Capítulo 2

Representación esparsiva y diccionarios de aprendizaje.

2.1 Reconstrucción de señales

A finales del siglo XIX da surgimiento una teoría matemática que sería de gran uso en el procesamiento de señales. El matemático francés Joseph Fourier estableció que una señal podría ser representada como la suma de series de senos y cosenos, que hoy en día tal postulado se ha dado a conocer como la transformada de Fourier. La transformada de Fourier ha sido de gran utilidad en la resolución de problemas científicos e ingenieriles en diferentes campos, tales como física cuántica, óptica, electrónica y muchos otros [8].

Desde un punto de vista ingenieril la transformada de Fourier se describe como un fenómeno físico más que una herramienta matemática. Las señales pueden ser interpretadas como una combinación lineal de ondas harmónicas por lo que se observa que la señal en un instante de tiempo es reemplazada por la suma de varios armónicos.

Otra herramienta útil para el análisis de señales es la transformada Wavelet. Inicialmente un geofísico francés llamado Jean Morlet trabajaba sobre un método para modelar la propagación del sonido a través de la corteza terrestre. Morlet utilizó un sistema basado en una función prototipo, que cumpliendo ciertos requerimientos matemáticos y mediante dos procesos denominados dilatación y traslación se formaban un set de bases que representaban las señales de propagación con la misma robustez y versatilidad que la transformada de Fourier [8].

Las características propias de la transformada Wavelet nos otorgan la posibilidad de representar señales en diferentes niveles de resolución, analizar señales no estacionarias permitiéndonos saber el contenido en frecuencia de una señal [8].

2.2 Representación esparsiva.

La representación esparsiva de señales ha sido de gran interés de estudio en los años recientes. El problema a resolver de la representación esparsiva está en la búsqueda de compactar una señal en términos de una combinación lineal de átomos en un "diccionario sobre-completo". Comparando los métodos basados en transformaciones orto-normales o procesamiento en el dominio del tiempo, la representación esparsiva ofrece un mejor rendimiento en el modelado de señales [9].

Las señales discretas x son vectores en *N-dimensiones* denotados en un espacio euclidiano por R^N. El conjunto de señales atómicas $\boldsymbol{D} = \{\boldsymbol{d_i}\}_{\boldsymbol{i=1}}^{\boldsymbol{k}}$ es llamado base para el conjunto $\boldsymbol{x} = \{\boldsymbol{x_i}\}_{\boldsymbol{i=0}}^{\boldsymbol{M}}$ si las señales atómicas, también llamadas átomos, son linealmente independientes.

Esto implica que cada señal en x puede ser únicamente reconstruida por una sola combinación lineal de átomos en D.

$$\boldsymbol{x} = \sum_{\boldsymbol{i=1}}^{\boldsymbol{k}} \boldsymbol{\alpha_i}\, \boldsymbol{d_i} = \boldsymbol{D\alpha}, \qquad (1)$$

donde $\boldsymbol{D} \in \boldsymbol{R^{NxK}}$ y es una matriz compuesta de K columnas $\boldsymbol{D} = [\boldsymbol{d_1}, \boldsymbol{d_2} \ldots \boldsymbol{d_k}]$ y las entradas del vector $\propto = [\propto_{\boldsymbol{1}}, \propto_{\boldsymbol{2}} \ldots \propto_{\boldsymbol{k}}]^{\boldsymbol{T}}$ son los coeficientes de esta combinación lineal.

Si los elementos de la base $\boldsymbol{D}$ son mutuamente otonormales entonces $\boldsymbol{d_i^T}\, \boldsymbol{d_j} = \boldsymbol{0}$ si $\boldsymbol{i} = \boldsymbol{j}$ y cada átomo radicara en la híper- esfera $\boldsymbol{d_i^T}\, \boldsymbol{d_i} = 1 (\boldsymbol{D^T}\, \boldsymbol{D} = \boldsymbol{I})$. Después de esto, la base está completa como es el caso de la base de Fourier.

Permitiendo el uso de átomos linealmente dependientes el diccionario respectivo de la matriz $\boldsymbol{D}$ llega a ser redundante. Existen múltiples opciones del vector $\boldsymbol{\alpha}$ para la construcción de una señal $\boldsymbol{x}$ acorde a $\boldsymbol{x} = \boldsymbol{D\alpha}$.

Decimos que la señal $\boldsymbol{x}$ admite una representación esparsiva sobre la base $\boldsymbol{D}$ cuando es concisamente reconstruida con pocos átomos y la señal es caracterizada por un factor esparsivo L cuando su representación esparsiva $\boldsymbol{\alpha}$ tiene L entradas diferente de cero ($l^0 - norm$).

$$||\boldsymbol{\alpha}||_0 \leq L \tag{2}$$

El resultado de la representación, es un poderoso modelo para reducción de almacenamiento y transmisión. Este modelo sugiere diccionarios redundantes óptimos para diferentes clases de señales.

2.3 Algoritmo K-SVD (K Descomposiciones de Valores Singulares)

El algoritmo K-SVD (K Descomposiciones de Valores Singulares) trabaja en conjunto con cualquier otro algoritmo de búsqueda. Se considera un algoritmo simple y diseñado para ser una generalización directa del algoritmo K-means. El K-SVD es muy eficiente debido a la alta codificación esparsiva además de tener un método de actualización de diccionarios acelerado.

2.3.1 K-means algoritmo para Vector de Cuantización (VQ).

Un código que incluye K palabras de código es usado para representar una amplia familia de vectores (señales) $\boldsymbol{Y} = \{\boldsymbol{y}_i\}_{i=1}^{N}$ $(N \gg K)$ mediante la búsqueda del vecino más cercano. Esto lleva a la comprensión o la descripción de esas señales como las agrupaciones en R^N rodeando las palabras de código elegido.

K-means puede ampliarse para sugerir una asignación difusa y una matriz de covarianza por cada grupo, por lo que los datos son modelados por una combinación de Gaussianas.

Denotamos la matriz de código $\boldsymbol{C} = [\boldsymbol{c}_1, \boldsymbol{c}_2, \ldots \boldsymbol{c}_k]$, donde el codewords vienen siendo las columnas.

Cuando **C** es dada, cada señal es representada por su palabra de código más cercano, por lo que nosotros podemos escribir $\boldsymbol{Y}_i = \boldsymbol{C}\boldsymbol{x}_i$, donde $\boldsymbol{x}_i = \boldsymbol{e}_j$ y es un vector de base trivial con todas las entradas de cero excepto el que está en la posición j-th.

La *j* indexada es seleccionada como:

$$\forall_{k \neq j} ||\boldsymbol{y}_i - \boldsymbol{C}\boldsymbol{e}_j||_2^2 \leq ||\boldsymbol{y}_i - \boldsymbol{C}\boldsymbol{e}_k||_2^2 \quad (3)$$

Esto es considerado como un caso extremo de codificación esparsiva en la que solamente un átomo es permitido para participar en la construcción de $\boldsymbol{y}_i$ y el coeficiente es forzado a ser 1. La representación MSE (Error Cuadrático Medio) por $\boldsymbol{y}_i$ es definida como $\boldsymbol{e}_i^2 = ||\boldsymbol{y}_i - \boldsymbol{C}\boldsymbol{x}_i||_2^2$ y el MSE total es:

$$E = \sum_{i=1}^{K} \boldsymbol{e}_i^2 = ||\boldsymbol{Y} - \boldsymbol{C}\boldsymbol{X}||_f^2 \quad (4)$$

El problema de entrenamiento del VQ (Cuantización de Vectores) es encontrar un código **C** que minimice el error E, sujeto al límite de la estructura de **X**, cuyas columnas se deben de tomar desde la base trivial,

$$\min_{C,X}\{||\boldsymbol{Y} - \boldsymbol{C}\boldsymbol{X}||_F^2\} \; sujeto\ a\ \forall_i, \boldsymbol{x}_i = \boldsymbol{e}_k \;\; k \neq i \quad (5)$$

El algoritmo K-means es un método iterativo usado para el diseño de un optimo código para VQ [13]. En cada iteración hay dos etapas, la primera es para la codificación esparsiva que evalúa esencialmente **X** y la segunda etapa es para la actualización del código.

La etapa de la codificación esparsiva asume un código conocido C^{J-1} y calcula una **X** factible que minimiza el valor de (5). Es evidente que en cada iteración se garantiza ya sea una reducción o que no haya ningún cambio en el MSE (Error Cuadrático Medio).

El problema de la representación esparsiva puede ser vista como una generalización de VQ (5), donde nosotros permitimos cada señal de entrada para

ser representada por una combinación lineal de palabras de código que llamaremos diccionarios.

Por lo tanto el vector de coeficientes no se le permite más de una entrada diferente de cero, por lo que estos pueden tener valores arbitrarios. Para este caso, la minimización corresponde a la ecuación (5) y de esa forma obtener el mejor diccionario posible de la representación esparsiva de este ejemplo **Y**.

$$\min_{D,X}\{||\boldsymbol{Y} - \boldsymbol{DX}||_F^2 \; sujeto\ a\ \forall_i\ ||\boldsymbol{x}_i||_0 \leq\ T_o \tag{6}$$

donde T_o es el factor de esparsividad.

Un objetivo principal, alternativamente, podría cumplirse considerando:

$$\min_{D,X}\sum_i ||\boldsymbol{x}_i||_0\ sujeto\ a\ ||\mathbf{Y}\text{-}\mathbf{DX}||_F^2 \leq\ \varepsilon, \tag{7}$$

Por un valor fijo ε.

donde ε es un error.

Se minimiza la ecuación (6) de forma iterativa. Primero se fija **D** donde el objetivo es encontrar el mejor coeficiente de la matriz **X**. Cualquier algoritmo puede ser utilizado para el cálculo de coeficientes siempre y cuando se puede suministrar una solución con un predeterminado número de elementos distintos de cero, T_o.

Una vez hecha la codificación esparsiva, el segundo paso sólo desempeña la búsqueda de un mejor diccionario. Este proceso actualiza una columna en el tiempo, fijando todas las columnas en **D**, excepto una, $\boldsymbol{d}_k$, y encontrando una nueva columna $\boldsymbol{d}_k$ y nuevos valores para sus coeficientes logrando reducir el MSE.

El proceso de actualización de una columna de **D** en el tiempo es un problema que tiene una sencilla solución basado en el “Descomposiciones de Valores Singulares” (SVD) permitiendo un cambio en los coeficientes mientras actualiza acelerando la convergencia de columnas, ya que las columnas posteriores basaran sus actualizaciones en los coeficientes más relevantes.

2.3.2 Descripción detallada del K-SVD.

Se muestra el K-SVD a detalle. Se recuerda que la función objetivo es

$$\min_{D,X}\{||\boldsymbol{Y}-\boldsymbol{DX}||_F^2\}\ sujeto\ a\ \forall_i,\ ||\boldsymbol{x}_i||_0 \leq\ T_o \quad (8)$$

Se considera en primer lugar la fase de codificación dispersa, donde asumimos que **D** es fija, y consideramos el problema de optimización anterior como una búsqueda de representación dispersa con coeficientes resumidos en la matriz **X** [10].

El término de penalización puede ser reescrito como:

$$||\boldsymbol{Y}-\boldsymbol{DX}||_F^2 = \sum_{i=1}^{N} ||\boldsymbol{y_i}-\boldsymbol{Dx}_i||_2^2$$

donde $||x||_2$ se refiere a la norma 2, que es distancia euclidiana.

Además el problema planteado en (8) puede ser desacoplado para N distintos problemas de la forma:

$$i = 1,2,\dots,N,\ \min_{x_i}\{||\boldsymbol{y}_i-\boldsymbol{Dx_i}||_2^2\} sujeto\ a\ ||\boldsymbol{x}_i||_0 \leq\ T_0 \quad (9)$$

Pasamos ahora a la segunda parte, al proceso de actualización de los diccionarios con los coeficientes diferente de cero. Asumimos que **X** y **D** son fijas poniendo en cuestión una columna en el diccionario, $\boldsymbol{d}_k$, y a los coeficientes que le corresponden de la ith fila en **X**, denotado como $\boldsymbol{x}_T^i$ (este no es un vector que está en la ith columna de **X**).

Regresando al objetivo de la función (5) el término de penalización pueden ser descritas como [10]:

$$||\boldsymbol{Y}-\boldsymbol{DX}||_F^2 = ||\boldsymbol{Y}-\sum_{j=1}^{K}\boldsymbol{d}_j\boldsymbol{x}_j||_F^2 = ||(\boldsymbol{Y}-\sum_{j\neq k}\boldsymbol{d}_j\boldsymbol{x}_T^j)-\boldsymbol{d}_k\boldsymbol{x}_T^k = ||\boldsymbol{E}_k-\boldsymbol{d}_k\boldsymbol{x}_T^k||_F^2 \quad (10)$$

Se ha descompuesto la multiplicación **DX** para la suma de **K** matrices de rango 1. La matriz $\boldsymbol{E}_k$ representa el error para todos los ejemplos **N** cuando se retira el átomo de k-ésimo. Para este caso estaría bien el uso del SVD para encontrar $\boldsymbol{d}_k$

y $\boldsymbol{x}_T^k$. El SVD encuentra el rango más cerca que se aproxime a $\boldsymbol{E}_k$, y esto reducirá el error el error definido en (10). Sin embargo habrá un error porque el nuevo vector x_T^k será ocupado por una nueva actualización de d_k no logrando cumplir la restricción de escasez.

Un remedio para el problema es simple y bastante intuitivo. Se define $\boldsymbol{\omega}_i$como el grupo de índices apuntados para ejemplos $\{y_i\}$ que usa el átomo $\boldsymbol{d}_k$donde $\boldsymbol{x}_T^k(i)$ es diferente de cero [10].

Así que,

$$\boldsymbol{\omega}_k = \{i | 1 \leq i \leq K, \boldsymbol{x}_T^k(i) \neq 0\}, \tag{11}$$

Donde se define a ω_k tiene una matriz de tamaño $Nx|\omega_i|$, con unos en la $(\boldsymbol{\omega}_k(i), i)$ y ceros en lo demás.

Multiplicando $\boldsymbol{x}_R^k$= $\boldsymbol{x}_T^k \boldsymbol{\omega}_k$, este se contrae a la fila vector $\boldsymbol{x}_T^k$ descartando las entradas de cero, resultando con la columna vector $\boldsymbol{x}_R^k$ de longitud $|\omega_k|$ que incluye un subconjunto de ejemplos que actualmente usan al átomo $\boldsymbol{d}_k$.

El mismo efecto pasa con $\boldsymbol{E}_k^R = \boldsymbol{E}_k \boldsymbol{\omega}_k$ implicando una selección de columnas de error que corresponde a ejemplos que usan el átomo $\boldsymbol{d}_k$.

Con esta notación se regresa a (10) y se sugiere minimizar con respecto a $\boldsymbol{d}_k$ y $\boldsymbol{x}_T^k$, pero la solución forzada de $\tilde{\mathrm{x}}_T^k$ para tener el mismo soporte como la original $\boldsymbol{x}_T^k$.

Esto es equivalente a la minimización de [10]:

$$\left|\left|\boldsymbol{E}_k \boldsymbol{\omega}_k - \boldsymbol{d}_k \boldsymbol{x}_T^k \boldsymbol{\omega}_k\right|\right|_F^{2=} \; ||\boldsymbol{E}_k^R - \boldsymbol{d}_k \boldsymbol{x}_R^k||_F^2 \tag{12}$$

Tomando la matriz restringida $\boldsymbol{E}_k^R$, el SVD la descompone a $\boldsymbol{E}_k^R = \boldsymbol{U}\Delta\boldsymbol{V}^T$. Se define la solución $\tilde{\mathbf{d}}_k$ como la primera columna de **U**, y el vector de coeficiente $\boldsymbol{x}_R^k$ como la primera columna de **V** multiplicada por $\Delta(1,1)$. Para esta solución fue necesario [10]:

- Las columnas de **D** deben permanecer normalizadas
- Cualquier soporte de todas las representaciones se queda igual o presenta una pequeña anulación de los términos

Se llama a esto algoritmo K-SVD. Mientas K-Means aplica K cálculos de principales actualizaciones del código, el K-SVD obtiene una actualización del diccionario por el cálculo K-SVD, que determina una columna.

A continuación se muestra el algoritmo del K-SVD [10],

Encuentra el mejor diccionario para representar los datos muestreados $\{y_i\}_{i=1}^{N}$ como composición esparsiva resolviendo,

$\overset{min}{D,X}\{||\boldsymbol{Y}-\boldsymbol{DX}||_F^2\} sujeto\ a\ \forall_i ||x_i||_0 \leq T_0.$

Inicialización: Establecer la matriz del diccionario $D^0 \in R^{nxk}$ con l^2columnas normalizadas.

Se establece J=1.

Se repite hasta converger.

- Etapa del código esparsivo: usa cualquier algoritmo de búsqueda para calcular la representación de vectores x_i por cada ejemplo y_i para aproximar la solución de,
  $i = 1,2, \dots, N, \overset{min}{x_i}\{||\boldsymbol{y}_i - \boldsymbol{Dx_i}||_2^2\}$ Sujeto a $||\boldsymbol{x}_k||_0 \leq T_o$.

- Actualización del código: por cada columna $k = 1,2 \dots, K\ en\ D^{(J-1)}$,
  Define el grupo de ejemplos que usa este átomo, $\omega_k = \{i \big| 1 \leq i \leq N, \boldsymbol{x}_t^k(i) \neq 0\}$.

  Calcula la representación total de la matriz de error, E_k, por,

  $\boldsymbol{E}_k = \boldsymbol{Y} - \sum_{j \neq k} \boldsymbol{d}_j \boldsymbol{x}_T^j$

Restringe $\boldsymbol{E}_k$ al tomar solamente las columnas correspondientes a $\boldsymbol{\omega}_k$ y obtener $\boldsymbol{E}_K^R$.

- Aplica la descomposición SVD $\boldsymbol{E}_K^R = \boldsymbol{U}\Delta\boldsymbol{V}^T$. Toma la columna $\tilde{\mathbf{d}}_k$ del diccionario actualizado para ser la primera columna de **U**. Actualiza el vector de coeficiente $\boldsymbol{X}_R^K$ para ser la primera columna de **V** multiplicado por $\Delta\,(1,1)$.
- Establece $J = J + 1$.

2.4 Aprendizaje de diccionarios redundantes.

En lugar de usar diccionarios predefinidos, tal como las wavelets para la reconstrucción esparsiva de señales, los diccionarios pueden ser adaptados para un conjunto de señales de entrenamiento.

Dejando las columnas de la matriz $\boldsymbol{X} = [\boldsymbol{x_1}, \boldsymbol{x_2}, \dots \boldsymbol{x_j}] \in \boldsymbol{R^{Nxj}}$ como el conjunto de señales de entrenamiento de una clase particular para ser reconstruida esparsivamente a través de un optimo diccionario redundante $\boldsymbol{D}$**.** El aprendizaje de diccionario se describe con el siguiente problema de minimización:

$$[\boldsymbol{D}, \boldsymbol{A}] = \arg min_{DA} ||\boldsymbol{X} - \boldsymbol{DA}||_F^2 \; subject\ to \; || \propto_i ||_0 \; ; L\forall\ i = 1,2 \dots j \qquad (13)$$

Las columnas de la matriz $\boldsymbol{A} = \left[\propto_{\boldsymbol{1}}, \propto_{\boldsymbol{2}}, \dots \propto_{\boldsymbol{j}}\right] \in \boldsymbol{R^{Kxj}}$ son la representación esparsiva del conjunto de señales en $\boldsymbol{X}$ tomando en cuenta que **L** es el factor esparsivo.

Diferentes enfoques para el aprendizaje de diccionarios han sido desarrollados y basados en dos pasos. El primer pasos consiste en encontrar una representación esparsiva A del entrenamiento de la señal $\boldsymbol{X}$ basado en un diccionario $\boldsymbol{D}$ a través de un algoritmo de búsqueda tal como el *Matching Pursuit* (MP) u *Ortogonal Matching Pursuit* (OMP).

Durante el segundo paso, las señales atómicas son actualizadas asumiendo coeficientes de reconstrucción fija. Un algoritmo adecuado para adaptar

diccionarios para la representación esparsiva es el método K-SVD (K-Descomposición de Valores Singulares).

La idea principal del algoritmo K-SVD consiste en expresar la reconstrucción total de la función de error.

$$\|\mathbf{X}-\mathbf{D}\,\mathbf{A}\|_{\mathrm{F}}^{2}=\left\|\mathbf{X}-\sum_{\mathrm{i}=1}^{\mathrm{K}}\mathbf{d}_{\mathrm{i}}\,\boldsymbol{\alpha}_{\mathrm{i}}^{\mathrm{T}}\right\|_{\mathrm{F}}^{2}=\left\|\mathbf{X}-\sum_{\substack{\mathrm{i}=1\\ \mathrm{i}\neq\mathrm{j}}}^{\mathrm{K}}\mathbf{d}_{\mathrm{i}}\,\boldsymbol{\alpha}_{\mathrm{i}}^{\mathrm{T}}-\mathbf{d}_{\mathrm{j}}\,\boldsymbol{\alpha}_{\mathrm{j}}^{\mathrm{T}}\right\|_{\mathrm{F}}^{2}=\left\|\mathbf{E}_{\mathrm{j}}-\mathbf{d}_{\mathrm{j}}\,\boldsymbol{\alpha}_{\mathrm{j}}^{\mathrm{T}}\right\|_{\mathrm{F}}^{2},$$

(14)

Donde el átomo $\boldsymbol{d_i}$es el ith columna de $\mathbf{D}$ y $\boldsymbol{\alpha}_{\mathbf{i}}^{\mathbf{T}}$ es la ith columna de **A**. El algoritmo K-SVD es un proceso iterativo de dos pasos para formar un diccionario, en cada iteración el primer paso consiste en estimar la representación esparsiva de **A** de acuerdo a [10]:

$$\boldsymbol{A}=arg\min_{\boldsymbol{A}}\|\boldsymbol{X}-\boldsymbol{D}\,\boldsymbol{A}\|_{F}^{2}\ subject\ to\ \|\boldsymbol{\alpha}_{i}\|_{0}<L;\ i=1,2,\dots,J. \tag{15}$$

donde J es el número de señales de prueba.

En el segundo paso, cada átomo $\boldsymbol{d_i}$ y correspondiendo a la columna $\boldsymbol{\alpha}_{\mathbf{i}}^{\mathbf{T}}$ son encontrados al usar el método SVD conforme a [10]:

$$\left[\boldsymbol{d}_{j},\boldsymbol{\alpha}_{j}^{T}\right]=\arg\min_{\boldsymbol{d}_{j},\boldsymbol{\alpha}_{j}^{T}}\left\|\boldsymbol{E}_{j}\boldsymbol{\Omega}_{j}-\boldsymbol{d}_{j}\,\boldsymbol{\alpha}_{j}^{T}\right\|_{F}^{2}, \tag{16}$$

Donde la matriz $\boldsymbol{\Omega}_j$ reduce $\boldsymbol{E}_j$ manteniendo sólo aquellas columnas que tienen coeficientes de reconstrucción diferentes de cero en $\boldsymbol{\alpha}_j^T$. Usando el SVD, la matriz $\boldsymbol{E}_j\boldsymbol{\Omega}_j$ puede ser aproximada por una matriz rank-1.

$$\boldsymbol{E}_{j}\boldsymbol{\Omega}_{j}=\sum_{k}\sigma_{k}\boldsymbol{u}_{k}\boldsymbol{v}_{k}^{\boldsymbol{T}}\approx\sigma_{1}\boldsymbol{u}_{1}\boldsymbol{v}_{1}^{\boldsymbol{T}};\ \sigma_{1}>\sigma_{2}>\sigma_{3}>\cdots, \tag{17}$$

donde $\boldsymbol{d}_j=\boldsymbol{u}_1$, y $\boldsymbol{\alpha}_j^T=\sigma_1\boldsymbol{v}_1^{\boldsymbol{T}}$

2.5 Clasificación de señales basadas en aprendizaje de diccionarios

El aprendizaje de diccionarios se ha aplicado con éxito a diferentes tareas de aplicación. Dejando el diccionario $\boldsymbol{D}_i \in R^{NxK}$ al ser entrenado esparsivamente reconstruye un conjunto de señales $X_i \in R^{NxJ}$ perteneciente a la ith clase C_i [10].

Una prueba de la señal $X \in R^N$ es asignada para una de las P clases para la primera estimación del conjunto de residuos $\{r_i(\boldsymbol{x}, \boldsymbol{D}_i)\}_{i=1}^{P}$ donde cada residuo corresponde a la reconstrucción esparsiva de la señal bajo un diccionario.

$$r_i(\boldsymbol{x}, \boldsymbol{D}_i) = \min_{\boldsymbol{\alpha}}[\|\boldsymbol{x} - \boldsymbol{D}_i\,\boldsymbol{\alpha}\|_2^2 + \lambda\,\|\boldsymbol{\alpha}\|_0]\,;\; i = 1, \dots, P; \tag{18}$$

Después se asigna una clase para encontrar el residuo más pequeño.

$$class = arg \min_{i\epsilon\{1,\dots,p\}} r_i(\boldsymbol{x}, \boldsymbol{D}_i) \tag{19}$$