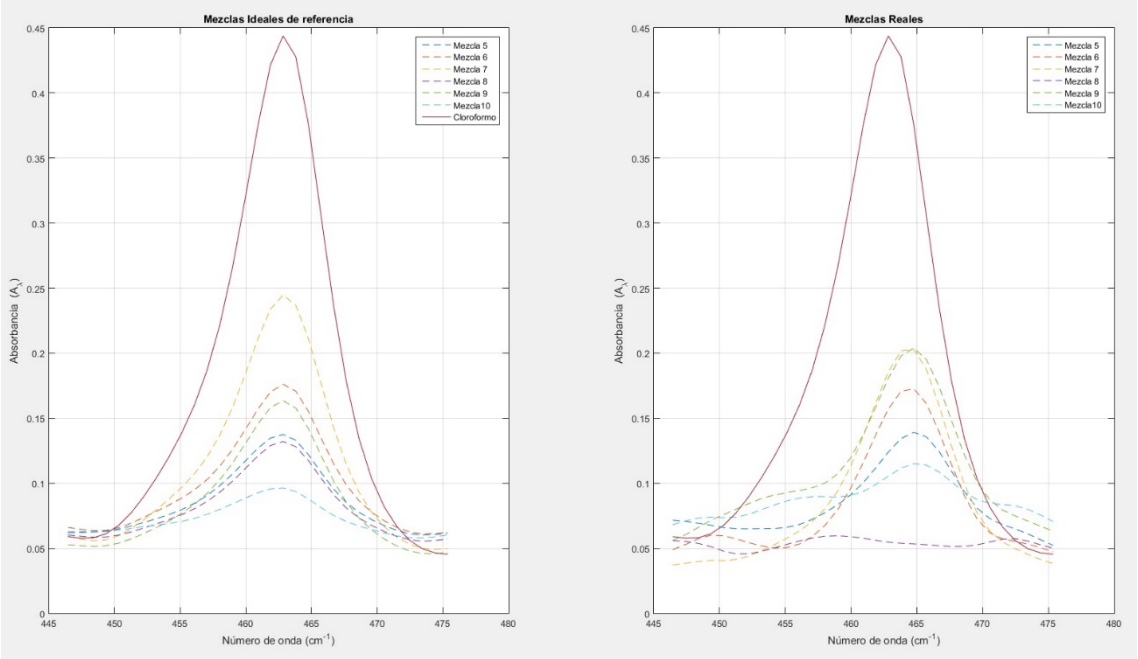



*Figura 5.10 Efecto del Tolueno sobre la banda 450-475 cm$^{-1}$*

La figura 5.10 muestra el efecto del tolueno sobre las mezclas en la banda de espectro comprendida entre los números de onda 450 y 475 cm$^{-1}$. Se muestran las mezclas de la 5 a la 10 ya que sobre el resto no debería afectar ya que su concentración teórica es 0%. Se observa en esta figura como la mezcla número 7 que teóricamente tiene un 50% de concentración de Tolueno, no solo que no llega al nivel de concentración deseado, sino que

se iguala al de la mezcla número 6 que posee un 30% de concentración. Otro tanto sucede con la mezcla número 8 que debe tener una concentración de un 20% y no llega siquiera al nivel de la mezcla 10 que tiene un 10%. Estas incongruencias en los niveles de concentración pueden achacarse a la no homogeneidad de las mezclas, siendo medidas muestras de estas que no contenían las concentraciones informadas.

## 5.4 Aplicación de un algoritmo más robusto.

Hemos visto en el punto anterior como las distorsiones que se generan en las mezclas se acentúan en los máximos más energéticos de estas. Estas distorsiones, sobre todo en el caso del desplazamiento del espectro, son difíciles de corregir debido a su comportamiento aleatorio. Pero si los componentes que generan estos máximos poseen características morfológicas capaces de diferenciarlos sin esta parte de su espectro, será posible una separación mejor eliminando estos máximos de las mezclas, para posteriormente reconstruirlos mediante su matriz de mezcla. En este caso particular, el Cloroformo posee dos máximos, uno altamente energético entre la banda de números de onda 705-765 $cm^{-1}$, y otro muy diferenciable pero menos energético en la banda 1200-1250 $cm^{-1}$. Por lo que si eliminamos del espectro de las mezclas la banda del primer máximo se puede esperar una mejora en la reconstrucción.

Por otra parte, como se vio en el capítulo 4, el algoritmo NMF genera separaciones con una calidad que, si bien no es muy alta, sí que es muy estable en cualquier condición. Se trata por tanto de un algoritmo altamente robusto que puede mejorar la separación en condiciones como las que se plantean en las mezclas reales. Además, añade la ventaja de su eficiencia temporal, lo que permite una ejecución múltiple generando reconstrucciones medias, lo que redunda en la atenuación de los errores.

La figura 5.11 ilustra la separación obtenida mediante el algoritmo NMF sobre las mezclas reales, eliminando de estas la banda correspondiente a los números de onda 705-765 $cm^{-1}$. La reconstrucción se lleva a cabo por medio de los valores medios de las matrices de mezcla separadas durante 30 iteraciones. Los resultados de esta separación pueden reproducirse mediante la ejecución del script de Matlab *'Separación_2'* que se adjunta a esta memoria.

Si bien es cierto que, en la mayoría de los casos, mediante el uso de un algoritmo como en NMF no se puede conseguir la calidad de separación del algoritmo $nGMCA^{Standard}$, en escenarios con datos que presentan comportamientos no lineales, el empleo de un algoritmo más robusto genera mejores resultados. La separación que muestra la figura 5.11 no es una separación óptima, pero dadas las condiciones de los datos se puede considerar suficiente.

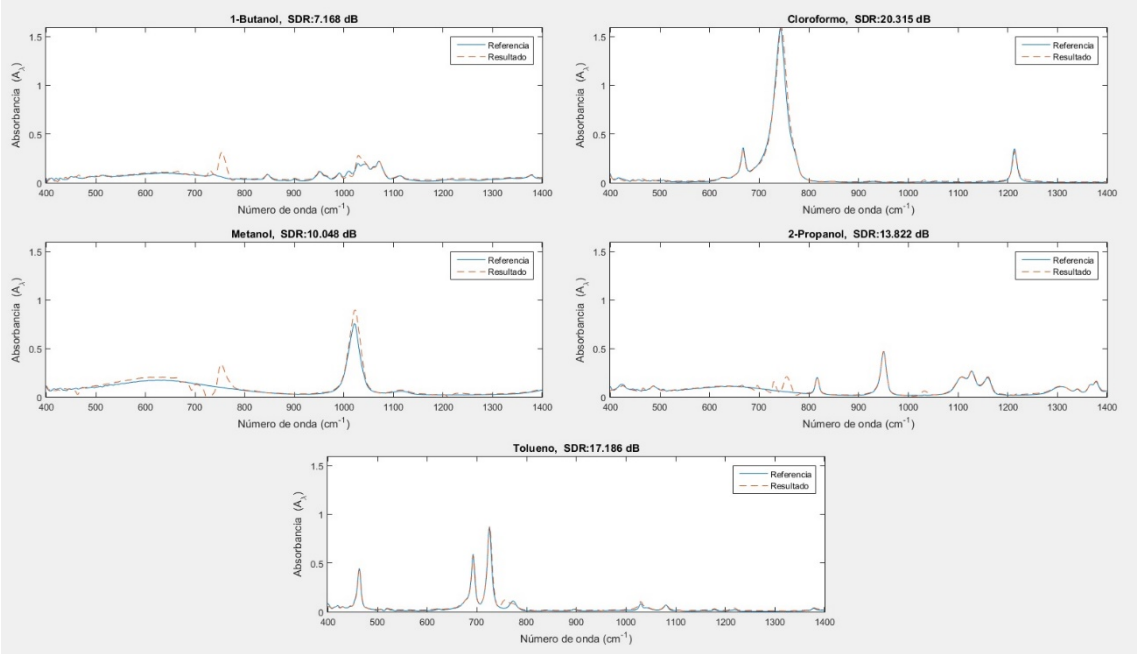



*Figura 5.11 Separación mediante NMF, valores medios sobre 30 iteraciones eliminando la banda 705-765 $cm^{-1}$*

# Capítulo 6

# Conclusiones y Trabajos futuros

## 6.1 Conclusiones.

El objetivo de este trabajo consistía en encontrar el método más adecuado para identificar los componentes de una mezcla y su concentración en ella. Para ello, a lo largo de esta memoria, se han desarrollado una serie de pruebas de las que podemos extraer las siguientes conclusiones.

- La naturaleza de los componentes a separar influye en la elección del método de separación. Hemos visto como debido a las características de los componentes y la naturaleza de sus espectros, unos métodos de separación se ajustan más que otros a la separación de mezclas compuestas por estos. Por una parte, la no negatividad de los espectros de componentes químicos descarta métodos que no discriminan valores negativos en las matrices de mezclas, como son el caso de los métodos SCA y GMCA. Por otro lado, la carencia de información previa sobre la independencia estadística de los espectros de unos componentes con respecto a otros descarta métodos basados en dicha independencia para realizar la separación de componentes, como por ejemplo los métodos ICA. Son, por lo tanto, los métodos basados en la factorización de matrices no negativas NMF y los derivados de estos, como los nGMCA, los que mejor se ajustan a la identificación de componentes en el escenario estudiado en este trabajo.

- La introducción de restricciones en los algoritmos de separación puede mejorar esta. Acotando los valores aceptados de las matrices de mezcla y componentes recuperados por los algoritmos de separación se obtienen mejores resultados a la hora de identificar componentes. Se ha comprobado como la introducción de la restricción de no negatividad de las matrices de mezcla mejora la recuperación de los espectros de los componentes, al igual que sucede con la introducción de la dispersión como restricción a los valores de los espectros de componentes recuperados.

- Una adecuada representación de las mezclas mejora la separación de componentes. Debido a las restricciones impuestas a los algoritmos con respecto a las características de los espectros recuperados, una representación de las mezclas en consonancia con dichas restricciones mejora la identificación final de los componentes. En el caso que nos ocupa, la representación de los espectros de las mezclas utilizando como unidad de medida la Absorbancia beneficia la separación, ya que introduce dispersión en los espectros, siendo esta una de las restricciones principales de los algoritmos estudiados.

- La cantidad de componentes que integran una mezcla influye en la identificabilidad de estos. Cuanto mayor es el número de componentes que integran las mezclas evaluadas, peor es la calidad en la separación de su espectro y, por lo tanto, la identificabilidad de este y de su concentración en las mezclas.

- Los algoritmos estudiados toleran ciertos niveles de ruido en los espectros de las mezclas. Se ha comprobado como niveles moderados de ruido en las mezclas no afectan de manera notable a la calidad de la separación. Los niveles de ruido tolerados por los algoritmos vienen determinados por la naturaleza morfológica de los espectros de las mezclas.

- La eficiencia temporal de los algoritmos estudiados depende en gran medida de la naturaleza de los componentes que integran las mezclas. Se ha comprobado como el tiempo requerido para la separación de los espectros de los componentes se incrementa de manera notable cuanto mayor sea el número y el tamaño de los componentes integrantes de las mezclas. El tamaño de los componentes apenas influye de manera positiva en los resultados de la separación, por lo tanto, se debe mantener el tamaño de las mezclas lo más reducido posible, siempre que no se pierda información, para mejorar los tiempos de separación.

- Existe una proporción óptima entre el número de componentes a separar y el número de mezclas a evaluar. Esta proporción no varía, en principio, con la complejidad morfológica de las mezclas y su efecto está asociado a la dispersión de estas. Por otra parte, los efectos negativos de la variación de esta proporción pueden paliarse mediante el procesado de los resultados de la separación, ya que estos son consecuencia de errores de escala.

- La utilización de una parte del espectro FTIR de las mezclas para realizar la separación puede mejorar esta. Se ha comprobado, en este caso particular, como la utilización de la zona del espectro llamada "zona de huella dactilar" conlleva una ligera mejora en la calidad de la separación.

- En mezclas con comportamientos no lineales, la eliminación de las zonas más energéticas de los espectros puede mejorar la separación de los componentes. Esto dependerá en gran medida de las características de los componentes, ya que han de tener una morfología diferenciable prescindiendo de esa zona del espectro.

- El algoritmo de separación que mejor se adapta al escenario de la identificación de componentes en mezclas es el nGMCA$^{\text{Standard}}$. De entre todos los algoritmos estudiados, este es el que mejores resultados ofrece en todos los escenarios en los que se han realizado pruebas, siempre que el comportamiento de las mezclas sea lineal.

## 6.2 Trabajos futuros.

La identificación automática de componentes en mezclas es un objetivo mucho mayor del alcanzado en este trabajo. Es por esto por lo que, para conseguir un método óptimo de identificación automática de componentes, se hace necesaria la realización de otra serie de estudios que ahonden en ciertos aspectos sobre los que esta memoria ha pasado de largo y que son de gran interés. A continuación, se exponen algunos de los que pueden arrojar mayores beneficios a la consecución del método óptimo de identificación.

- Una característica que no poseen los algoritmos estudiados en esta memoria es un método para reconocer de manera autónoma el número exacto de componentes que integran una mezcla, es necesario indicarlo en forma de parámetro de entrada para que estos ejecuten una separación efectiva. La obtención de este método introduciría una enorme mejora en la automatización de la identificación de los componentes. En [Stark et alt., 2010] se detalla un algoritmo que permite este objetivo mediante el método GMCA, aunque el autor de este trabajo no haya conseguido adaptarlo al método nGMCA, parece un buen punto de partida.

- Otro punto de gran interés es el de la representación de las mezclas. Como se ha comprobado en este trabajo, una adecuada representación de las mezclas conduce a una mejora sustancial en la calidad de las separaciones. La búsqueda de mejores representaciones o representaciones más dispersas para las mezclas podría conducir a una representación óptima que permitiera una separación de gran calidad en cualquier circunstancia. La representación de una mezcla implica dos aspectos, por una parte, la elección de un diccionario de representación, y por otra, la selección de un algoritmo para calcular dicha representación [Comon and Jutten, 2010]. Con respecto a la elección del diccionario, el autor de este trabajo ha experimentado con diversos diccionarios predefinidos sin obtener una mejora en la representación. Sin embargo, el aprendizaje automático de diccionarios es un campo de estudio que abre un amplio abanico de oportunidades para introducir mejoras en la representación de las mezclas, algoritmos como ILS-DLA/MOD, K-SVD, ODL o RLS-DLA, son buenos candidatos de estudio. En lo que respecta al algoritmo de representación, se recogen varios en la literatura, M-FOCUSS, Basis Pursuit, Matching Pursuit, etc. Un estudio detallado de estos puede descubrirnos el que mejor se adapta al escenario de la separación de componentes en mezclas.

- A lo largo de este trabajo se ha comprobado como la introducción de nuevas restricciones en los algoritmos de separación conduce a una mejor calidad en la recuperación de los componentes. El hecho de que las matrices de mezcla recuperadas por los algoritmos de este trabajo no tengan en cuenta que la suma de las concentraciones de una mezcla ha de ser el 100% de esta, supone un hándicap en su aplicación al escenario que plantea este trabajo. La modificación de los algoritmos más prometedores para que admitan esta restricción mejoraría en gran medida el rendimiento de estos y evitaría la necesidad de un procesado posterior de la matriz de mezcla, mejorando, de este modo, la eficiencia temporal de los mismos.

- Por último, uno de los métodos más prometedores para la identificación de los componentes era, a priori, el método nGMCA$^{Convolutivo}$, ya que nos permite manejar las restricciones de dispersión y no negatividad en el mismo dominio transformado. En opinión del autor de este trabajo, el bajo rendimiento de este algoritmo se debe a una parametrización poco precisa. El estudio de una parametrización adecuada para este algoritmo podría conducir a un método más potente para la separación de componentes.

# Bibliografía

[Aoulass and Chakkour, 2020] Aoulass, N., & Chakkour, O. (2020). Non-Negative Matrix Factorization for Blind Source Separation. In Handbook of Research on Recent Developments in Electrical and Mechanical Engineering (pp. 259-282). IGI Global.

[Balan and Rosca, 2000] Balan, R., & Rosca, J. (2000). Statistical properties of STFT ratios for two channel systems and applications to blind source separation. *ratio*, *1*(X2), X2.

[Bobin et alt., 2006] Bobin, J., Moudden, Y., & Starck, J. L. (2006, May). Enhanced source separation by morphological component analysis. In *2006 IEEE International Conference on Acoustics Speech and Signal Processing Proceedings* (Vol. 5, pp. V-V). IEEE.

[Bobin et alt., 2007] Bobin, J., Starck, J. L., Fadili, J., & Moudden, Y. (2007). Sparsity and morphological diversity in blind source separation. *IEEE Transactions on Image Processing*, *16*(11), 2662-2674.

[Bofill and Zibulevsky, 2001] Bofill, P., & Zibulevsky, M. (2001). Underdetermined blind source separation using sparse representations. *Signal processing*, *81*(11), 2353-2362.

[Cardoso and Souloumiac, 1993] Cardoso, J. F., & Souloumiac, A. (1993, December). Blind beamforming for non-Gaussian signals. In *IEE proceedings F (radar and signal processing)* (Vol. 140, No. 6, pp. 362-370). IET Digital Library.

[Cherry, 1953] Cherry, E. C. (1953). Some experiments on the recognition of speech, with one and with two ears. *The Journal of the acoustical society of America*, *25*(5), 975-979.

[Cichocki et alt., 2009] Cichocki, A., Zdunek, R., Phan, A. H., & Amari, S. I. (2009). *Nonnegative matrix and tensor factorizations: applications to exploratory multi-way data analysis and blind source separation*. John Wiley & Sons.

[Comon and Jutten, 2010] Comon, P., & Jutten, C. (Eds.). (2010). *Handbook of Blind Source Separation: Independent component analysis and applications*. Academic press.

[Haykin and Chen, 2005] Haykin, S., & Chen, Z. (2005). The cocktail party problem. *Neural computation*, *17*(9), 1875-1902.

[Hoyer, 2002] Hoyer, P. O. (2002, September). Non-negative sparse coding. In *Proceedings of the 12th IEEE Workshop on Neural Networks for Signal Processing* (pp. 557-565). IEEE.

[Hoyer, 2004] Hoyer, P. O. (2004). Non-negative matrix factorization with sparseness constraints. *Journal of machine learning research*, *5*(Nov), 1457-1469.

[Hyvärinen and Oja, 2000] Hyvärinen, A., & Oja, E. (2000). Independent component analysis: algorithms and applications. *Neural networks*, *13*(4-5), 411-430.

[Hyvärinen et al., 2001] Hyvarinen, A., Karhunen, J., & Oja, E. (2001). Independent component analysis and blind source separation.

[Kim and Park, 2008] Kim, J., & Park, H. (2008, December). Toward faster nonnegative matrix factorization: A new algorithm and comparisons. In *2008 Eighth IEEE International Conference on Data Mining* (pp. 353-362). IEEE.

[Kimura et alt., 2015] Kimura, K., Tanaka, Y., & Kudo, M. (2015, February). A fast hierarchical alternating least squares algorithm for orthogonal nonnegative matrix factorization. In *Asian Conference on Machine Learning* (pp. 129-141).

[Lee and Seung, 1999] Lee, D. D., & Seung, H. S. (1999). Learning the parts of objects by non-negative matrix factorization. *Nature*, *401*(6755), 788-791.

[Lee and Seung, 2001] Lee, D. D., & Seung, H. S. (2001). Algorithms for non-negative matrix factorization. In *Advances in neural information processing systems* (pp. 556-562).

[Lee et alt., 2000] Lee, T. W., Girolami, M., Bell, A. J., & Sejnowski, T. J. (2000). A unifying information-theoretic framework for independent component analysis. *Computers & Mathematics with Applications*, *39*(11), 1-21.

[Mondragón, 2017] Mondragón, P (2017). Espectroscopia de infrarrojo. *Centro de Investigación y Asistencia en Tecnología y Diseño del Estado de Jalisco, A.C.*

[Mur et alt., 2017] Mur, A., Dormido, R., Duro, N., & Mercader, D. (2017). An unsupervised method to determine the optimal number of independent components. *Expert Systems with Applications*, *75*, 56-62.

[Oja and Plumbley, 2004] Oja, E., & Plumbley, M. (2004). Blind separation of positive sources by globally convergent gradient search. *Neural Computation*, *16*(9), 1811-1825.

[Paatero and Tapper, 1994] Paatero, P., & Tapper, U. (1994). Positive matrix factorization: A non-negative factor model with optimal utilization of error estimates of data values. *Environmetrics*, *5*(2), 111-126.

[Pérez-Alonso et alt., 2006] Pérez-Alonso, M., Castro, K., & Madariaga, J. M. (2006). Vibrational spectroscopic techniques for the analysis of artefacts with historical, artistic and archaeological value. Current Analytical Chemistry, 2(1), 89-100.

[Plumbley, 2003] Plumbley, M. D. (2003). Algorithms for nonnegative independent component analysis. *IEEE Transactions on Neural Networks*, *14*(3), 534-543.

[Rakesh and Charmi, 2014] Rakesh, P., & Charmi, P. (2014). Quantitative analytical applications of FTIR spectroscopy in pharmaceutical and allied areas. Journal of Advanced Pharmacy Education & Research Apr-Jun, 4(2).

[Rapin et alt., 2013] Rapin, J., Bobin, J., Larue, A., & Starck, J. L. (2013). Sparse and non-negative BSS for noisy data. *IEEE Transactions on Signal Processing*, *61*(22), 5620-5632.

[Rapin et alt., 2014] Rapin, J., Bobin, J., Larue, A., & Starck, J. L. (2014). NMF with sparse regularizations in transformed domains. *SIAM journal on Imaging Sciences*, *7*(4), 2020-2047.

[Sánchez, 2003] Sánchez, M. L. (2013). La espectroscopia vibracional como herramienta analítica: aplicaciones en áreas de interés científico, agrícola e industrial (Doctoral dissertation, Universidad de Jaén).

[Stark et alt., 2010] Starck, J. L., Murtagh, F., & Fadili, J. M. (2010). *Sparse image and signal processing: wavelets, curvelets, morphological diversity*. Cambridge university press.

[Vavasis, 2009] Vavasis, S. A. (2010). On the complexity of nonnegative matrix factorization. *SIAM Journal on Optimization*, *20*(3), 1364-1377.

[Vielva et alt., 2001] Vielva, L., Erdogmus, D., & Principe, J. C. (2001, December). Underdetermined blind source separation using a probabilistic source sparsity model. In *Proc. ICA* (Vol. 2001, pp. 675-679).

[White, 1990] White, R. (1989). *Chromatography/Fourier transform infrared spectroscopy and its applications* (Vol. 10). CRC press.

[Yilmaz and Rickard, 2004] Yilmaz, O., & Rickard, S. (2004). Blind separation of speech mixtures via time-frequency masking. *IEEE Transactions on signal processing*, *52*(7), 1830-1847.

[Yuanqing et alt., 2003] Li, Y., Cichocki, A., & Amari, S. I. (2003, April). Sparse component analysis for blind source separation with less sensors than sources. In *Ica* (Vol. 2003, pp. 89-94).

[Zdunek and CiChoki, 2007] Zdunek, R., & Cichocki, A. (2007, November). Blind image separation using nonnegative matrix factorization with Gibbs smoothing. In *International Conference on Neural Information Processing* (pp. 519-528). Springer, Berlin, Heidelberg.

# Listado de siglas, abreviaturas y acrónimos

**FTIR.** (Fourier Transform Infrared) Transformada de Fourier infrarroja.

**IR**. (Infrared) Infrarojo.

**CPP.** (Cocktail Party Problem) Problema de la fiesta de Cocktail.

**BSS.** (Blind Source Separation) Separación ciega de fuentes.

**NMF**. (Non-Negative Matrix Factorization) Factorización de matrices no negativas.

**nGMCA**. (Non-Negative Generalized Morphological Component Analysis) Análisis generalizado de componentes morfológicos no negativos.

**ICA** (Independent Component Analysis) Análisis de componentes independientes.

**SCA**. (Sparse Component Analysis) Análisis de componentes dispersos.

**GMCA**. (Generalized Morphological Component Analysis) Análisis generalizado de componentes morfológicos

**JADE**. (Joint Approximate Diagonalization of Eigenmatrices) Diagonalización conjunta aproximada de matrices propias

**DUET**. (Degenerate Unmixing Estimation Technique) Técnica de estimación de separación de mezclas degeneradas.

**SDBS**. (Spectral Database for Organic Compounds) Base de datos de espectros de compuestos orgánicos.

**AIST**. (National Institute of Advanced Industrial Science and Technology) Instituto Nacional de Ciencia y Tecnología Industrial Avanzada.

**SNR**. (Signal to Noise Ratio) Relación señal ruido.

**SDR**. (Source Distortion Ratio) Relación fuente distorsión.

# Anexo I

# Software

Con el fin de poder replicar los resultados del análisis presentado, este trabajo se acompaña de un conjunto de software. Este conjunto agrupa funciones y scripts desarrolladas mediante el uso del lenguaje de programación Matlab, concretamente la versión R2015a de 64 bits.

Todo el software viene empaquetado en un archivo comprimido en formato RAR que contiene 4 grupos de archivos: Scripts, Resultados, Funciones y Datos. A continuación, se detallan el contenido de estos grupos, así como sus funciones principales.

## I. Scripts.

El trabajo viene acompañado de 15 scripts que reproducen cada uno de los test realizados para completar el análisis de este trabajo, estos son:

- *Test_A.m*. Presenta el resultado de la separación de mezclas compuestas por dos componentes (Taurina y Ácido Fólico) mediante algoritmos diferentes para ilustrar su comportamiento con respecto a la restricción de no-negatividad en sus resultados.

- *Test_B.m*. Muestra por consola la Matriz de distancia de correlación entre los componentes de un conjunto y genera la gráfica de los componentes, de manera individual y en conjunto.

- *Test1_I.m*. Evalúa la calidad de reconstrucción de los algoritmos bajo estudio con respecto al número de componentes de las mezclas. Se emplea el denominado Conjunto_I de componentes como conjunto de pruebas.

- *Test2_I.m*. Evalúa la calidad de reconstrucción de los algoritmos bajo estudio con respecto al nivel de ruido de las mezclas. Se emplea el denominado Conjunto_I de componentes como conjunto de pruebas.

- *Test3_I.m*. Evalúa la calidad de reconstrucción de los algoritmos bajo estudio con respecto al número de mezclas del conjunto. Se emplea el denominado Conjunto_I de componentes como conjunto de pruebas.

- *Test4_I.m*. Evalúa la calidad de reconstrucción de los algoritmos bajo estudio con respecto al número de muestras de los componentes. Se emplea el denominado Conjunto_I de componentes como conjunto de pruebas.

- ***Test5_I.m***. Evalúa la calidad de reconstrucción de los algoritmos bajo estudio con respecto a la dispersión de los componentes. Se emplea el denominado Conjunto_I de componentes como conjunto de pruebas. Para modificar la dispersión de los componentes se modificará el coeficiente de activación de generación [0...1]. Este coeficiente representa el porcentaje de puntos que tienen valor distinto de 0 en el componente generado (a mayor coeficiente menor dispersión).

- ***Test1_II.m***. Evalúa la calidad de reconstrucción de los algoritmos bajo estudio con respecto al número de componentes de las mezclas. Se emplea el denominado Conjunto_II de componentes como conjunto de pruebas.

- ***Test2_II.m***. Evalúa la calidad de reconstrucción de los algoritmos bajo estudio con respecto al nivel de ruido de las mezclas. Se emplea el denominado Conjunto_II de componentes como conjunto de pruebas.

- ***Test3_II.m***. Evalúa la calidad de reconstrucción de los algoritmos bajo estudio con respecto al número de mezclas del conjunto. Se emplea el denominado Conjunto_II de componentes como conjunto de pruebas.

- ***Test4_II.m***. Evalúa la calidad de reconstrucción de los algoritmos bajo estudio con respecto al número de muestras de los componentes. Se emplea el denominado Conjunto_II de componentes como conjunto de pruebas.

- ***Test1_III.m***. Evalúa la calidad de reconstrucción de los algoritmos bajo estudio con respecto al número de muestras de las mezclas. Se emplea el denominado Conjunto_III de componentes como conjunto de pruebas.

- ***Test2_III.m***. Evalúa la calidad de reconstrucción de los algoritmos bajo estudio con respecto a la amplitud del espectro de los componentes. Se emplea el denominado Conjunto_III de componentes como conjunto de pruebas.

- ***Separación_1.m***. Realiza la separación de un grupo de 5 componentes sobre un conjunto de 15 mezclas. La separación se lleva a cabo mediante la ejecución de un número definido de iteraciones (*N_Iter*) del algoritmo nGMCA_Standard. Se puede realizar la separación sobre 2 conjunto de mezclas diferentes. Modificando la generación del conjunto *Set.Y* la sección " Crear Conjunto de datos" se puede seleccionar entre un conjunto de mezclas reales o sus equivalentes teóricas. El test muestra gráficamente el resultado de la separación indicando la calidad de esta.

- ***Separación_2.m***. Realiza la separación de un grupo de 5 componentes sobre un conjunto de 15 mezclas reales. La separación se lleva a cabo mediante la ejecución de un número definido de iteraciones (*N_Iter*) del algoritmo NMF. Se realiza la separación sin tener en cuenta la banda del espectro de las mezclas correspondientes a los números de onda 705-765 $cm^{-1}$. El test muestra gráficamente el resultado de la separación indicando la calidad de esta.

# II. Resultados.

Debido a que la mayoría de los test necesarios para realizar el análisis de este trabajo requieren un número elevado de iteraciones, el tiempo necesario para que muestren resultados es elevado. Es por esto que se acompaña esta memoria con los datos obtenidos de la ejecución de estos test en forma de archivo de datos en formato Matlab. Concretamente se suministra un conjunto de resultados (*.*mat)* para cada uno de los test que evalúan los diferentes conjuntos de datos (Conjuntos I, II y III), lo que hace un total de 11 archivos. Cada uno de estos archivos contiene 5 estructuras de datos, 4 de ellas son comunes a todos ellos y una de ellas es específica a cada tipo de test. Las estructuras comunes son las siguientes:

- ***SDR_S***. Consiste en una matriz de dimensión 5xN que recoge los resultados medios de calidad de reconstrucción de los componentes de cada uno de los algoritmos bajo evaluación, en cada uno de los escenarios de evaluación. Cada fila corresponde a los resultados de un algoritmo, siendo la fila 1 la correspondiente al algoritmo NMF, la fila 2 al $nGMCA^{Standard}$, la fila 3 al $nGMCA^{Análisis}$, la fila 4 al $nGMCA^{Síntesis}$ y la fila 5 al $nGMCA^{Convolutivo}$. Cana una de las N columnas corresponde a los resultados de cada uno de los escenarios de evaluación. Por ejemplo, si se evalúan los algoritmos respecto al número de componentes de las mezclas y se proponen 5 escenarios con 5 cantidades de mezclas diferentes, la matriz SDR_S tendrá 5 columnas.

- ***SDR_A***. Consiste en una matriz de dimensiones 5xN que recoge los resultados medios de calidad de reconstrucción de la matriz de mezcla de cada uno de los algoritmos bajo evaluación, en cada uno de los escenarios de evaluación. Su contenido, al igual que en caso anterior, asigna cada fila a un algoritmo y cada columna a un escenario de evaluación.

- ***Tiempos***. Esta matriz de dimensión 5xN contiene el tiempo empleado por cada algoritmo en la ejecución de separación de cada uno de los escenarios. Su estructura se asemeja a las matrices anteriores. Asigna cada fila a un algoritmo y cada columna a un escenario de test.

- ***Dispersión***. Se trata de un vector de longitud N que contiene los valores medios de dispersión (medido en el índice Hoyer) de cada conjunto de componentes empleado en cada uno de los escenarios de test planteados. Cada uno de sus N valores corresponde a los escenarios de cada una de las N columnas de las matrices anteriores.

Como ya hemos dicho, además de las 4 estructuras anteriores, cada uno de los archivos de resultados contiene una estructura específica que depende del tipo de test al que pertenezca. Estas estructuras son las siguientes:

- ***N_Fuentes***. Se trata de un vector de longitud N que contiene el número de fuentes con que se evaluará cada escenario. Por ejemplo, en el Test1_I el vector N_Funetes contiene los valores (4, 8, 16, 24 y 32) que corresponderán a 5 escenarios de tert diferentes en los que las mezclas generadas se compondrán de este número de componentes.

- ***Ruido***. Se trata de un vector de longitud N que contiene los valores del nivel ruido que se añadirá a las mezclas en cada escenario de evaluación. Los niveles de ruido deben indicarse en *-dB* y el valor *Inf* corresponde a un escenario sin ruido añadido. Por ejemplo, en el Test2_I el vector Ruido contiene los valores (10, 20, 30, 40, Inf) lo que corresponde a escenarios con niveles de ruido de -10dB, -20dB, -30dB, -40dB y un último escenario de evaluación carente de ruido añadido.

- ***N_Mezclas***. Se trata de un vector de longitud N que contiene el número de mezclas con que se evaluará cada escenario.

- ***N_Muestras***. Se trata de un vector de longitud N que contiene el número de muestras que componen cada componente y mezcla con que se evaluará cada escenario.

- ***Act***. Es un vector de longitud N que contiene el coeficiente de activación que se usará para generar cada componente del Conjunto_I de datos. El coeficiente de activación pude tener valores en el rango [0…1], y representa el porcentaje de muestras de cada componente que tendrán un valor distinto de 0. Mediante la modificación de este coeficiente se puede manipular la dispersión de los componentes generados sintéticamente. Este vector solo se encuentra en el Test5_I, ya que es el Conjunto_I de datos el único que permite manipular esta característica.

# III. Funciones.

Los algoritmos testados en este trabajo han sido desarrollados por diferentes autores, pero para adaptar su funcionamiento a los conjuntos de prueba de este trabajo ha sido necesario el desarrollo de un conjunto de funciones accesorias. Estas funciones permiten llevar a cabo los diferentes escenarios de test recogidos en los scripts anteriores. A continuación, se enumeran y detallan cada una de las funciones desarrolladas en Matlab para este trabajo:

- ***Evaluar_Escenario_I.m***. Esta función ejecuta la separación de componentes mediante los algoritmos objeto de test con el fin de evaluar su comportamiento sobre el Conjunto_I de datos. Se ejecuta la separación un número indicado de veces empleando siempre el mismo conjunto de datos para todos los algoritmos en cada iteración.

**Parámetros de entrada:**

- *Componentes*: Número de componentes que tendrá el conjunto de prueba.
- *Mezclas*: Número de mezclas que tendrá el conjunto de prueba.
- *Muestras*: Longitud de los componentes del conjunto.
- *Ruido*: Nivel de ruido en -dB [0...Inf] que se añadirá a las mezclas.
- *Actcoef*: Coeficiente [0...1] de activación de los componentes, indica el porcentaje de puntos de muestra en los que cada componente será distinto de 0, regulando su dispersión.
- *Iter*: Número de iteraciones de pruebas que se ejecutarán.

**Salida:** La salida de esta función es un conjunto de datos formado por las 4 estructuras comunes detalladas en el apartado anterior de "Resultados". Consta de 3 matrices de dimensiones 5xN llamadas SDR_S, SDR_A y Tiempos, además de un vector de longitud N llamado Dispersión

- ***Evaluar_Escenario_II.m***. Esta función ejecuta la separación de componentes mediante los algoritmos objeto de test con el fin de evaluar su comportamiento sobre el Conjunto_II de datos. Se ejecuta la separación un número indicado de veces empleando siempre el mismo conjunto de datos para todos los algoritmos en cada iteración.

  **Parámetros de entrada:**

  - *Componentes*: Número de componentes que tendrá el conjunto de prueba.
  - *Mezclas*: Número de mezclas que tendrá el conjunto de prueba.
  - *Muestras*: Longitud de los componentes del conjunto.
  - *Ruido*: Nivel de ruido en -dB [0...Inf] que se añadirá a las mezclas.
  - *Iter*: Número de iteraciones de pruebas que se ejecutarán.

  **Salida:** La salida, al igual que el caso anterior, es un conjunto de datos formado por 3 matrices de dimensiones 5xN llamadas SDR_S, SDR_A y Tiempos, además de un vector de longitud N llamado Dispersión

- ***Evaluar_Escenario_III.m***. Esta función ejecuta la separación de componentes mediante los algoritmos objeto de test con el fin de evaluar su comportamiento sobre el Conjunto_III de datos. Se ejecuta la separación un número indicado de veces empleando siempre el mismo conjunto de datos para todos los algoritmos en cada iteración.

  **Parámetros de entrada:**

  - *Muestras*: Longitud de los componentes del conjunto.
  - *Iter*: Número de iteraciones de pruebas que se ejecutarán.

- *Espectro*: Selecciona que parte del espectro de los componentes se empleará en el test. Epectro íntegro (0), o espectro de zona de huella dactilar (1).

**Salida:** La salida, al igual que en las anteriores funciones de esta serie, es un conjunto de datos formado por 3 matrices de dimensiones 5xN llamadas SDR_S, SDR_A y Tiempos, además de un vector de longitud N llamado Dispersión.

- **CompararComponentes.m**. Compara de forma gráfica los componentes de dos conjuntos. Muestra la superposición de sus espectros, así como la calidad de reconstrucción de uno con respecto a otro.
  **Parámetros de entrada:**
  - *Set*: Registro de datos que contiene, al menos, un campo S, siendo S una matriz sxn que representa a un conjunto de s componentes ordenados que se tomarán como referencia para la comparación.
  - *Comp*: Registro de datos que contiene, al menos, un campo S, siendo S una matriz sxn que representa a un conjunto de s componentes a comparar con el conjunto original según el orden de referencia.

- **Generar_A.m**. Genera una matriz aleatoria MxN que representa la concentración de cada N componente en cada M mezcla. Se emplean valores de concentración [0...1] donde cada mezcla M tendrá una concentración total $\sum_{i=1}^{N} M_i = 1$.
  **Parámetros de entrada:**
  - *Mezclas*: Número de mezclas que tendrá la matriz (Filas).
  - *Componentes*: Número de componentes que tendrá la matriz (Columnas).

  **Salida:**
  - *A:* Matriz de mezcla MxN con los índices de concentración de cada componente en cada mezcla.

- **MatrizDC.m**. Calcula la distancia de correlación entre un conjunto de componentes.
  **Parámetros de entrada:**
  - *S*: Matriz NxM que contiene N componentes de M puntos de medida cada uno.

  **Salida:**
  - *M*: Matriz de distancia de correlación de cada uno de los componentes con el resto.

- **OrdenaCM.m**. Ordena un conjunto de componentes y su matriz de mezcla según el orden de referencia de otro conjunto de componentes.
  **Parámetros de entrada:**
  - *O*: Registro de datos que contiene, al menos, un campo S, siendo S una matriz de dimensión sxn que representa a un conjunto de s componentes ordenados que se tomarán como referencia para la ordenación.

- *R*: Registro de datos que contiene, al menos, un campo S y un campo A, siendo S una matriz de dimensión sxn que representa a un conjunto de s componentes a ordenar según el orden de referencia, y siendo A una matriz de dimensión mxs que representa a un conjunto de m mezclas a ordenar según el orden de referencia.

**Salida:**

- *CompR*: Matriz de componentes de dimensión sxn ordenada según referencia.
- *MezcA*: Matriz de mezcla de dimensión mxs ordenada según referencia.

- **PAA.m**. Reduce la dimensionalidad de un vector sin pérdida de información aparente mediante el método PAA. La representación PAA (Piecewise Aggregate Aproximation) es una técnica de reducción de la dimensionalidad consistente en fraccionar una serie de valores en un número de segmentos de longitud idéntica que tendrán como valor el valor medio de los puntos de la serie en ese segmento.

  **Parámetros de entrada:**

  - *Vector*: Vector original con 1xM número de muestras.
  - *N*: Nuevo número de muestras al que se quiere reducir el vector.

  **Salida:**

  - *V_PAA*: Vector reducido 1xN.

- **ProcesaM.m**. Procesa una matriz de mezcla NxM recuperada de una separación ciega para que cumpla la condición $\sum_{i=1}^{N} M_i = 1$.

  **Parámetros de entrada:**

  - *MM*: Matriz NxM de mezcla a procesar.
  - *MINIMO*: Factor de redondeo de mezcla nula.

  **Salida:**

  - *MP*: Matiz NxM procesada.

- **SRD.m**. Cuantifica la calidad de reconstrucción de un componente según su índice SDR (Source Distortion Ratio).

  **Parámetros de entrada:**

  - *O*: Vector componente referencia.
  - *R*: Vector componente reconstruido.

  **Salida:**

  - *SDR*: Índice de calidad de reconstrucción.

- **Sparseness.m** Calcula el índice Hoyer de dispersión de un componente o un conjunto de ellos.

  **Parámetros de entrada:**

  - *V*: Vector 1xM que representa un componente con M muestras, o matriz NxM que representa un conjunto de N componentes con M Muestras.

  **Salida:**

  - *H*: Índice Hoyer de dispersión del componente V, o vector 1xN de índices Hoyer de dispersión de cada N componente.

Como se especifica al inicio de este punto, los algoritmos evaluados en este trabajo han sido desarrollados por diferentes autores, esto ha hecho necesario el uso de funciones ajenas para el desarrollo de las pruebas. Dichas funciones están publicadas y son de libre acceso, y las necesarias para el correcto funcionamiento de los test han sido empaquetadas en una carpeta llamada TOOLS junto con las funciones descritas anteriormente. Dentro de esta carpeta podemos encontrar una colección de funciones agrupadas por el nombre del algoritmo al que pertenecen. A continuación, se facilitan los enlaces de descarga a estas funciones:

- Funciones para la ejecución del algoritmo ICA/JADE.
- Funciones para la ejecución del algoritmo SCA.
- Funciones para la ejecución del algoritmo GMCA.
- Funciones para la ejecución del algoritmo NMF.
- Funciones para la ejecución del algoritmo nGMCA.

# IV. Datos.

Para la realización de las pruebas que permiten el análisis expuesto en este trabajo son necesarios diferentes conjuntos de datos. Estos conjuntos se suministran junto a esta memoria en una carpeta con nombre DATOS. Esta carpeta está compuesta por 5 archivos de datos en formato Matlab (*.mat) que contienen diferentes estructuras de datos, las cuales se detallan a continuación:

- ***Set_A.*** El archivo Set_A contiene el conjunto de datos empleado para la ejecución del script Test_A. Contiene una estructura de datos llamada *Set* de tipo struct que está compuesta por 3 componentes.

  - *S*: Matriz de dimensión 2x1024 que contiene la representación de dos componentes (Taurina y Ácido fólico) con 1024 muestras cada uno.

- *A*: Matriz de dimensión 4x2 que representa la matriz de mezcla de los componentes de S para generar Y.

- *Y*: Matriz de dimensión 4x1024 que contiene la representación de las 4 mezclas generadas mediante los componentes de S y la matriz de mezcla A.

- ***Set_B***. El archivo Set_B contiene el conjunto de datos empleado para la ejecución del script Test_B. Contiene 3 estructuras de datos que son las siguientes:

  - *S*: Matriz de dimensión 5x3735 que contiene la representación de cinco componentes (Butanol, Cloroformo, Metanol, Propanol y Tolueno) con 3735 muestras cada uno.

  - *Nombres*: Vector de dimensión 1x5 que contiene los nombres de los anteriores compuestos.

  - *X*: Vector de dimensión 1x3735 que contiene la representación del eje X del espectro de los componentes que va desde 400 a 4000 $cm^{-1}$.

- ***Conjunto_II***. Este archivo contiene el conjunto de datos empleado para la ejecución de los scripts Test1_II, Test2_II, Test3_II y Test4_II. Contiene una matriz de nombre "*Componentes*" y de dimensión 15x2048 que contiene la representación de 15 componentes con 2048 muestras cada uno. Estos componentes son los que constituyen el llamado Conjunto_II de datos y son por este orden, Mentol, Cafeína, Fenilalanina, Lactosa, Ácido Ascórbico, Ácido Cítrico, Hidroxibenzoato de Sodio, Alcohol Frutal, Ácido Fólico, Manitol, Colesterol, Sacarosa, Myo-inositol, Ácido Oleico y Glicerol.

- ***Conjunto_III***. Este archivo contiene el conjunto de datos empleado para la ejecución de los scripts Test1_III y Test2_III. Contiene 2 estructuras de datos que constituyen el llamado Conjunto_III, y son las siguientes:

  - *Componentes*: Matriz de dimensión 5x3736 que contiene la representación de cinco componentes (Butanol, Cloroformo, Metanol, Propanol y Tolueno) con 3736 muestras cada uno y que contiene su espectro en el rango 400-4000 $cm^{-1}$.

  - *Matriz_A*: Se trata de una matriz de dimensión 10x5 que contiene la matriz de mezcla A representada en (4.3).

- ***Conjunto_Real_Refinado***. Este archivo contiene el conjunto de datos empleado para la ejecución de los scripts Separación_1 y Separación_2. Contiene 5 estructuras de datos que son las siguientes:

  - *Componentes*: Matriz de dimensión 5x1039 que contiene la representación de cinco componentes (Butanol, Cloroformo, Metanol, Propanol y Tolueno) con 1039 muestras cada uno y que contiene su espectro en el rango 400-1400 $cm^{-1}$.

  - *Matriz_A*: Se trata de una matriz de dimensión 10x5 que contiene la matriz de mezcla A representada en (4.3).

- *Mezclas*: Matriz de dimensión 10x1039 que contiene la representación de 10 mezclas reales obtenidas mediante un espectrofotómetro compuestas por los 5 componentes recogidos en la matriz "Componentes". Representa el espectro de estas mezclas en el rango 400-1400 $cm^{-1}$.

- *Nombres*: Vector de dimensión 1x5 que contiene los nombres de los anteriores compuestos.

- *X*: Vector de dimensión 1x1039 que contiene la representación del eje X del espectro de los componentes que va desde 400 a 1400 $cm^{-1}$.