Técnicas de Regularización en el Aprendizaje Estadístico.

escrito por

JUAN RAFAEL PEREA LUQUE

Tutor: Dr. Hilario Navarro Veguillas.

Facultad de Ciencias
UNIVERSIDAD NACIONAL DE EDUCACIÓN A DISTANCIA

Trabajo presentado para la obtención del título de
Master Universitario en Matemáticas Avanzadas de la UNED.
Especialidad. Estadística e Investigación Operativa.

SEPTIEMBRE 2019

Extracto

En el presente trabajo se exponen las principales técnicas de regularización en el aprendizaje estadístico para sistemas de alta dimensión ($p \gg n$) asumiendo la consideración de dispersión (*sparsity*), y su implementación en los más comunes métodos de agrupamiento (*clustering*), como son el agrupamiento jerárquico y el agrupamiento k-medias, (*k-means*) ilustrando su aplicación con ejemplos realizados en R.

Abstract:

In this paper, the main techniques of regularization in statistical learning for high-dimensional systems ($p \gg n$) are shown, assuming the consideration of sparsity, and its implementation in the most common clustering methods, such as hierarchical clustering and k-means clustering. Illustrating its application with examples developed in R language.

Keywords: Clustering, Elastic Net, LASSO, Hierarchical clustering, High-dimensional statistic, k-means clustering, R, Regularization methods, Ridge, Sparse systems, Statistic learning,

DEDICATORIA Y AGRADECIMIENTOS

Mi más sincero agradecimiento a aquellos que han hecho posible este trabajo, en primer lugar al Dr. Navarro por la tutela y orientación en su realización, a los distintos autores en los que se basa el mismo por haber proporcionado el soporte necesario que en él se expone, y a mi familia por haberme concedido el tiempo necesario para poder llevarlo a cabo.

ÍNDICE DE GRÁFICAS

ÍNDICE DE TABLAS

Los avances tecnológicos han facilitado notablemente la obtención, almacenamiento y tratamiento de datos en formato digital, por lo que actualmente es posible disponer de bases de datos de gran tamaño que encierran información relevante y en cuyo análisis confluyen tanto consideraciones analíticas orientadas por la aplicación de técnicas estadísticas como consideraciones tecnológicas provenientes del campo de la informática.

Por lo tanto, nuestro contexto científico de estudio es la estadística, que fruto de los avances tecnológicos, en cuanto a la recopilación, almacenamiento y tratamiento de datos, dispone de un volumen de datos y una capacidad de tratamiento impensable anteriormente, todo este soporte tecnológico (informático) conlleva que su desarrollo vaya de la mano de las ciencias de la computación y se tengan en consideración la relevancia de los criterios de eficiencia en cuanto a tiempo y capacidad de computación. La diferencia entre ambas disciplinas paulatinamente se torna más difusa, y sus avances más complementarios y simultáneos, de manera que, en muchas ocasiones, y cada vez con mayor frecuencia se usan los términos aprendizaje estadístico y aprendizaje automático (*Machine Learning*) de forma indistinta.

Nuestro punto de partida es un conjunto de datos o de información y a partir de aquí fijamos como objetivo la obtención de un modelo, una función, una aplicación que nos permita predecir el valor, el comportamiento o la imagen de nuevos datos, con la mayor exactitud posible.

Comenzaremos esbozando porque el aprendizaje estadístico requiere de las técnicas de regulación, en el contexto de los sistemas de alta dimensión y los modelos dispersos.

En nuestra base de partida (los datos) hay dos aspectos estructurales que suponen un inconveniente importante para obtener este conocimiento/aprendizaje:

1. Desconocemos el número de variables o características relevantes que realmente intervienen en la modelación del resultado final.

2. Por distintos condicionantes como coste, naturaleza de la medida, incertidumbre, ruido de la medida, disponibilidad, etc. el número de muestras es escaso o relativamente escaso respecto del número de variables potenciales que describan el modelo.

Conjugando que se trata de un proceso inductivo, que desconocemos los factores determinantes que intervienen y que disponemos de relativamente pocos datos, podemos resumir el proceso como la obtención de una aplicación partiendo de espacios de alta dimensión y con un conjunto limitado, de calidad desconocida (atípicos, ruido, etc.), de datos.

Sin perder de vista lo anterior, el objetivo se centra en obtener la simplicidad subyacente en un modelo que describa de forma adecuada la realidad.

Lo anteriormente expuesto nos lleva a focalizarnos en los **sistemas de alta dimensión** y concretamente en los sistemas dispersos.

El término "alta dimensión" se refiere al caso en el que el número de parámetros desconocidos que se estimarán, p, es de un orden mucho mayor que el número de observaciones, n, es decir, $(p \gg n)$. Como los métodos estadísticos tradicionales suponen muchas observaciones y algunas variables desconocidas, no pueden hacer frente a las situaciones cuando $p \gg n$.

Para describir los problemas de alta dimensión y las limitaciones y dificultades asociadas con ellos consideramos un modelo lineal

$$\boldsymbol{Y} = \boldsymbol{X}\boldsymbol{\beta}^0$$

donde $\boldsymbol{Y} \in \mathbb{R}^n$ es el vector respuesta, $\boldsymbol{X} \in \mathbb{R}^{n\times p}$ es la matriz de variables independientes y $\boldsymbol{\beta}^0 \in \mathbb{R}^p$ es el vector de coeficientes de regresión verdaderos desconocidos.

Cuando $p > n$ el sistema es linealmente subdeterminado y no existe una solución única, es más, realmente hay un conjunto infinito de soluciones.

Por consiguiente, es imposible hallar la solución correcta del conjunto de soluciones infinitas sin alguna información adicional o restricción que acote el número de variables a incorporar al modelo. Una de esas restricciones que nos permiten obtener el modelo, y que con mucha frecuencia responde a la realidad, es el supuesto de dispersión que considera que solo un reducido número de variables tiene valor distinto de cero en los coeficientes de regresión verdaderos.

Lo que podría asimilarse a una especie de truncamiento de coeficientes llevados a cero que proporciona la simplificación necesaria para hacer manejable e interpretable el modelo, que podría expresarse como

$$\widehat{\boldsymbol{\beta}} = \underset{\boldsymbol{\beta}\in\mathbb{R}^p}{\operatorname{argmin}} \left\{ \sum_{i=1}^{n} \left(y_i - \beta_0 - \sum_{j=1}^{p} \beta_j x_{ij} \right)^2 \right\}, \qquad \text{sujeto a} \quad \sum_{j=1}^{p} \left\| \beta_j \right\|_0 \leq t$$

Un **modelo estadístico disperso** es aquel en el que solo un número relativamente pequeño de parámetros (o predictores) juegan un papel importante siendo nulos el resto, y dentro del aprendizaje estadístico es en los modelos estadísticos dispersos donde las técnicas de regulación que son las implementadas para llevar a cabo la selección de variables (inducir la dispersión del modelo) adquieren su principal aplicación.

Por lo tanto, el aprendizaje estadístico trabajando con sistemas dispersos en conjuntos de alta dimensión ($p \gg n$) requiere la aplicación de **técnicas de regularización** que permitan la obtención de modelos que ofrezcan un equilibrio entre simplicidad (menor complejidad donde solo un reducido número de características nos permita definir el modelo) y precisión de la respuesta (mínimo error de predicción).

CAPÍTULO 1

RESULTADOS BÁSICOS Y SU EVOLUCIÓN HASTA LA ACTUALIDAD

En este primer capítulo se describirán los métodos de regulación que permiten ofrecer una solución a los sistemas dispersos mediante la búsqueda y selección de aquellas variables que son relevantes para el modelo permitiendo descartar (aplicando un coeficiente cero) a aquellas otras que no son relevantes en la construcción de dicho modelo.

Se realiza un recorrido desde los métodos más directos, como es la **selección de variables** a través de distintos algoritmos, que debido tanto al crecimiento de los volúmenes de datos como a la dimensionalidad de estos pronto presentaron tanto problemas de inestabilidad en sus resultados como problemas computacionales.

El recorrido, con enfoque un temporal, continúa describiendo el estimador ***Ridge*** que, si bien no es estrictamente aplicable a sistemas dispersos, ya que no realiza la selección de variables, constituye la primera aplicación de las técnicas de regularización mediante aplicación de penalización, siendo este estimador posteriormente incorporado a técnicas más elaboradas en cuanto a selección de variables, al presentar una característica tan importante como es su funcionamiento ante la colinealidad de las variables.

La aparición del método **Garrote No Negativo** (GNN) llevó a cabo la primera selección de variables aplicando un procedimiento de penalización.

GNN sirvió de inspiración al método básico más conocido y utilizado **LASSO** que incorpora tanto la capacidad de selección de variables como la contracción de coeficientes. La implementación de algoritmos computacionales (LAR y camino de descenso coordinado) significó una amplia proyección e implantación de LASSO, que sin embargo alberga inconvenientes de consistencia, así como el no cumplimiento de la propiedad oráculo y debilidades como la “sensibilidad” a la correlación de variables.

Se describe brevemente el estimador ***Bridge*** que contempla la generalización del método de penalización y concretamente la capacidad de aplicación de diferentes normas como función de penalización, entre ellas las normas $0 < q < 1$ con mayor potencialidad para "anular" variables, aunque con el gran inconveniente de su complejidad computacional debido a la no convexidad de estas normas.

Una vez recorrida la exposición de los métodos básicos de regularización con especial relevancia de LASSO, se describen aquellos métodos que constituyen básicamente la aplicación de mejoras que corrigen parte de los inconvenientes de LASSO, como el comportamiento ante correlaciones de las variables, con este objetivo aparece una combinación con *Ridge* que supuso la creación de ***Elastic Net***.

Orientado a mejorar la consistencia de LASSO se desarrolló un método en dos fases que incorpora una ponderación en las variables del modelo, **LASSO-Adaptativo**.

Otro avance que mejoraba sesgo y consistencia de LASSO además de proporciona una mayor selección de variables en el sentido de trabajar con sistemas de muy alta dimensión supuso la aparición de **LASSO-Relajado**, siendo también un método en dos fases que requiere una preselección de variables sobre las que operar.

Finalmente también se exponen y describen métodos de regulación con base en LASSO que constituyen aplicaciones o adaptaciones a situaciones específicas de los sistemas dispersos de alta dimensión como es la necesidad de contemplar la agrupación de variables que realiza el **LASSO-Grupal**, y ahondando en la especificidad de esta agrupación contemplar las situaciones de dispersión a su vez dentro de cada grupo que se realiza mediante **LASSO-Grupal-Disperso** o el caso en que los grupos contemplan en su estructura variables que están presente en más de un grupo, situación abordada mediante **LASSO-Grupal-Solapado**.

Otra de las situaciones específicas es la inclusión de la consideración de la ordenación de las variables o su distancia o proximidad relativa, aspectos que son considerados mediante el método de **LASSO-Fusionado.**

Es decir, realizaremos un recorrido por las estrategias implementadas que permiten la obtención de modelos eficientes en sistemas dispersos de alta dimensión mediante técnicas de regulación.

REGULARIZACIÓN INICIAL BÁSICA

Comenzamos con la descripción de los primeros métodos implementados que permiten la selección de variables mediante la incorporación de un condicionante, es decir, la aplicación de una penalización lo cual constituye la estructura de los métodos de regularización.

Si bien, inicialmente la selección directa y/o secuencial de variables no tuvo el enfoque de regularización, al considerar estas técnicas como incorporación de una penalización condicionada por el número de variables las consideramos regularización, en este caso determinada por la norma[1] L_0 como veremos seguidamente.

Aunque no determina la selección de variables *Ridge*, sí es propiamente un método que implementa una penalización L_2, y que finalmente será incorporado para la mejora de métodos que sí obtienen selección de variables. Como método de selección de variables mediante la aplicación de penalización (regularización) surgió Garrote No Negativo, y seguidamente surgió LASSO.

De forma general aquí se describen las técnicas de regularización mediante penalización y se presenta el estimador *Bridge* como generalización de regularización en función de las diferentes normas.

[1] Aunque estrictamente L_0 no es una norma, de manera frecuente se suele considerar de esta forma.

SELECCIÓN DE VARIABLES

La búsqueda de la simplicidad en un modelo pasa por la reducción de su complejidad, y la complejidad viene condicionada por el número de variables que interviene en el modelo.

La selección de variables nos permite modelos más simples y estables, por lo tanto, con una mejor interpretabilidad, y es en los sistemas dispersos $(p \gg n)$ donde la necesaria reducción de variables adquiere inevitablemente un papel protagonista.

Consideramos el conjunto de p variables de entrada, independientes o características como $\{X_1, X_2, \cdots, X_p\}$ y una variable respuesta Y. Para n observaciones tendremos un conjunto $D = \{(x_{i1}, x_{i2}, \cdots, x_{ip}, y_i),\ \ i = 1,2, \cdots, n\}$ con $\boldsymbol{x}_i \in \mathbb{R}^p$. Siendo el vector de respuestas $\boldsymbol{Y}$, y la matriz de características $n \times p$ la definimos como $\boldsymbol{X}$ con sus columnas $\boldsymbol{X}_{\cdot j} \in \mathbb{R}^n$, es decir.

$$\boldsymbol{X} = (\boldsymbol{X}_{\cdot 1} \cdots \boldsymbol{X}_{\cdot j} \cdots \boldsymbol{X}_{\cdot p}) = \begin{pmatrix} x_{11} & \cdots & x_{1j} & \cdots & x_{1p} \\ \vdots & & \vdots & & \vdots \\ x_{i1} & \cdots & x_{ij} & \cdots & x_{ip} \\ \vdots & & \vdots & & \vdots \\ x_{n1} & \cdots & x_{nj} & \cdots & x_{np} \end{pmatrix}$$

Si consideramos la pertenencia o inclusión de cada variable p al modelo final, por simple combinatoria podremos generar 2^p potenciales modelos. Una primera aproximación consiste en ajustar los 2^p modelos posibles y comparar los mejores de cada tamaño $k \in \{1, \cdots, p\}$.

La comparación de los distintos modelos obtenidos se realiza a través de alguna medida que considere el ajuste a los datos de entrenamiento, pero que penalice la complejidad del modelo, es decir, que "busque" la reducción de variables en pos de potenciar modelos simples, de forma tal que posea un alto poder predictivo (generalización sobre datos de validación).

Para determinar cuál es el mejor subconjunto de variables se consideran distintos criterios:

- Estadístico C_p de Mallows, (Mallows, 1973) [01]. Selecciona el modelo con mayor capacidad de predicción en lugar del que está mejor ajustado. Ésta se mide con el error cuadrático medio. C_p solo es posible cuando $\hat{\sigma}^2 \cong \sigma^2$, y esto lo conseguimos con una muestra lo suficientemente grande y $\hat{\sigma}^2 = \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{n-p-k}$, estando las p variables necesarias incluidas en las $(p + k)$, para que el estimador sea insesgado.

Cuando el modelo es el adecuado, $C_p \approx p$, nos quedamos con el modelo que tenga el más próximo a p y preferiblemente menor.

$$C_p = \frac{\sum_{i=1}^{n}\left(y_i - \hat{y}_{p_i}\right)^2}{\hat{\sigma}^2} + 2p$$

- Criterio de información de Akaike, AIC (Akaike, 1974) [01]. Basado en la teoría de la información, bonifica la bondad de ajuste y penaliza la inclusión de parámetros, la idea principal es maximizar el logaritmo del estimador de máxima verosimilitud esperado de un modelo determinado. $AIC = n\log\left(\hat{\sigma}_p^2\right) + 2p = log\|\hat{\boldsymbol{y}} - \boldsymbol{y}\|_2^2 + 2df$. $\hat{\sigma}_p^2$ estimador de máxima verosimilitud de la varianza del modelo con p parámetros df grados de libertad.

- Criterio de información Bayesiano, BIC (Schwarz, 1978) [01]. Misma filosofía que AIC, pero desde un enfoque bayesiano que se basa en las probabilidades a posteriori de los modelos. $BIC = n\log\left(\hat{\sigma}_p^2\right) + log(n)\cdot p = log\|\hat{\boldsymbol{y}} - \boldsymbol{y}\|_2^2 + log(n)\cdot df$ $\hat{\sigma}_p^2$ estimador de máxima verosimilitud de la varianza del modelo con p parámetros df grados de libertad.

- Coeficiente de determinación corregido $\bar{R}_{adj}^2$, (Goldeberger, 1991) [01]. Corrige el problema del coeficiente de correlación R^2 (aumenta al aumentar el número de variables consideradas) y puede ser apropiado cuando $p \cong n$.

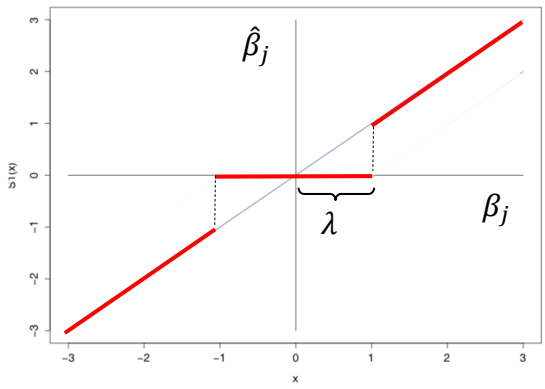
$$\bar{R}_{adj}^2 = 1 - \frac{Varianza\ residual}{Varianza\ de\ y} = 1 - \frac{n-1}{n-k-1}\frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2}$$

Los diferentes criterios de selección de variables consisten básicamente en la aplicación de una penalización a la inclusión de variables en el sistema (número de variables) y bajo determinadas consideraciones podríamos considerarlos equivalentes (Peña, 2002) [01]. Lo que puede expresarse como

$$\hat{\boldsymbol{\beta}} = \underset{\boldsymbol{\beta}\in\mathbb{R}^{\boldsymbol{p}}}{\operatorname{argmin}}\{\|\boldsymbol{Y} - \boldsymbol{X\beta}\|_2^2 + \lambda\|\boldsymbol{\beta}\|_0\}$$

Es decir, que considerando un determinado valor límite (umbral) los coeficientes que superen dicho valor serán considerados en el sistema y los que se encuentren con valores iguales o inferiores serán excluidos del modelo

$$\hat{\beta}_j = \begin{cases} \beta_j & si \quad |\beta_j| > \lambda \\ 0 & si \quad |\beta_j| \leq \lambda \end{cases}$$



Gráfica 1. Función de penalización. Umbral Duro. Selección de variables.

Constituyendo el denominado Umbral Duro (*Hard Thresholding*) que observamos en la Gráfica 1.

Pero para obtener un subconjunto de modelos con los que comparar requiere establecer un criterio, siendo el más inmediato, la comparación con todos los subconjunto posibles, sin embargo el método del mejor subconjunto de cada tamaño actualmente solo es practicable si p no es demasiado grande ($p \approx 50$, $2^{50} \approx 1{,}12589 \times 10^{15}$) no siendo factible esta estrategia cuando el número de predictores relevantes es superior (Hastie et al, 2015) [01].

Cuando no es viable una búsqueda exhaustiva de todos los submodelos posibles, una opción razonable consiste en considerar un "buen camino" a través del espacio de modelos.

Las técnicas más conocidas en estos casos son los métodos secuenciales o por pasos (*stepwise*), en los cuales en el paso de un modelo a otro se agregan o eliminan variables en cada etapa o iteración.

Estos modelos fueron los primeros métodos en aparecer y se trataba de algoritmos anidados:

- Selección regresiva (*Backward selection*), (Efroymson, 1960) [01].
- Selección progresiva (*Forward selection*), (Efroymson, 1966 [01]; Draper y Smith, 1966 [01]).
- Selección del mejor subconjunto (*Best Subset Selection*)[2], (Beale et al., 1967 [01]; Hocking y Leslie, 1967 [01]).
- Selección progresiva por etapas (*Forward Backward stagewise selection*).

Son métodos codiciosos[3] (*greedy*) que reemplazan la búsqueda de un óptimo global por la consideración sucesiva de óptimos locales, con lo cual no garantizan la mejor solución y ni siquiera la misma entre sus distintas variantes.

La mayor desventaja que poseen es su fuerte inestabilidad en el sentido de que pequeños cambios en el conjunto de datos pueden producir grandes modificaciones en los resultados, en particular en las variables seleccionadas (Breiman, 1996) [01].

Esto se debe principalmente a que realizan un proceso discreto de exploración del espacio de modelos (cada variable es seleccionada o descartada).

[2] Recientemente, Bertsimas et al. (2016) [01] presentaron una formulación denominada Optimización Entera Mixta, MIO (*Mixed Integer Optimization*) para los problemas de selección del mejor subconjunto. Usando soluciones MIO, problemas con cientos e incluso miles *p* pueden ser abordados de forma eficiente. Estudios de simulación en Bertsimas et al. (2016) [02] demostraron que la selección del mejor subconjunto generalmente proporciona una precisión de predicción superior en comparación con la selección por pasos hacia delante (*forward stepwise selection*) y LASSO, en una variedad de configuraciones de problemas.

[3] Una vez descartada o incorporada una variable al modelo esta decisión no vuelve a replantearse.

ESTIMADOR *RIDGE*

Hoerl (1962) [01] y Hoerl y Kennard (1970a, 1970b) [01] [01] abordando los problemas de colinealidad, en un modelo lineal estimado por mínimos cuadrados, en contextos $p < n$, partiendo de que $\widehat{\boldsymbol{\beta}}^{MCO} = (\boldsymbol{X}^T\boldsymbol{X})^{-1}\boldsymbol{X}^T\boldsymbol{Y}$ es la estimación por mínimos cuadrados de $\boldsymbol{\beta}$, se plantearon en un principio que la potencial inestabilidad de $\widehat{\boldsymbol{\beta}}^{MCO}$ podría ser corregida agregando una pequeña constante $k > 0$ a cada término de la diagonal de $\boldsymbol{X}^T\boldsymbol{X}$ antes de invertir la matriz.

Este proceso resultó el estimador *Ridge*:

$$\widehat{\boldsymbol{\beta}}^{Ridge}(k) = \left(\boldsymbol{X}^T\boldsymbol{X} + k\boldsymbol{I}_p\right)^{-1}\boldsymbol{X}^T\boldsymbol{Y}$$

Puesto que $\boldsymbol{I}_p = [\boldsymbol{X}^T\boldsymbol{X}]^{-1}[\boldsymbol{X}^T\boldsymbol{X}]$

$$\widehat{\boldsymbol{\beta}}^{Ridge}(k) = \left[[\boldsymbol{X}^T\boldsymbol{X}] + k[\boldsymbol{X}^T\boldsymbol{X}]^{-1}[\boldsymbol{X}^T\boldsymbol{X}]\right]^{-1}\boldsymbol{X}^T\boldsymbol{Y}$$

Podemos extraer $[\boldsymbol{X}^T\boldsymbol{X}]^{-1}$

$$\widehat{\boldsymbol{\beta}}^{Ridge}(k) = \left[\boldsymbol{I}_p + k[\boldsymbol{X}^T\boldsymbol{X}]^{-1}\right]^{-1}[\boldsymbol{X}^T\boldsymbol{X}]^{-1}\boldsymbol{X}^T\boldsymbol{Y}$$

Observamos que $[\boldsymbol{X}^T\boldsymbol{X}]^{-1}\boldsymbol{X}^T\boldsymbol{Y} = \widehat{\boldsymbol{\beta}}^{MCO}$, con lo que

$$\widehat{\boldsymbol{\beta}}^{Ridge}(k) = \left[\boldsymbol{I}_p + k[\boldsymbol{X}^T\boldsymbol{X}]^{-1}\right]^{-1}\widehat{\boldsymbol{\beta}}^{MCO}$$

Si consideramos $\boldsymbol{Z} = \left[\boldsymbol{I}_p + k[\boldsymbol{X}^T\boldsymbol{X}]^{-1}\right]^{-1}$ tenemos que $\widehat{\boldsymbol{\beta}}^{Ridge}(k) = \boldsymbol{Z}\widehat{\boldsymbol{\beta}}^{MCO}$, es decir, el estimador *Ridge* es proporcional al estimador por mínimos cuadrados. Además, si consideramos los autovalores de $\boldsymbol{X}^T\boldsymbol{X}$ como $\lambda_{max} = \lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_p = \lambda_{min} > 0$, verificamos que los autovalores de $\boldsymbol{Z}$ son $\frac{\lambda_i}{\lambda_i + k} < 1$ para $i = 1, \cdots, p$ por lo tanto, los estimadores $\widehat{\boldsymbol{\beta}}^{Ridge}$ suponen una reducción de los estimadores mínimo cuadráticos $\widehat{\boldsymbol{\beta}}^{MCO}$, por ello son denominados métodos de contracción o de encogimiento de coeficientes (*Shrinkage Methods*).

De la observación de los autovalores se desprende también que se producirá una variación desde $k = 0$ o no aplicación de ninguna penalización, donde en este caso los coeficientes obtenidos son los coeficientes mínimo cuadráticos $\widehat{\boldsymbol{\beta}}^{Ridge}(0) = \widehat{\boldsymbol{\beta}}^{MCO}$, hasta valores de máxima penalización $k \to \infty$, donde todos los autovalores de $\boldsymbol{Z}$ tienden a cero anulando los coeficientes $\widehat{\boldsymbol{\beta}}^{Ridge}(\infty) \approx \mathbf{0}$

El principal problema a resolver en la aplicación de Regresión *Ridge* es la determinación del valor de k más adecuado. La elección de este parámetro involucra un equilibrio entre los componentes de sesgo y variancia del error cuadrático medio al estimar $\boldsymbol{\beta}$.

En este sentido (y asumiendo un modelo lineal), cuanto mayor es k más grande es el sesgo, pero menor es la variancia del estimador, y la determinación final implica un compromiso entre ambos términos (Izenman, 2008) [01]. Para la obtención de valor de penalización k más adecuado Friedman et al. (2001) [01] recomiendan la validación cruzada.

La expresión más extendida es propuesta por Friedman et al. (2001) [02] partiendo del estimador *Ridge* de Hoerl y Kennard (1970a) [02], donde el valor de la penalización k es sustituido por la suma de los cuadrados de los coeficientes.

$$\widehat{\boldsymbol{\beta}}^{Ridge} = \underset{\boldsymbol{\beta}\in\mathbb{R}^{\boldsymbol{p}}}{\operatorname{argmin}} \left\{ \sum_{i=1}^{n} \left(y_i - \beta_0 - \sum_{j=1}^{p} \beta_j x_{ij} \right)^2 \right\}, \qquad sujeto\ a \quad \sum_{j=1}^{p} \beta_j^2 \leq t$$

Puesto que el estimador *Ridge* no es invariante a la escala, lo adecuado es trabajar con valores estandarizados, por lo tanto, tenemos

$$\widehat{\boldsymbol{\beta}}^{Ridge} = \underset{\boldsymbol{\beta}\in\mathbb{R}^{\boldsymbol{p}}}{\operatorname{argmin}} \left\{ \sum_{i=1}^{n} \left(y_i - \sum_{j=1}^{p} \beta_j x_{ij} \right)^2 \right\}, \qquad sujeto\ a \quad \sum_{j=1}^{p} \beta_j^2 \leq t$$

Que expresado en su versión Lagrangiana[4]:

$$\widehat{\boldsymbol{\beta}}^{Ridge} = \underset{\boldsymbol{\beta}\in\mathbb{R}^{\boldsymbol{p}}}{\operatorname{argmin}} \left\{ \sum_{i=1}^{n} \left(y_i - \sum_{j=1}^{p} \beta_j x_{ij} \right)^2 + \lambda \sum_{j=1}^{p} \beta_j^2 \right\}$$

siendo $t, \lambda \geq 0$, los respectivos parámetros de penalización por complejidad.

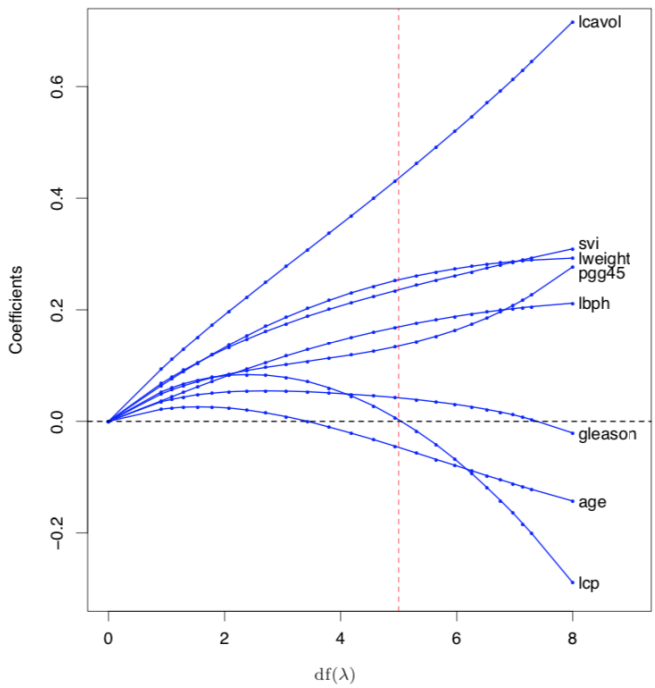
[4] Existe una correspondencia uno a uno entre t y λ

Las expresiones anteriores muestran que el estimador de *Ridge* realiza un balance entre sesgo y variancia controlando el "tamaño" del vector de coeficientes mediante una penalización de norma L_2. El estimador contrae (*shrinkage*) los coeficientes β_j hacia cero respecto de los obtenidos por MCO.

En general, regresión *Ridge* produce predicciones más precisas que los modelos obtenidos por MCO + selección "clásica" de variables.

Podemos observar en la Gráfica 2 el efecto de reducción de coeficientes al variar la penalización λ, donde las estimaciones de los coeficientes *Ridge*, para un ejemplo de cáncer de próstata, están expresadas en función de los grados efectivos de libertad[5] implicados en cada penalización λ. Siendo d_j el autovalor[6] j de la matriz $\boldsymbol{X}$, tenemos

$$df(\lambda) = tr[\boldsymbol{X}(\boldsymbol{X}^T\boldsymbol{X} + \lambda\boldsymbol{I})^{-1}\boldsymbol{X}^T] = tr(\boldsymbol{H}_\lambda) = \sum_{j=1}^{p} \frac{d_j^2}{d_j^2 + \lambda}$$

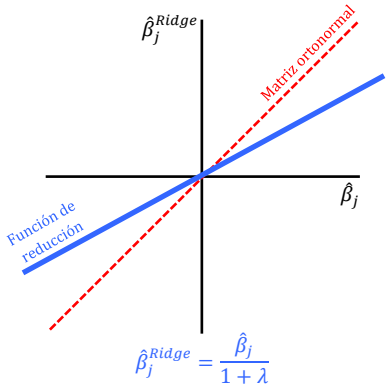


Gráfica 2. Coeficientes *Ridge* para un ejemplo de cáncer de próstata (Stamey et al., 1989) [01], en función de los grados efectivos de libertad $\boldsymbol{df(\lambda)}$. Línea roja vertical $\boldsymbol{df(\lambda) = 5.0}$ determinado por validación cruzada. Tomado de Friedman, et al. (2001), página 65 [03].

[5] Friedman et al. (2001) [04] denominan grados efectivos de libertad.

[6] En el caso de matriz ortonormal $d_j = 1,\ j = 1, \cdots, p$

Al aumentar λ (mayor penalización) los coeficientes estimados se contraen siendo esta contracción mayor en los valores altos y paulatinamente menor a medida que los coeficientes son más pequeños, como podemos observar en la Gráfica 3. Variando la pendiente de la función de reducción[7] actuando sobre λ podemos actuar sobre el grado de reducción o encogimiento de los coeficientes, a mayor valor de λ menor pendiente y en consecuencia mayor efecto en la reducción de coeficientes.



Gráfica 3. Función de Penalización *Ridge.*

Sin embargo, no conseguimos nuestro objetivo de selección de variables, especialmente necesario en los sistemas dispersos $(p \gg n)$ ya que ninguno de los coeficientes vale exactamente cero, por lo cual realmente no se produce la buscada selección de variables. Todas las variables originales permanecen en el modelo final.

[7] Al considerar ortonormalidad en $\boldsymbol{X}$, la función de contracción es $\boldsymbol{\beta}^{Ridge} = \frac{1}{1+\lambda}\boldsymbol{\beta}^{MCO}$

ESTIMADOR GARROTE NO NEGATIVO

Con el objetivo de encontrar un compromiso entre la simplicidad de obtener un modelo a través de la selección de variables, y la estabilidad y la precisión de la regresión *Ridge*, Breiman (1995) [01] propuso la técnica del Garrote No Negativo (GNN).

La idea fue minimizar respecto de $\boldsymbol{c} = (c_1, \cdots, c_p),\ t \geq 0$

$$\widehat{\boldsymbol{\beta}}^{GNN} = \underset{\boldsymbol{c}\in\mathbb{R}^p}{\operatorname{argmin}} \left\{ \sum_{i=1}^{n} \left(y_i - \sum_{j=1}^{p} c_j \hat{\beta}_j x_{ij} \right)^2 \right\}, \qquad sujeto\ a \quad c_j \geq 0 \quad y \quad \sum_{j=1}^{p} c_j \leq t$$

donde $\hat{\beta}_j$ son los estimadores obtenidos por MCO. Luego, los coeficientes estimados por GNN son:

$$\hat{\beta}_j^{GNN} = \hat{c}_j \hat{\beta}_j, \qquad j = 1, \cdots, p$$

Observamos que el estimador $\hat{\beta}^{GNN}$ depende de $\hat{\beta}^{MCO}$ y, por lo tanto, no está bien definido cuando $p \gg n$ (situación no muy común por entonces, (Tibshirani, 2011) [01]).

Posteriormente otros autores (Yuan y Lin 2007 [01], Zou 2006 [01]) han mostrado que GNN presenta interesantes propiedades cuando se usan otros estimadores iniciales tales como $\hat{\beta}^{Ridge}$, ya comentado, así como $\hat{\beta}^{LASSO}$ o $\hat{\beta}^{E\_Net}$ que veremos más adelante, lo que permiten extender el uso de GNN en problemas de altas dimensiones en sistemas dispersos.

Los valores de c_j son obtenidos resolviendo el problema de programación cuadrática y la naturaleza de las soluciones no negativas de garrote se puede ver cuando las columnas de $\boldsymbol{X}$ son ortogonales. Suponiendo que t está en el rango donde se puede satisfacer la restricción de igualdad $\|c\|_1 = t$, las soluciones tienen la forma explícita

$$\hat{c}_j = \left(1 - \frac{\lambda}{\hat{\beta}_j^2}\right)_+, \qquad j = 1, \cdots, p$$

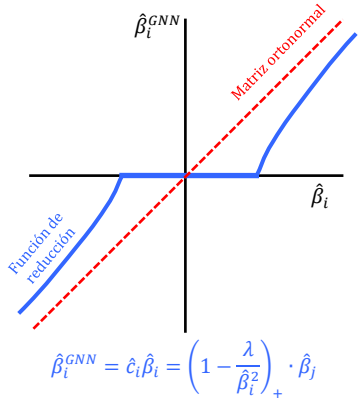
Por lo tanto

$$\hat{\beta}_j^{GNN} = \hat{c}_j \hat{\beta}_j = \left(1 - \frac{\lambda}{\hat{\beta}_j^2}\right)_+ \cdot \hat{\beta}_j$$

El parámetro de regularización λ es determinado por validación cruzada con el propósito de minimizar el error de predicción esperado (Breiman, 1995) [02].

A medida que aumenta λ, la mayoría de los c_j se hacen cero y los restantes $\hat{\beta}_j^{GNN}$ no nulos son contraídos hacia cero. Por lo tanto, la técnica del GNN elimina algunas variables, contrae otras y es relativamente estable (los resultados no cambian drásticamente con pequeñas modificaciones en los datos).

En la Gráfica 4 podemos observar el comportamiento de contracción y selección de GNN, los valores pequeños son contraídos en mayor medida en tanto que a los valores grandes les afecta menos la restricción de forma de forma paulatina, contrariamente a lo que sucede en *Ridge*. Asintóticamente no presentaría reducción.



Gráfica 4. Función de Penalización Garrote No Negativo.

El Garrote No Negativo es relativamente estable y es más preciso cuando hay pocos coeficientes no cero (Izenman, 2008) [02].

ESTIMADOR LASSO (*Least Absolute Shrinkage and Selection Operator*)

Basándose o inspirado en el planteamiento propuesto por Leo Breiman del Garrote No Negativo (Tibshirani, 2011) [02] y también motivado por el objetivo de encontrar una técnica de regresión lineal que fuera estable, pero que realizara selección de variables, Tibshirani (1996) [01] propuso LASSO (*Least Absolute Shrinkage and Selection Operator*).

LASSO es una técnica de regresión lineal regularizada que aplica una penalización, como *Ridge*, con una diferencia en la penalización (norma L_1 en lugar de L_2) que implica consecuencias importantes. LASSO resuelve el problema de mínimos cuadrados con restricción sobre la norma L_1 del vector de coeficientes:

$$\widehat{\boldsymbol{\beta}}^{LASSO} = \underset{\boldsymbol{\beta}\in\mathbb{R}^{\boldsymbol{p}}}{\operatorname{argmin}} \left\{ \sum_{i=1}^{n} \left(y_i - \sum_{j=1}^{p} \beta_j x_{ij} \right)^2 \right\}, \qquad sujeto\ a \quad \sum_{j=1}^{p} |\beta_j| \leq t$$

O en forma equivalente langrangiana, minimizando:

$$\widehat{\boldsymbol{\beta}}^{LASSO} = \underset{\boldsymbol{\beta}\in\mathbb{R}^{\boldsymbol{p}}}{\operatorname{argmin}} \left\{ \sum_{i=1}^{n} \left(y_i - \sum_{j=1}^{p} \beta_j x_{ij} \right)^2 + \lambda \sum_{j=1}^{p} |\beta_j| \right\}$$

siendo $t, \lambda \geq 0$ los respectivos parámetros de penalización por complejidad.

Que puede expresarse

$$\widehat{\boldsymbol{\beta}}^{LASSO} = \underset{\boldsymbol{\beta}\in\mathbb{R}^{\boldsymbol{p}}}{\operatorname{argmin}} \{ \|\boldsymbol{Y} - \boldsymbol{X}\boldsymbol{\beta}\|_2^2 + \lambda \|\boldsymbol{\beta}\|_1 \}$$

Siendo $\boldsymbol{Y} \to (n \times 1),\ \boldsymbol{X} \to (n \times p), \boldsymbol{\beta} \to (p \times 1)$

Asumimos que las variables están incorreladas $\boldsymbol{X}_i^T \boldsymbol{X}_j = \boldsymbol{0}$ para $i \neq j$ y consideramos los datos estandarizados $\boldsymbol{X}^T \boldsymbol{X} = \boldsymbol{I}_p$.

Si suponemos que $\widehat{\boldsymbol{\beta}}$ es solución y considerando que $\|\widehat{\boldsymbol{\beta}}\|_1 = |\widehat{\boldsymbol{\beta}}| = \widehat{\boldsymbol{\beta}} \cdot sign(\widehat{\boldsymbol{\beta}})$. Derivando respecto β e igualando a cero, tenemos

$$-2\boldsymbol{X}^T \left(\boldsymbol{Y} - \boldsymbol{X}\widehat{\boldsymbol{\beta}} \right) + \lambda sign(\widehat{\boldsymbol{\beta}}) = \boldsymbol{0}$$

$$\boldsymbol{X}^T(\boldsymbol{Y} - \boldsymbol{X}\widehat{\boldsymbol{\beta}}) = \frac{\lambda}{2}\, sign(\widehat{\boldsymbol{\beta}})$$

$$\boldsymbol{X}^T\boldsymbol{Y} - \boldsymbol{X}^T\boldsymbol{X}\widehat{\boldsymbol{\beta}} = \boldsymbol{X}^T\boldsymbol{Y} - \widehat{\boldsymbol{\beta}} = \frac{\lambda}{2}\, sign(\widehat{\boldsymbol{\beta}})$$

Por lo tanto,

$$\widehat{\boldsymbol{\beta}} = \boldsymbol{X}^T\boldsymbol{Y} - \frac{\lambda}{2}\, sign(\widehat{\boldsymbol{\beta}})$$

Obteniendo

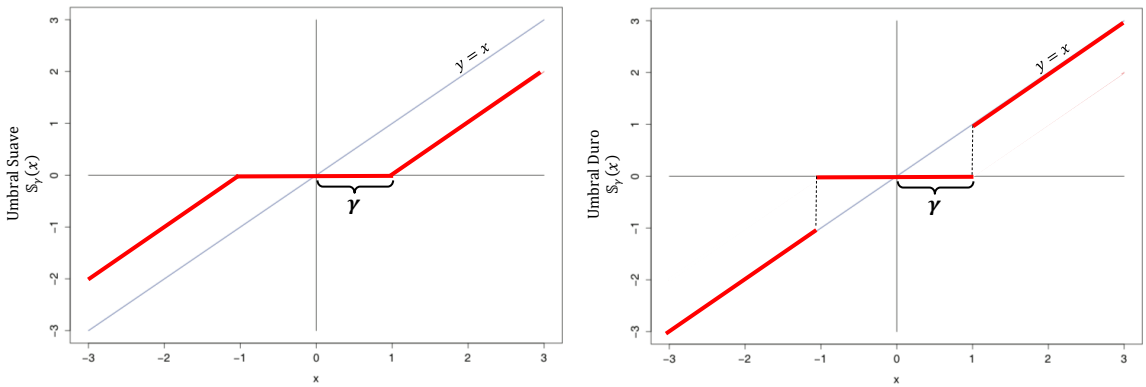
$$\hat{\beta}_j = \begin{cases} (\boldsymbol{X}^T\boldsymbol{Y})_j + \dfrac{\lambda}{2} & si\ \ (\boldsymbol{X}^T\boldsymbol{Y})_j < -\dfrac{\lambda}{2} \\ 0 & si\ \ -\dfrac{\lambda}{2} \le (\boldsymbol{X}^T\boldsymbol{Y})_j \le \dfrac{\lambda}{2} \\ (\boldsymbol{X}^T\boldsymbol{Y})_j - \dfrac{\lambda}{2} & si\ \ \dfrac{\lambda}{2} < (\boldsymbol{X}^T\boldsymbol{Y})_j \end{cases}$$

Este resultado puede expresarse de forma compacta mediante el uso del operador Umbral Suave (*Soft Thresholding*) también denominado Contracción *Wavelet*[8]. Ver Gráfica 5.

$$\mathbb{S}_\gamma(x) = \begin{cases} x + \gamma & si\ x < -\gamma \\ 0 & si - \gamma \le x < +\gamma \\ x - \gamma & si\ \gamma < x \end{cases}$$

que puede expresarse como[9]

$$\mathbb{S}_\gamma(x) = sign(x) \cdot (|x| - \gamma)_+$$



Gráfica 5. Contracción *Wavelet.*
Umbral Suave (*Soft Thresholding*) (izqu) y Umbral Duro (*Hard Thresholding*) (drch)

[8] El Umbral Suave es un caso particular de la Contracción *Wavelet*, la cual también comprende el Umbral Duro (*Hard Thresholding)* propio de los métodos de selección de variables.

[9] La función $sign(x)$ es igual al signo de su argumento (± 1); y la función $(x)_+$ es igual a la parte positiva de su argumento.

Considerando el valor de los coeficientes de Mínimos Cuadrados Ordinario (MCO) la aplicación de la penalización LASSO queda definida en el caso de $\boldsymbol{X}$ ortonormal, mediante

$$\hat{\beta}_j^{LASSO} = sign\left(\hat{\beta}_j^{MCO}\right) \cdot \left(\left|\hat{\beta}_j^{MCO}\right| - \gamma\right)$$

Siendo $\gamma = f(\lambda) = g(t)$

O bien

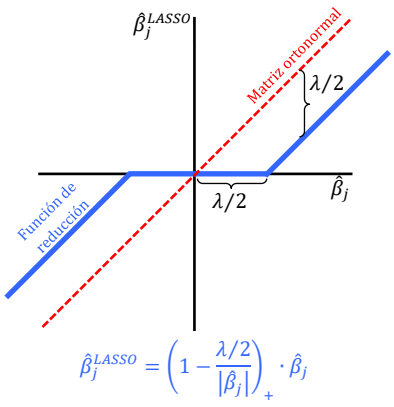
$$\hat{\beta}_j^{LASSO} = \left(1 - \frac{\lambda/2}{\left|\hat{\beta}_j^{MCO}\right|}\right)_+ \cdot \hat{\beta}_j^{MCO}$$

En forma similar a GNN y a diferencia de *Ridge* y MCO, el estimador de $\hat{\beta}^{LASSO}$ es no lineal en el vector de respuesta $\boldsymbol{Y}$.

Al aumentar la penalización λ los coeficientes estimados $\hat{\beta}_j$ se contraen de forma similar a GNN (Gráfica 4), es decir, los valores $\lambda/2 < \left|\hat{\beta}_j\right|$ estarán más sesgados hacia cero al aumentar λ (aunque en LASSO el sesgo es constante $\left(\left|\hat{\beta}_j\right| - \lambda\right)$ a diferencia de GNN que es mayor para valores de coeficientes próximos a λ y desaparece asintóticamente con λ).

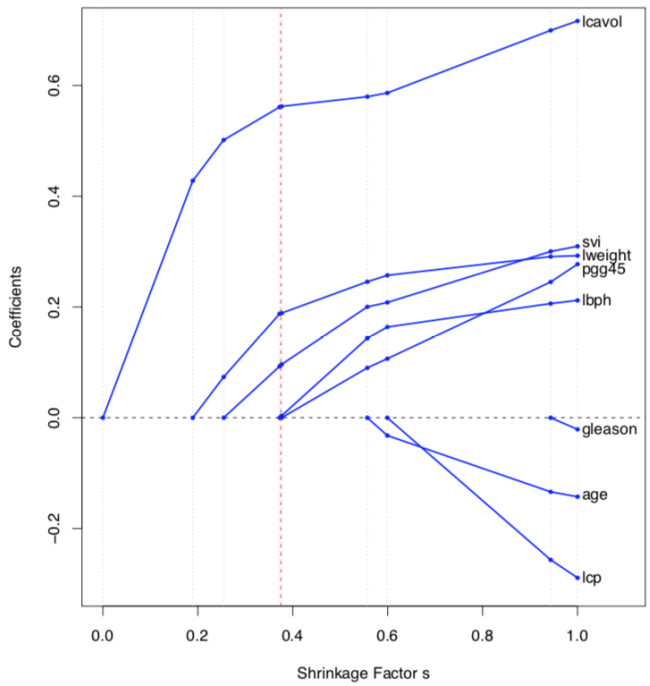
Los valores inferiores a la penalización $\lambda/2 \geq \left|\hat{\beta}_j\right|$ son "forzados" a cero al situarse por debajo del umbral.

La contracción es de valor constante $(\lambda/2)$ para penalizaciones pequeñas, y el aumento de la penalización iría sacando del modelo (llevando a cero los coeficientes) a aquellos en los que $\lambda/2 \geq \left|\hat{\beta}_j\right|$, observamos en la Gráfica 6 que coincide con el operador Umbral Suave.



Gráfica 6. Función de Penalización LASSO.

Para valores crecientes de λ o decrecientes de t, los coeficientes β_j se contraen hacia cero como en *Ridge*, con la diferencia de que algunos de ellos se anulan al alcanzar el umbral, anulación que no se da en *Ridge*. Podemos observar en la Gráfica 7 el efecto de reducción de coeficientes al variar la penalización λ, donde las estimaciones de los coeficientes LASSO, para el mismo ejemplo de cáncer de próstata mostrado en la Gráfica 2, están expresados en función de un factor de ajuste estandarizado $s = t/\sum_{j=1}^{p}|\hat{\beta}_j|$



Gráfica 7. Coeficientes *LASSO* para un ejemplo de cáncer de próstata (Stamey et al., 1989) [02], en función del factor de contracción $\boldsymbol{s}$. Línea roja vertical $\boldsymbol{s = 0.36}$ determinado por validación cruzada. Tomado de Friedman et al, (2001), página 70 [05].

La ventaja incorporada por LASSO es la posibilidad de controlar la selección de variables mediante la elección de λ para coeficientes pequeños y los coeficientes mayores son afectados de forma homogénea por una reducción constante igual al parámetro de penalización λ lo que, por otro lado, introduce un sesgo en la estimación.

Por lo tanto, conseguimos nuestro objetivo de selección de variables, especialmente útil y necesario en los sistemas dispersos $(p \gg n)$.

Una propiedad considerada como inconveniente es que con LASSO para $p \geq N$ el número de coeficientes distintos de cero que se puede obtener es como máximo n (Rosset el al., 2004) [01] y (Tibshirani, 2013) [01].

Meinshausen y Buhlmann (2006) [01] y Zhao y Yu (2006) [01] mostraron que LASSO es consistente[10] para la selección de variables, siempre que p no sea demasiado grande y el parámetro de penalización λ crezca más rápido que $\sqrt{nlog(p)}$. Específicamente, se permite que p sea tan grande como $e^{n\alpha}$ para $0 < \alpha < 1$ cuando los errores tienen colas gaussianas. Sin embargo, el valor de λ requerido para la consistencia de selección de variable reduce demasiado los coeficientes distintos de cero, lo que conduce a estimaciones sesgadas asintóticamente. Por consiguiente, LASSO es una selección de variable consistente bajo ciertas condiciones, pero no en general. Además, si LASSO es una selección de variable consistente, entonces no es eficiente para estimar los parámetros distintos de cero. Estos estudios confirman que LASSO no posee la propiedad oráculo (Fan y Li (2001) [01] y Fan y Peng (2004)) [01].

La propiedad oráculo de un método significa que puede seleccionar correctamente los coeficientes distintos de cero con probabilidad convergente a uno, y que los estimadores de los coeficientes distintos de cero son asintóticamente normales con las mismas medias y covarianza que tendrían si se conocieran los coeficientes cero por anticipado.

El principal inconveniente que presenta LASSO tiene una mayor relevancia en lo referente al aspecto computacional, que por otro lado recobra protagonismo desde el punto de vista de eficiencia computacional al tratarse de sistemas de alta dimensión y sistemas dispersos.

Los avances en los algoritmos para implementar regresión LASSO en forma eficiente han sido muy importantes ya que el método no es eficiente para un número grande de variables.

En sus comienzos, la estimación se realizaba resolviendo para cada valor de t el problema de programación cuadrática $\min_{\boldsymbol{\beta}} \left\{ \sum_{i=1}^{n} \left(y_i - \sum_{j=1}^{p} \beta_j x_{ij} \right)^2 \right\}, \ sujeto\ a\ \sum_{j=1}^{p} |\beta_j| \leq t.$

[10] Consistencia estadística significa que si el tamaño de la muestra aumenta las estimaciones convergen a valores verdaderos.

Posteriormente, surgieron los algoritmos[11] *Least Angle Regression* (LARS), (Efron et al., 2004) [01] y de camino de descenso coordinado (*pathwise coordinate decent methods*) (Friedman et al., 2010) [01] que permitieron reducir enormemente el costo computacional (Tibshirani, 2011) [03]. LARS es potencialmente revolucionario, ofrece modelos interpretables, estabilidad, predicciones precisas, resultados gráficos que muestran la compensación clave en la complejidad del modelo y una regla simple basada en datos para determinar el nivel óptimo de complejidad que casi evita el sesgo en las pruebas de hipótesis.

La implementación de LARS y los métodos de camino de descenso coordinado supusieron una mayor popularización de la aplicación de la norma L_1 como método de penalización para la selección de parámetros.

[11] No se abordan la descripción de estos algoritmos a pesar de la importancia relevante que tienen en los métodos de regularización, ya que como se ha indicado en los sistemas dispersos de alta dimensión las consideraciones estadísticas y computacionales están muy estrechamente vinculadas en su implementación.

ESTIMADOR *BRIDGE*

Antes de avanzar y describir las mejoras incorporadas, basándose en el principal método de regularización (LASSO), describimos el planteamiento generalizado de estos métodos.

Una formulación amplia de las técnicas de penalización/regularización puede plantearse de forma general en la denominada regresión penalizada de mínimos cuadrados:

$$\widehat{\boldsymbol{\beta}}_{\lambda} = \underset{\boldsymbol{\beta}\in\mathbb{R}^{\boldsymbol{p}}}{\operatorname{argmin}} \left\{ \sum_{i=1}^{n} \left(y_i - \sum_{j=1}^{p} \beta_j x_{ij} \right)^2 + \phi_{\lambda}(\boldsymbol{\beta}) \right\}$$

donde $\boldsymbol{\beta} = (\beta_1, \cdots, \beta_p), \geq 0$ y ϕ es una función de penalización sobre el "tamaño" de $\boldsymbol{\beta}$, en general de la forma $\phi_{\lambda}(\boldsymbol{\beta}) = \lambda \sum_{j=1}^{p} \phi_j(|\beta_j|)$ con ϕ_j creciente en $|\beta_j|$.

Una familia de funciones de penalización muy utilizada será, como hemos visto, la correspondiente a la norma-L_{γ}, con $\lambda \geq 0$, $\gamma \geq 0$, dada por:

$$\phi_{\lambda,\gamma}(\beta) = \lambda \|\beta\|_{\gamma}^{\gamma} = \begin{cases} \lambda \sum_{j=1}^{p} |\beta_j|^{\gamma} & si\ \gamma > 0 \\ \lambda \sum_{j=1}^{p} \mathbf{1}_{\{\beta_j \neq 0\}} & si\ \gamma = 0 \end{cases}$$

Con expresión lagrangiana

$$\widehat{\boldsymbol{\beta}}_{\lambda,\gamma} = \underset{\boldsymbol{\beta}\in\mathbb{R}^{\boldsymbol{p}}}{\operatorname{argmin}} \{ \|\boldsymbol{Y} - \boldsymbol{X}\boldsymbol{\beta}\|^2 + \lambda \|\boldsymbol{\beta}\|_{\gamma}^{\gamma} \} = \underset{\boldsymbol{\beta}\in\mathbb{R}^{\boldsymbol{p}}}{\operatorname{argmin}} \left\{ \sum_{i=1}^{n} \left(y_i - \sum_{j=1}^{p} \beta_j x_{ij} \right)^2 + \lambda \sum_{j=1}^{p} |\beta_j|^{\gamma} \right\}$$

Conocida ampliamente como regresión puente (*bridge regression*) o estimadores 'puente' con el origen del nombre atribuido a Frank y Friedman (1993) [01]. Aunque Izenman (2008) [03] señala que este nombre (*Bridge*) nunca aparece en esa referencia, aparentemente fue utilizado por primera vez por Friedman en una charla (Tibshirani, comunicación personal). Los estimadores resultantes $\widehat{\boldsymbol{\beta}}^{Bridge}$ en este caso son también conocidos como estimadores *Bridge* (Fu, 1998) [01].

Como podemos observar cuando $\gamma = 1$ tenemos el estimador LASSO y cuando $\gamma = 2$ obtenemos el estimador *Ridge*, de igual forma para $\gamma = 0$ nos situaríamos en un modelo de selección de variables. Como características en función del valor de la norma tenemos que:

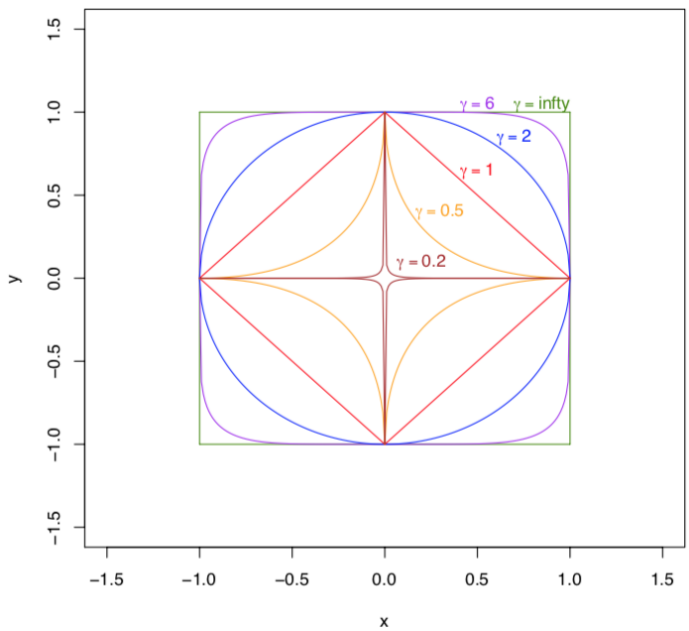
- Para $\gamma > 0$ el estimador *Bridge* existe
- Para $0 \leq \gamma < 1$ la penalización no es convexa y en consecuencia puede ser difícil obtener el estimador *Bridge* en alta dimensión. Los costes computacionales en esta situación son considerables.
- Para $\gamma = 1$ la penalización es convexa con una única solución (unicidad de las soluciones LASSO, Hastie et al. (2015)) [02].
- Para $\gamma > 1$ la penalización es estrictamente convexa y tiene una única solución, aunque casi seguramente los estimadores *Bridge* no serán cero, por lo que no ofrece la característica de selección de variables.
- Para $\gamma \geq 1$ existe correspondencia uno a uno entre el planteamiento lagrangiano

$$\widehat{\boldsymbol{\beta}}_{\lambda,\gamma} = \underset{\boldsymbol{\beta}\in\mathbb{R}^p}{\operatorname{argmin}}\left\{\|\boldsymbol{Y} - \boldsymbol{X}\boldsymbol{\beta}\|^2 + \lambda\|\boldsymbol{\beta}\|_\gamma^\gamma\right\}$$

y el problema restringido o condicionando

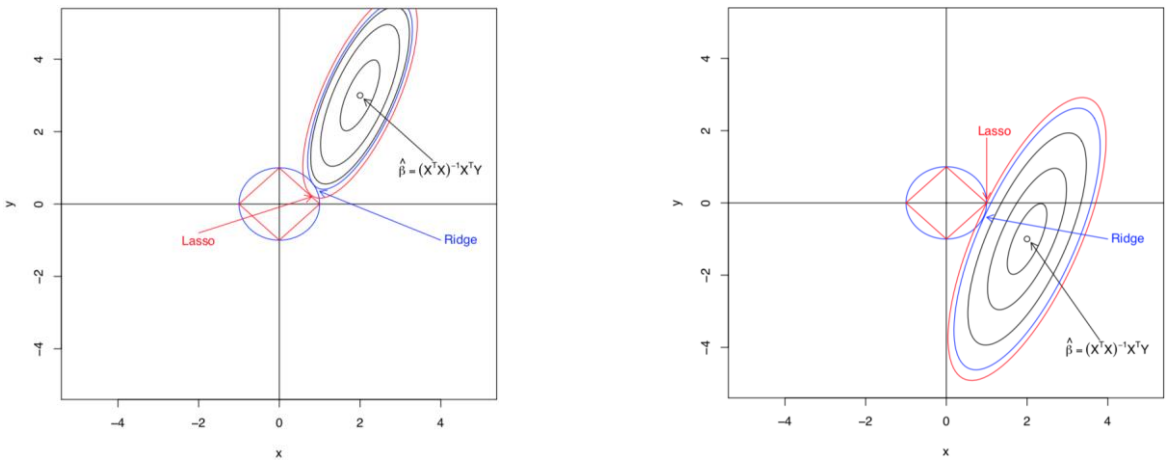
$$\widehat{\boldsymbol{\beta}}_{\lambda,\gamma} = \underset{\boldsymbol{\beta}\in\mathbb{R}^p}{\operatorname{argmin}}\{\|\boldsymbol{Y} - \boldsymbol{X}\boldsymbol{\beta}\|^2\} \quad sujeto\ a\ \ \|\boldsymbol{\beta}\|_\gamma^\gamma \leq t$$

En la Gráfica 8 podemos observar la forma de las regiones de penalización en función de γ



Gráfica 8. Regiones de penalización $\|\boldsymbol{\beta}\|_\gamma^\gamma$ para diferentes valores de $\boldsymbol{\gamma}$. Las regiones son convexas solo para $\boldsymbol{\gamma \geq 1}$.

Partiendo de la observación de las regiones de penalización es más fácil observar la mayor potencialidad que LASSO presenta respecto de *Ridge* en cuanto a la selección de variables en sistemas dispersos, es decir, la capacidad de anulación de coeficientes, aunque siempre dependiendo de los valores de $\widehat{\boldsymbol{\beta}}^{MCO}$, en la Gráfica 9 se observa claramente el caso de $p = 2$.



Gráfica 9. Intersección en regiones de penalización LASSO y *Ridge.*

Para $q > 1$ el estimador no realiza selección de variables (Fan y Li, 2001) [02]. Por otro lado, LASSO corresponde al valor de q más pequeño que produce una región factible convexa.

La convexidad del problema de optimización es deseable desde el punto de vista computacional. Funciones en varias variables no convexas pueden tener múltiples óptimos locales, y si bien las regiones de penalización no convexa contribuyen en mayor medida a la selección de variables, aspecto prioritario en los sistemas dispersos de alta dimensión, la complejidad y menor rendimiento computacional son factores que condicionan muy severamente la aplicación de penalizaciones no convexas $0 < \gamma < 1$

En los últimos años se han presentado algunas generalizaciones y extensiones de las técnicas presentadas anteriormente, especialmente diseñadas para ciertas situaciones particulares.

El auge en los últimos años en la investigación y aplicación de técnicas tipo LASSO se debe principalmente a la existencia de problemas donde $p \gg n$ y al desarrollo paralelo de algoritmos eficientes (Tibshirani, 2011) [04].

Todas ellas buscan retener las ventajas de LASSO como método de estimación y selección de variables, y al mismo tiempo corregir algunas de sus posibles desventajas.

A continuación, se presentan algunas de las más importantes.

MEJORAS EN LA REGULARIZACIÓN LASSO

Hemos visto que LASSO presenta la propiedad principal en cuanto a que facilita la selección de variables, siendo el método de regularización que potencia la obtención de modelos dispersos, sin embargo, también se han descrito inconvenientes o deficiencias, como el comportamiento ante la colinealidad y el condicionante respecto al número de variables seleccionadas en el caso de sistemas de alta dimensión $p \gg n$ que queda limitado a n.

Seguidamente se describen técnicas que buscan retener las ventajas de LASSO como método de estimación y selección de variables, y al mismo tiempo corregir algunas de sus posibles desventajas.

A continuación, se presentan algunas de las más importantes, como es *Elastic Net* que corrige el deficiente comportamiento frente a la colinealidad de LASSO o LASSO-Adaptativo y LASSO-Relajado que palían los inconvenientes respecto la consistencia de LASSO.

ESTIMADOR *ELASTIC NET*

En la práctica, los predictores son diferentes, pero pueden estar fuertemente correlacionados.

Si bien hemos visto que LASSO permite una ventaja necesaria sobre *Ridge* especialmente en sistemas dispersos $p \gg n$ donde es necesaria la consiguiente selección de variables, aparecen unos inconvenientes, especialmente en sistemas dispersos, que presenta LASSO y que hace conveniente las características de *Ridge* en cuanto su comportamiento respecto de la correlación de variables.

Para este tipo de situación en sistema dispersos de alta dimensión $p \gg n$ y con variables agrupadas, LASSO no es el método ideal, porque solo puede seleccionar a lo sumo n variables de p candidatos (Rosset el al., 2004) [02] y (Tibshirani, 2013) [02], y carece de la capacidad de detectar el agrupamiento de información. En cuanto al rendimiento de predicción, si hay correlación entre los predictores, *Ridge* presenta mejor rendimiento que LASSO siendo esta situación frecuente en sistemas dispersos.

En caso de alta correlación entre dos predictores la estimación de LASSO puede ocultar la relevancia de uno de ellos, simplemente porque está altamente correlacionado con otro. Los coeficientes de dos predictores correlacionados deben estar cercanos.

En la parte izquierda de la Gráfica 10 tenemos un conjunto de seis variables altamente correlacionadas en grupos de tres, podemos observar claramente la ruta de coeficientes donde el comportamiento es algo "errático" a pesar de ser dos conjuntos de variables con correlaciones por pares alrededor del 0,97.

Zou y Hastie, (2005a, 2005b) [01] [01], propusieron *Elastic Net* como un método de regularización el cual combina los beneficios de *Ridge* y LASSO obteniendo un compromiso entre las penalizaciones de *Ridge* y LASSO: dados $\lambda_1 > 0$ y $\lambda_2 > 0$, ambos parámetros de complejidad. Partiendo del estimador *Naïve Elastic Net*

$$\widehat{\boldsymbol{\beta}}_{\lambda_1,\lambda_2}^{Naïve\_E\_NET} = \underset{\boldsymbol{\beta}\in\mathbb{R}^p}{\operatorname{argmin}}\{\|\boldsymbol{Y}-\boldsymbol{X}\boldsymbol{\beta}\|^2 + \lambda_1\|\boldsymbol{\beta}\|_1 + \lambda_2\|\boldsymbol{\beta}\|_2^2\}$$

$$\widehat{\boldsymbol{\beta}}^{Naïve\_E\_NET}_{\lambda_1,\lambda_2} = \underset{\boldsymbol{\beta}\in\mathbb{R}^p}{\operatorname{argmin}} \left\{ \sum_{i=1}^{n} \left(y_i - \sum_{j=1}^{p} \beta_j x_{ij} \right)^2 + \lambda_1 \sum_{j=1}^{p} |\beta_j| + \lambda_2 \sum_{j=1}^{p} \beta_j^2 \right\}$$

Si tomamos $\alpha = \frac{\lambda_2}{\lambda_1+\lambda_2}$

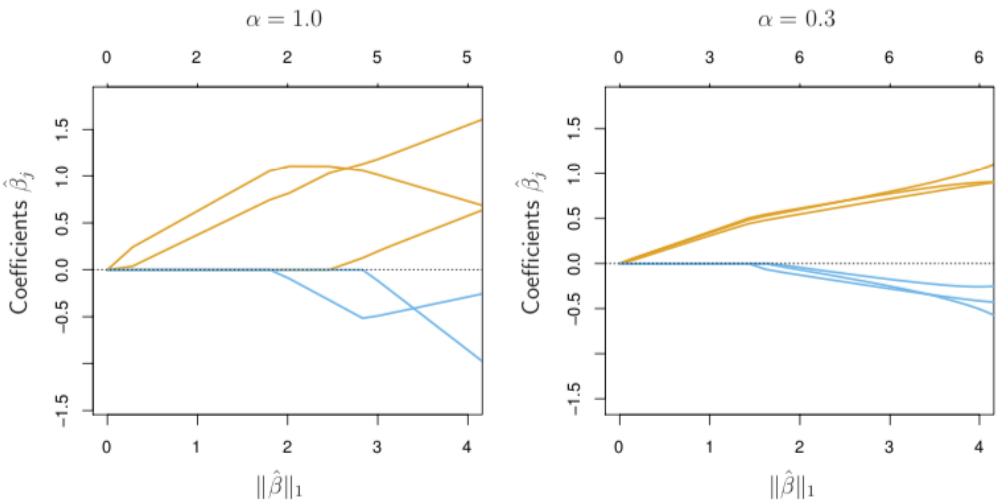
$$\widehat{\boldsymbol{\beta}}^{Naïve\_E\_NET}_{\lambda_1,\lambda_2} = \underset{\boldsymbol{\beta}\in\mathbb{R}^p}{\operatorname{argmin}} \left\{ \sum_{i=1}^{n} \left(y_i - \sum_{j=1}^{p} \beta_j x_{ij} \right)^2 + (1-\alpha) \sum_{j=1}^{p} |\beta_j| + \alpha \sum_{j=1}^{p} \beta_j^2 \right\}$$

o bien

$$\widehat{\boldsymbol{\beta}}^{Naïve\_E\_NET}_{\lambda_1,\lambda_2} = \underset{\boldsymbol{\beta}\in\mathbb{R}^p}{\operatorname{argmin}} \{ \|\boldsymbol{Y} - \boldsymbol{X}\boldsymbol{\beta}\|^2 \} \qquad sujeto\ a \quad (1-\alpha)\|\boldsymbol{\beta}\|_1 + \alpha\|\boldsymbol{\beta}\|_2^2 \leq t$$

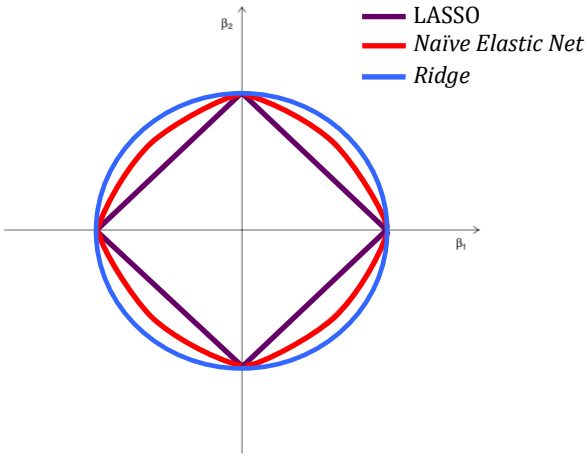
Este criterio es estrictamente convergente, por lo tanto, existe un mínimo único.

El estimador de *Naïve Elastic Net* combina las fortalezas de LASSO (la penalización L_1 promueve soluciones dispersas, es decir, obtenemos selección de variables), y de *Ridge* (predictores altamente correlacionados presentan coeficientes estimados similares). Como podemos observar en la parte derecha de la Gráfica 10, la aplicación de *Naïve Elastic Net* permite la presencia de todas las variables y la clara agrupación de los grupos correlacionados frente a las trayectorias dispersas de LASSO en esta situación de alta correlación.



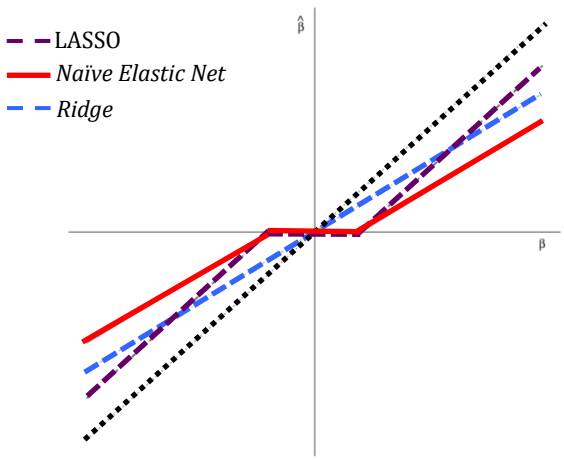
Gráfica 10. Seis variables altamente correlacionadas (0,95) en grupos de tres. Itinerario de los coeficientes LASSO (izqu) e itinerario de los coeficientes *Elastic Net.* (drch).
Tomado de Hastie, et al. (2015), página 56 [03].

Si observamos la forma del área de penalización en $p = 2$ vemos que la resolución será equivalente a resolver un problema de optimización tipo LASSO. Como vemos en la Gráfica 11 el área de penalización es convexa, (convexidad que vendrá en función de α), y podemos ver también que los vértices en los ejes no son derivables ya que considera la componente del valor absoluto, lo que le confiere una mayor capacidad de selección de variables. Es conveniente señalar que los contornos de la región de penalización para penalización *Bridge* con $1 < \gamma < 2$ son similares, salvo en la importante característica de que *Naïve Elastic Net* presenta los vértices no diferenciables.



Gráfica 11. Regiones de penalización
LASSO, *Elastic Net* y *Ridge*.

En cuanto a la función de penalización podemos observar en la Gráfica 12 que *Naïve Elastic Net* supone una unión de las características de LASSO y *Ridge*. *Naïve Elastic Net* mantiene la característica de selección *Soft Thresholding* propia de LASSO y para valores superiores a $\lambda_1/2$, la pendiente de la recta de penalización definida por *Ridge* vendrá determinada en función de λ_2.



Gráfica 12. Función de penalización.
LASSO, *Naïve Elastic Net* y *Ridge*.

La función de penalización correspondiente a un diseño ortogonal $\boldsymbol{X}^T\boldsymbol{X} = \boldsymbol{I}$ es la siguiente

$$\hat{\beta}_j^{Naïve\_E\_Net} = \frac{1}{1+\lambda_2}\left(1 - \frac{\lambda_1/2}{|\hat{\beta}_j|}\right)_+ \cdot \hat{\beta}_j$$

donde igualmente se puede verificar que se trata de la composición de los dos tipos de penalizaciones (LASSO y *Ridge*).

Como método de selección automática de variables, *Naïve Elastic Net* supera las limitaciones de LASSO; cuando $p > n$ restringe a n el número máximos de coeficientes que se pueden obtener, así como cuando existen grupos de variables altamente correlacionadas. Se ha observado empíricamente que para situaciones usuales $n > p$, cuando existe alta correlación entre las variables, el rendimiento de predicción de *Ridge* supera al de LASSO. Sin embargo, la evidencia empírica también muestra que *Naïve Elastic Net* no funciona satisfactoriamente a menos que esté muy cerca de la regresión *Ridge* o LASSO. Por eso Zou y Hastie (2005a) [02] lo denominaron *Naïve* (ingenuo).

En la configuración de predicción de regresión, un método de penalización preciso logra un buen desempeño de predicción a través del equilibrio sesgo-varianza. El estimador *Naïve Elastic Net* es un procedimiento de dos etapas: para cada λ_2 fijo primero encontramos los coeficientes de regresión de *Ridge*, y luego hacemos la contracción de tipo LASSO a lo largo de las rutas de solución de coeficiente de LASSO. Por lo tanto, parece incurrir en una doble cantidad de contracción. La doble contracción no ayuda a reducir mucho las varianzas e introduce un sesgo adicional innecesario, en comparación con la contracción pura de LASSO o *Ridge*.

Para corregir esta situación, si consideramos un conjunto de datos $(\boldsymbol{X}, \boldsymbol{y})$, los parámetros de penalización λ_1 y λ_2; y los datos aumentados $(\boldsymbol{X}^*, \boldsymbol{y}^*)$ tales que

$$\boldsymbol{X}^*_{(n+p)\times p} = (1+\lambda_2)^{-1/2}\begin{pmatrix}\boldsymbol{X}\\ \sqrt{\lambda_2}\cdot\boldsymbol{I}\end{pmatrix}, \qquad \boldsymbol{y}^*_{(n+p)} = \begin{pmatrix}\boldsymbol{y}\\ \boldsymbol{0}\end{pmatrix}$$

Si tomamos $\gamma = \frac{\lambda_1}{\sqrt{(1+\lambda_2)}}$ y $\boldsymbol{\beta}^* = \sqrt{(1+\lambda_2)} \cdot \boldsymbol{\beta}$. Entonces el criterio *Naïve Elastic Net* puede escribirse como

$$L(\gamma, \boldsymbol{\beta}) = L(\gamma, \boldsymbol{\beta}^*) = |\boldsymbol{Y}^* - \boldsymbol{X}^*\boldsymbol{\beta}^*|^2 + \gamma|\boldsymbol{\beta}^*|_1$$

Tomando

$$\widehat{\boldsymbol{\beta}}^* = \underset{\boldsymbol{\beta} \in \mathbb{R}^p}{\operatorname{argmin}} L\{(\gamma, \boldsymbol{\beta}^*)\}$$

Entonces

$$\widehat{\boldsymbol{\beta}} = \frac{1}{\sqrt{(1+\lambda_2)}} \widehat{\boldsymbol{\beta}}^*$$

Naïve Elastic Net resuelve un problema tipo LASSO

$$\widehat{\boldsymbol{\beta}}^* = \underset{\boldsymbol{\beta}^* \in \mathbb{R}^p}{\operatorname{argmin}} \left\{ |\boldsymbol{Y}^* - \boldsymbol{X}^*\boldsymbol{\beta}^*|^2 + \frac{\lambda_1}{\sqrt{(1+\lambda_2)}} |\boldsymbol{\beta}^*|_1 \right\}$$

Elastic Net (corregido) $\hat{\beta}$ está definido por $\widehat{\boldsymbol{\beta}}^{E\_Net} = \sqrt{(1+\lambda_2)} \cdot \widehat{\boldsymbol{\beta}}^*$

Por lo tanto

$$\widehat{\boldsymbol{\beta}}^{E\_Net} = (1+\lambda_2)\widehat{\boldsymbol{\beta}}^{Naïve\_E\_Net}$$

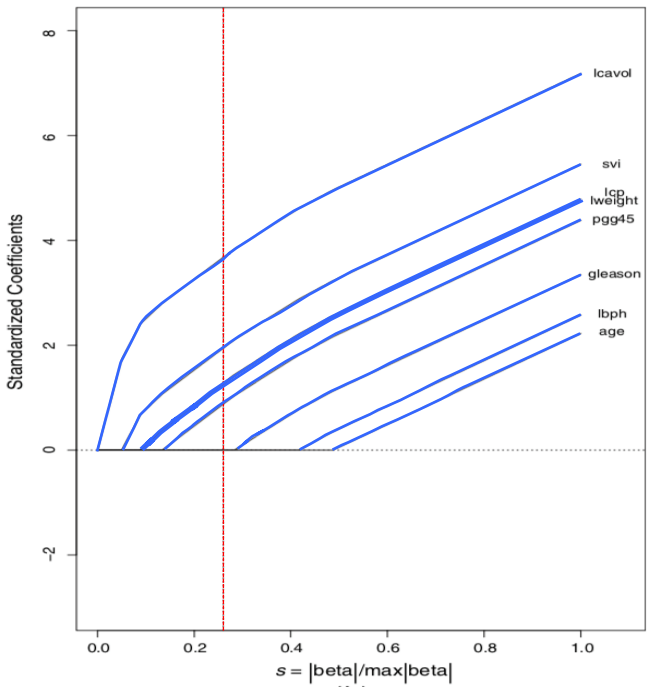
El coeficiente *Elastic Net* es un coeficiente *Naïve Elastic Net* reescalado. Tal transformación de escala preserva la propiedad de selección variable de *Naïve Elastic Net* y es la forma más simple de deshacer la contracción. En consecuencia, todas las buenas propiedades de *Naïve Elastic Net* se mantienen en *Elastic Net*.

Empíricamente la *Elastic Net* funciona muy bien en comparación con la regresión de LASSO y *Ridge*.

Por consiguiente, debido a que la doble penalización puede introducir sesgo en la estimación, se corrige el estimador obteniéndose $\widehat{\boldsymbol{\beta}}^{E\_NET} = (1+\lambda_2)\widehat{\boldsymbol{\beta}}^{Naïve\_E\_NET}$ (Zou y Hastie, 2005a) [03].

Con objeto también de mejorar el rendimiento computacional existen algoritmos eficientes del tipo LARS (LARS-EN) y de coordenada descendente, para su implementación (Friedman et al., 2010) [02].

En la Gráfica 13 podemos observar la aplicación del *Elastic Net* al mismo conjunto de datos para cáncer de próstata (visto en la Gráfica 2 con aplicación *Ridge* y en el Gráfica 7 con aplicación LASSO. *Elastic Net* con selección optima de la penalización por validación cruzada (línea roja) selecciona 5 variables, (similar a LASSO) y presenta un comportamiento de umbral suave propio de LASSO.



Gráfica 13. Coeficientes *Elastis Net* para un ejemplo de cáncer de próstata (Stamey et al., 1989) [03], en función del factor de contracción s. Línea roja vertical $s = 0.26$ determinado por validación cruzada.
Tomado de Zou y Hastie, (2005a), página 331 [04].

El anterior ejemplo sometido a comparación entre los diferentes modelos presenta para *Elastic Net* el menor error cuadrático medio y una selección de variables ligeramente diferente a LASSO, ver Tabla 1.

Zou y Hastie (2005a) [05] señalan, en este caso, a *Elastic Net* como el método campeón entre los competidores en términos de precisión y dispersión (*sparsity*). Hay que señalar que *Naïve Elastic Net* en este ejemplo falla en la selección de variables. Los datos del ejemplo presentan correlaciones[12] lo cual perjudica a LASSO y es corregido por *Elastic Net*.

A modo de comparación Zou y Hastie (2005a) [06] indican que "Siempre que la regresión *Ridge* mejore a MCO, *Elastic Net* mejorará a LASSO"[13]

Método	Parámetro(s)	Test Error cuadrático medio	Variables seleccionadas
MCO		0.586 (0.184)	Todas
Ridge regressión	$\lambda = 1$	0.566 (0.188)	Todas
LASSO	$s = 0.39$	0.499 (0.161)	(1,2,4,5,8)
Naïve Elastic Net	$\lambda = 1, s = 1$	0.566 (0.188)	Todas
Elastic Net	$\lambda = 1000, s = 0.26$	0.381 (0.105)	(1,2,5,6,8)

Tabla 1. Datos del cáncer de próstata. Comparación de modelos.
Tomado de Zou y Hastie (2005a), página 311 [07].

[12] Algunas incluso superiores a 0.76 como son las variables pgg45 y gleason.
[13] Traducción propia

ESTIMADOR LASSO-DAPTATIVO (*ADAPTIVE* LASSO)

Vimos anteriormente en la descripción de LASSO que para la selección de variable es consistente bajo ciertas condiciones, pero no en general. Además, si LASSO es una selección de variable consistente, entonces no es eficiente para estimar los parámetros distintos de cero Meinshausen y Buhlmann (2006) [02] y Zhao y Yu (2006) [02]. Así como que LASSO no posee la propiedad del oráculo (Fan y Li (2001) [03] y Fan y Peng (2004)) [02].

Zou, (2006) [02] propuso una versión adaptativa de LASSO motivada en el hecho de que bajo ciertas condiciones el estimador de LASSO no siempre es consistente[14] como método de selección de variables.

LASSO-Adaptativo es una generalización de LASSO que permite aplicar diferentes penalizaciones a las variables mediante la asignación de pesos distintos, los cuales dependen de los datos.

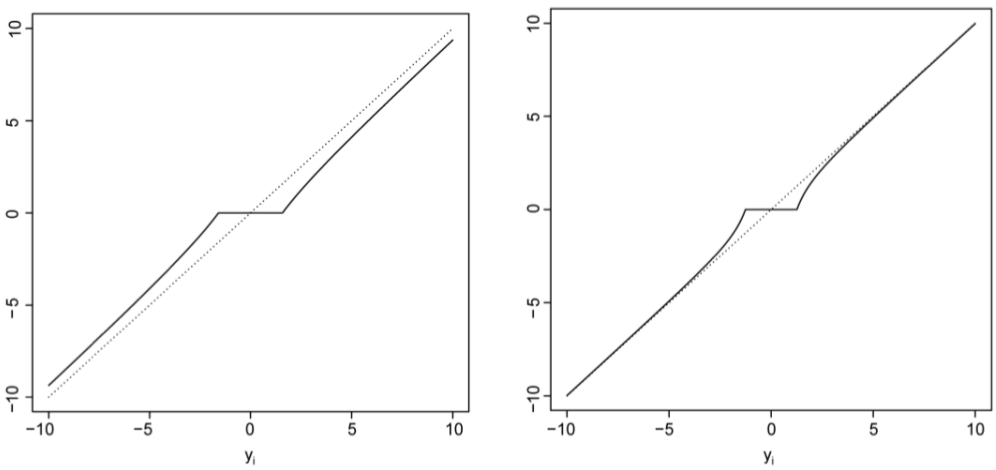
En LASSO-Adaptativo, el problema consiste en minimizar respecto de $\boldsymbol{\beta}$ la expresión:

$$\boldsymbol{\beta}^{LASSO_{Adaptativo}} = \underset{\boldsymbol{\beta}\in\mathbb{R}^n}{\operatorname{argmin}} \left\{ \sum_{i=1}^{n} \left(y_i - \sum_{j=1}^{p} \beta_j \, x_{ij} \right)^2 + \lambda \sum_{j=1}^{p} w_j |\beta_j| \right\}$$

donde $w_j = 1/|\tilde{\beta}_j|^{\gamma}$, $j = 1, \cdots, p$ son pesos positivos que aseguran propiedades de consistencia del estimador LASSO-Adaptativo, $\gamma > 0$ es un parámetro de ajuste adicional y $\tilde{\beta}_j$ es un estimador inicial de β_j (Zou, 2006) [03]. La penalización LASSO-Adaptativo puede ser vista como una aproximación a la penalización de norma L_q con $q = 1 - \gamma$.

En este sentido, si observamos la función de penalización de LASSO-Adaptativo en la Gráfica 14 comprobamos que el umbral suave (*Soft Thresholding*) es muy similar a la función de penalización de Garrote No Negativo (Gráfica 4).

[14] Consistencia estadística significa que si el tamaño de la muestra aumenta las estimaciones convergen a valores verdaderos



Gráfica 14. Función de penalización LASSO-Adaptativo con $\gamma = 0.5$ (izqu) y $\gamma = 0.2$ (drch). Tomado de Zou, (2006), página 1421 [04].

El LASSO-Adaptativo es un método de dos pasos. En el primer paso, se obtiene un estimador inicial, que puede ser generado mediante la resolución de un problema de optimización aplicando cualquier tipo de penalización, MCO, *Ridge* e incluso el propio LASSO (Zou, 2006) [05].

Si $p < n$ se pueden usar las soluciones de mínimos cuadrados como las estimaciones iniciales $\tilde{\beta}_j$. Cuando $p \geq n$, las estimaciones de mínimos cuadrados no están definidas, pero bajo la ortogonalidad parcial y ciertas otras condiciones, el LASSO-Adaptativo logra la consistencia de selección y la eficiencia de la estimación cuando los estimadores de regresión marginal se usan como estimadores iniciales, aunque no producen una estimación consistente de los parámetros (Huang et al., 2008) [01].

Una ventaja de LASSO-Adaptativo es que, dadas las estimaciones iniciales, el criterio es convexo en β. Además, si las estimaciones iniciales son $\sqrt{n}$ consistentes, Zou (2006) [06] mostró que el método recupera el modelo verdadero en condiciones más generales que LASSO (Hastie et al., 2015) [04].

Al permitir una penalización relativamente más alta para coeficientes cero y una penalización más baja para coeficientes distintos de cero, el LASSO-Adaptativo espera reducir el sesgo de estimación y mejorar la precisión de la selección variable, en comparación con el LASSO (Huang et al., 2008) [02].

El LASSO-Adaptativo, como hemos comentado, tiene la propiedad de oráculo que no presenta el LASSO, propiedad presente en los métodos no convexos como SCAD y *Bridge* (en $0 < \gamma < 1$), pero estos métodos ofrecen una limitación importante en cuanto a eficiencia computacional. Por consiguiente la ventaja que presenta LASSO-Adaptativo es, teniendo la propiedad del oráculo al ser un método convexo, un costo computacional bastante menor que los métodos no convexos, de hecho toda la trayectoria de regularización del LASSO-Adaptativo se puede calcular con la misma complejidad computacional que la solución de mínimos cuadrados utilizando el algoritmo LARS (Efron et al., 2004) [02].

El LASSO-Adaptativo es un método útil para analizar datos de alta dimensión.

ESTIMADOR LASSO-RELAJADO (*RELAXED* LASSO)

Como hemos comentado la contracción del LASSO provoca que las estimaciones de los coeficientes distintos de cero estén sesgadas hacia cero, y en general no son consistentes. Un enfoque para reducir este sesgo es ejecutar el LASSO para identificar el conjunto de coeficientes distintos de cero, y luego ajustar un modelo lineal restringido al conjunto de características seleccionado. Esto no siempre es factible si el conjunto seleccionado es grande.

Alternativamente, se puede usar el LASSO para seleccionar el conjunto de predictores que no son cero, y luego aplicar el LASSO nuevamente, pero usando solo los predictores seleccionados del primer paso. Esto se conoce como el LASSO-Relajado (Meinshausen, 2007) [01].

Se trata de una generalización de la estimación del LASSO. La principal motivación son los problemas de regresión de muy alta dimensión, donde el LASSO tiene dos deficiencias:

- Selección de variables de ruido: si el parámetro de penalización se elige mediante validación cruzada, el número de variables seleccionadas suele ser muy grande. Muchas variables de ruido son potencialmente seleccionadas.

- Baja precisión de las predicciones: La precisión de la predicción (en términos de pérdida de error cuadrático) está afectada negativamente por la presencia de muchas variables de ruido, en particular para las relaciones altas de señal-ruido.

Las ventajas de LASSO-Relajado sobre LASSO en este entorno de alta dimensión son dobles.

- Estimaciones más dispersas: el número de coeficientes seleccionados es, en general, mucho más pequeño para LASSO-Relajado, sin comprometer la precisión de las predicciones. Los modelos producidos por LASSO-Relajado son más susceptibles de interpretación.

- Predicciones más precisas: si la relación señal-ruido es muy baja, la precisión predictiva de LASSO y LASSO-Relajado es comparable. Para una alta relación señal-ruido, LASSO-Relajado logra a menudo predicciones mucho más precisas.

Desde un punto de vista computacional, para altas relaciones de señal-ruido, ambas ventajas de LASSO-Relajado, estimaciones más dispersas y predicciones más precisas, se pueden lograr alternativamente mediante el uso del híbrido LARS-OLS. Sin embargo, el híbrido LARS-OLS no se adapta siempre a la relación señal / ruido y tiene un rendimiento mucho peor que el LASSO para las relaciones señal / ruido bajas. LASSO-Relajado se adapta a la relación señal-ruido y logra un rendimiento casi óptimo en una amplia variedad de conjuntos de datos.

La idea es utilizar la validación cruzada para estimar el parámetro de penalización inicial para LASSO, y luego nuevamente para un segundo parámetro de penalización aplicado al conjunto seleccionado de predictores.

Es un procedimiento en dos etapas, propuesto como una generalización de LASSO y especialmente diseñado para problemas de regresión en altas dimensiones (Meinshausen, 2007) [02].

En una primera etapa y para $\lambda > 0$ fijo, se aplica LASSO sobre el modelo completo y se define la función indicador $\mathbf{1}_{\mathcal{M}_\lambda}$ sobre el conjunto de variables $\mathcal{M}_\lambda \subseteq \{1, 2, \cdots, p\}$ con $\mathcal{M}_\lambda = \left\{1 \leq k \leq p \middle| \hat{\beta}_{\lambda,k}^{LASSO} \neq 0\right\}$ tal que para todo $k \in \{1, 2, \cdots, p\}$

$$\left\{\hat{\beta}_{\lambda,k}^{LASSO} \cdot \mathbf{1}_{\mathcal{M}_\lambda}\right\}_k = \begin{cases} 0 & si \quad k \notin \mathcal{M}_\lambda \\ \beta_k & si \quad k \in \mathcal{M}_\lambda \end{cases}$$

Luego, el estimador de LASSO-Relajado, $\hat{\beta}_{\lambda,\phi}^{LASSO_{Relajado}}$, se define para $\lambda \in [0, \infty)$ $\phi \in (0,1]$ como:

$$\hat{\beta}_{\lambda,\phi}^{LASSO_{Relajado}} = \underset{\boldsymbol{\beta} \in \mathbb{R}^p}{\operatorname{argmin}} \left\{ \frac{1}{n} \left(\sum_{i=1}^{n} y_i - X_i^T \left\{ \beta \cdot \mathbf{1}_{\mathcal{M}_\lambda} \right\} \right)^2 + \phi\lambda \|\beta\|_1 \right\}$$

El parámetro λ regula la parte de selección de variables como en LASSO, mientras que el parámetro de relajación ϕ controla la contracción de los coeficientes.

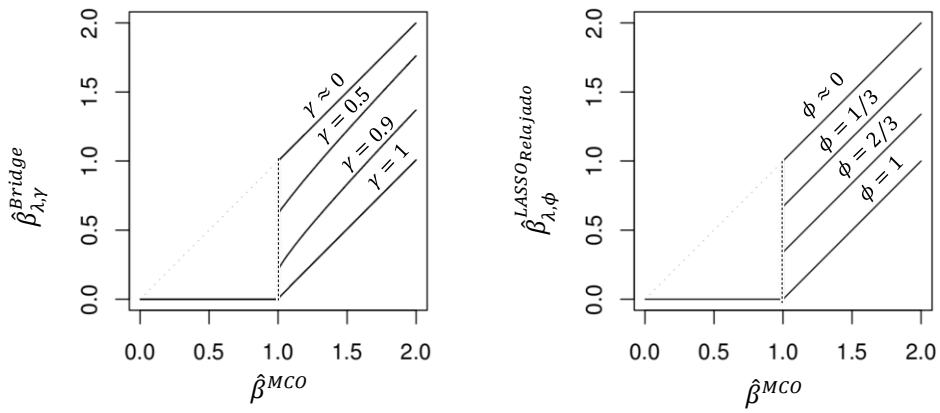
Si $\phi = 1$ la estimación coincide con LASSO mientras que si $\phi < 1$ la contracción de los coeficientes en el modelo seleccionado es menor que en LASSO. Los parámetros λ y ϕ pueden ser elegidos por validación cruzada.

La estimación de los coeficientes se puede obtener en forma eficiente a través de un algoritmo basado en LARS (Meinshausen, 2007) [03].

Si $\boldsymbol{X}$ es ortogonal, la función de penalización es

$$\hat{\beta}^{LASSO_{Relajado}}_{\lambda,\phi,k} = \begin{cases} \hat{\beta}^{MCO}_k - \phi\lambda & si \quad \hat{\beta}^{MCO}_k > \lambda \\ 0 & si \quad -\lambda \leq \hat{\beta}^{MCO}_k \leq \lambda \\ \hat{\beta}^{MCO}_k + \phi\lambda & si \quad \hat{\beta}^{MCO}_k < -\lambda \end{cases}$$

Si observamos la función de penalización LASSO-Relajado en comparación con la función de penalización *Bridge* para $0 < \gamma < 1$ en la Gráfica 15 podemos ver que son bastante similares en tanto que, para valores en los que $\gamma \to 0$, así como para valores en los que $\phi \to 0$, los coeficientes se aproximan a los coeficientes sin penalización, es decir, selección de variables con el consiguiente umbral duro (*Hard Thresholding*). Y de igual forma en el caso de $\gamma = 1$ o $\phi = 1$ tenemos la penalización LASSO. En ambas, la variación de γ y ϕ permiten controlar el sesgo que introduce en los coeficientes presentes en el modelo la penalización LASSO.



Gráfica 15. Funciones de penalización. *Bridge* (izqu) y LASSO-Relajado (drch). Tomado de Meinshausen, (2007), página 378 [04].

REGULARIZACIÓN EN SISTEMAS ESPECÍFICOS

Hay situaciones en las que los sistemas que pretendemos modelar presentan propiedades o estructuras específicas que son conocidas previamente. Al tratarse de sistemas de alta dimensión y dispersos en sus variables la obtención de un modelo adecuado y correcto debe tener en consideración dichas propiedades o estructura.

Por consiguiente, la incorporación a las técnicas de regularización de los condicionantes estructurales del sistema constituye el desarrollo de métodos específicos y adaptados.

Cuando las variables están naturalmente agrupadas y en consecuencia su consideración o inclusión en el modelo deberá estar condicionada por la incorporación previa del grupo al que pertenecen, o cuando la presencia de una variable no es exclusiva de un único grupo sino que puede encontrarse solapada en distintos grupos o cuando hemos de considerar restricciones o penalizaciones adicionales en función del lugar que ocupa una determinada variable o su posición relativa respecto de otras (distancia), son todas ellas situaciones en las que la información de las propiedades o estructura del sistema debe incorporase y tenerse en cuenta en el método de regularización que aplicamos.

Se describen algunos métodos específicos basados en LASSO, como son LASSO-Grupal o LASSO-fusionado que responden a este requerimiento de adaptación al sistema objeto de análisis. Veamos algunos ejemplos

ESTIMADOR LASSO-GRUPAL (*GROUP* LASSO)

Hay situaciones en que las variables están naturalmente agrupadas, un ejemplo destacado es cuando tenemos factores cualitativos entre nuestros predictores. Normalmente codificamos sus niveles usando un conjunto de variables ficticias o contrastes. Una típica situación es el caso de variables categóricas donde cada nivel se codifica como un conjunto de variables ficticias indicadoras.

En la elaboración del modelo se considera que es adecuada la presencia del grupo natural de variables o en caso contrario que no esté presente en el modelo y es deseable que todos los coeficientes dentro de un grupo sean distintos de cero (o cero) simultáneamente, es decir, nos gustaría incluir o excluir este grupo de variables conjuntamente.

Para seleccionar de forma simultánea un grupo de variables surge el LASSO-Grupal propuesto por Bakin (1999) [01] y Lin y Zhang (2006) [01], aunque ampliado y generalizado por Yuan y Lin (2006) [01].

Consideramos un modelo que involucra J grupos de variables, para cada grupo $j = 1,2,\cdots,J$ el vector $\boldsymbol{Z}_j \in \mathbb{R}^{p_j}$ representa las variables presentes en ese grupo j. El objetivo será obtener el modelo basado en el conjunto de grupos $(\boldsymbol{Z}_1, \boldsymbol{Z}_2, \cdots, \boldsymbol{Z}_J)$. Considerando J grupos conocidos y no solapados.

En una función de regresión obtendríamos $\boldsymbol{Y} = \sum_{j=1}^{J} \boldsymbol{Z}_j^T \boldsymbol{\theta}_j$, donde $\boldsymbol{\theta}_j \in \mathbb{R}^{p_j}$

Dada una colección de n muestras $\{(y_i, z_{i1}, z_{i2}, \cdots, z_{iJ})\}_{i=1}^{n}$ la solución LASSO-Grupal se define como $\boldsymbol{\beta}^{LASSO_{Grupal}} = \underset{\{\boldsymbol{\beta}_j \in \mathbb{R}^{p_j}\}_{j=1}^{J}}{\operatorname{argmin}} \left\{ \|\boldsymbol{Y} - \boldsymbol{Z}\boldsymbol{\beta}_j\|_2^2 + \lambda \sum_{j=1}^{J} \|\boldsymbol{\beta}_j\|_2 \right\}$

que puede expresarse también en la forma

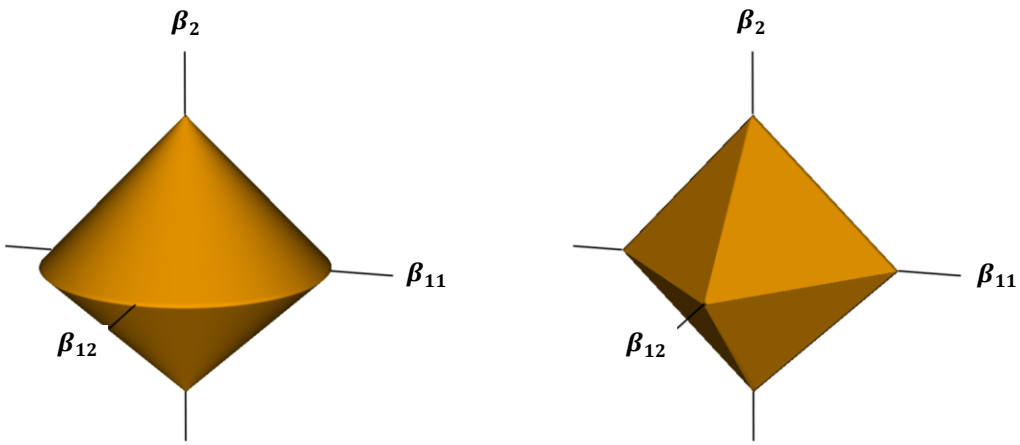
$$\boldsymbol{\beta}^{LASSO_{Grupal}} = \underset{\{\boldsymbol{\beta}_j \in \mathbb{R}^{p_j}\}_{j=1}^{J}}{\operatorname{argmin}} \left\{ \sum_{i=1}^{n} \left(y_i - \sum_{j=1}^{J} z_{ij}^T \boldsymbol{\beta}_j \right)^2 + \lambda \sum_{j=1}^{J} \|\boldsymbol{\beta}_j\|_2 \right\}$$

Como podemos observar en la penalización, se trataría de una aplicación de penalización LASSO con norma L_1 si todos los grupos $\boldsymbol{Z}$ fuesen de un único elemento $p_j = 1, \forall j$ puesto que si $\beta_j \in \mathbb{R} \rightarrow \left\|\boldsymbol{\beta}_j\right\|_2 = \left|\boldsymbol{\beta}_j\right|$ y por otro lado, si consideramos un único grupo $J = 1$ nos encontramos en una situación de penalización *Ridge.*

Por lo tanto, las variables pertenecientes a un grupo tienen una penalización *Ridge* en tanto que las variables individuales tiene el efecto de una penalización LASSO. Es decir, se trata de una combinación de penalización LASSO y *Ridge*, pero a diferencia de *Elastic Net* no afecta por igual a todas las variables, sino las que están agrupadas respecto de las no agrupadas tendrían un comportamiento LASSO, pero la agrupadas tendrán un comportamiento *Ridge* entre ellas.

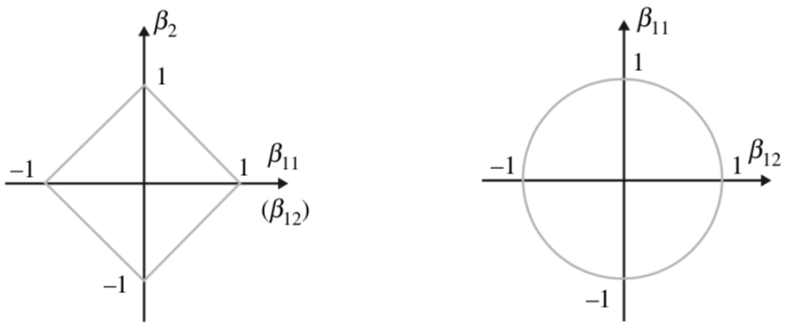
Dependiendo de $\lambda \geq 0$ o el vector $\widehat{\boldsymbol{\beta}}_j$ será cero para todos sus elementos o serán distintos de cero, consiguiendo el propósito de incorporación o no al modelo del grupo de variables en su conjunto.

El efecto combinado de la penalización LASSO-Grupal podemos observarlo en la Gráfica 16, que necesariamente se presenta en 3 dimensiones para plantear un grupo de variables, con coeficientes agrupados $\boldsymbol{\beta}_1 = (\beta_{11}, \beta_{12})$ y $\boldsymbol{\beta}_2 = (\beta_2)$



Gráfica 16. Superficie de penalización LASSO-Grupal (izqu) y LASSO (drch).
Tomado de Hastie, et al. (2015), página 59 [05].

Considerando la intersección sobre el plano de los ejes en la Gráfica 17 se observa para las variables agrupadas la clásica región de penalización *Ridge* en tanto que para β_2 sobre las otras variables β_{11} y β_{12} la región de penalización responde a LASSO.



Gráfica 17. Intersecciones en el plano de los ejes de la región de penalización sobre variables en LASSO-Grupal.
Tomado de Yuan, M., y Lin, Y. (2006), página 52 [02].

Una condición suficiente y necesaria para que LASSO-Grupal tenga solución para el vector de coeficientes $\boldsymbol{\beta} = (\beta_1, \beta_2, \cdots, \beta_J)$ es

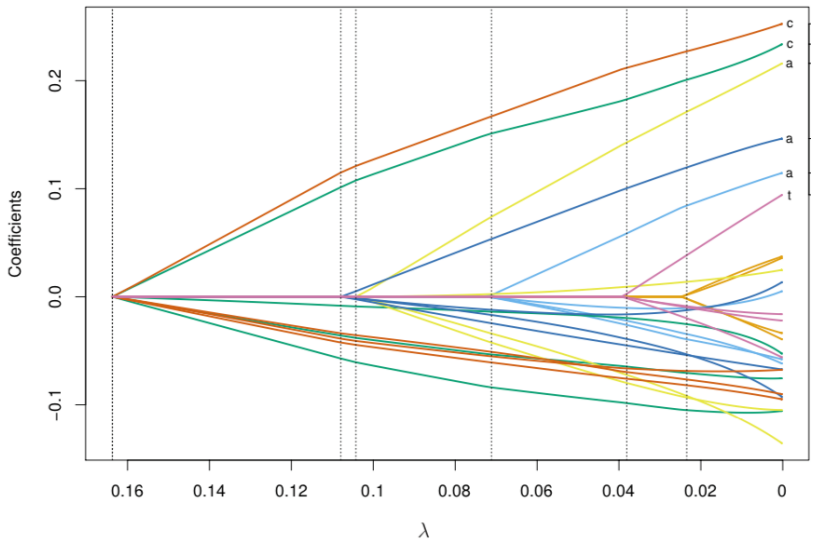
$$\begin{cases} 2\boldsymbol{Z}_j^T(\boldsymbol{Y} - \boldsymbol{Z\beta}) = \dfrac{\lambda\widehat{\boldsymbol{\beta}}_j}{\|\widehat{\boldsymbol{\beta}}_j\|_2} & si \quad \widehat{\boldsymbol{\beta}}_j \neq 0 \\ \left\|-2\boldsymbol{Z}_j^T(\boldsymbol{Y} - \boldsymbol{Z\beta})\right\| \leq \lambda & si \quad \widehat{\boldsymbol{\beta}}_j = 0 \end{cases}$$

Cuyas soluciones son

$$\beta_j = \left(1 - \frac{\lambda}{\|S_j\|}\right)_+ S_j$$

donde $S_j = \boldsymbol{Z}_j^T(\boldsymbol{Y} - \boldsymbol{Z\beta}_{-j})$ con $\beta_{-j} = (\beta_1^T, \beta_2^T, \cdots, \beta_{j-1}^T, 0^T, \beta_{j+1}^T, \cdots, \beta_J^T)$

Veamos su aplicación en el ejemplo de la Gráfica 18 donde se analizan los datos de ADN humano en los que cada observación consta de siete secuencias de ADN (intrones y exones) y cada una de ellas con valores de bases nitrogenadas $\{A, G, C, T\}$. La aplicación de LASSO-Grupal muestra como cada secuencia (grupo de cuatro bases nitrogenadas) es incluida o no al variar la penalización λ. Cada línea vertical indica cuando entra una secuencia completa, es decir, $p_j = 4,\ j = 1,2,\cdots,7$. Como hemos comentado las trayectorias de las bases nitrogenadas pertenecientes a una secuencia responden a una penalización *Ridge*, en tanto que, si consideramos las siete secuencias (grupos) son seleccionados (incluidos en el sistema) de acuerdo con las trayectorias de una penalización LASSO mediante Umbral Suave.



Gráfica 18. Trayectoria de coeficientes por aplicación de LASSO-Grupal a una muestra de datos de ADN humano donde los coeficientes se agrupan en conjuntos de cuatro bases nitrogenadas $\{\boldsymbol{A}, \boldsymbol{G}, \boldsymbol{C}, \boldsymbol{T}\}$ para cada una de las siete secuencias consideradas.
Tomado de Hastie et al. (2015), página 61 [06].

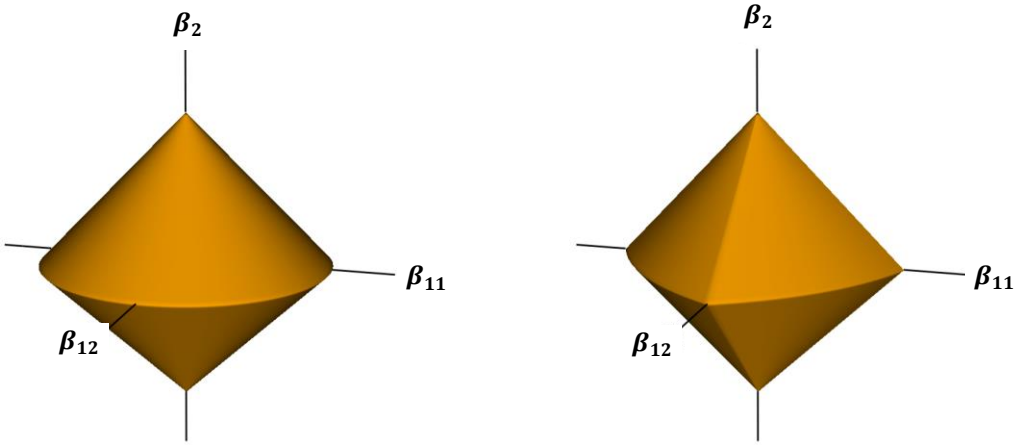
ESTIMADOR LASSO-GRUPAL-DISPERSO (*SPARSE GROUP* LASSO)

Hemos visto como la inclusión de un grupo en la selección de variables mediante el método LASSO-Grupal implica la inclusión de todas las variables pertenecientes al grupo incluido en cuestión. Sin embargo, puede considerarse la situación en la que no todas las variables pertenecientes a un grupo necesariamente estén activas de forma conjunta, por ejemplo, en un tipo de enfermedad pueden estar implicado un determinado cromosoma, pero no todos los genes de ese cromosoma necesitan estar activos.

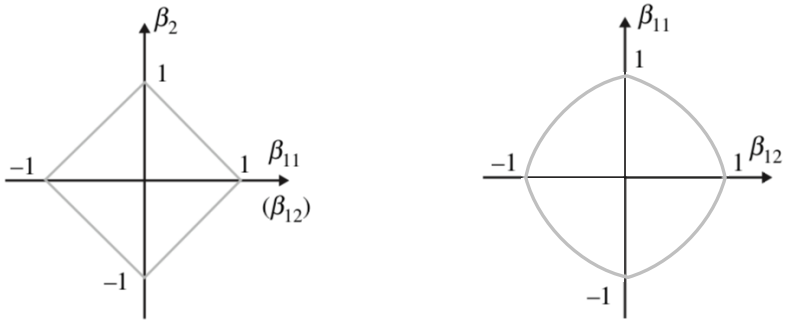
Cuando se incluye un grupo en LASSO-Grupal, la penalización que "fuerza" la inclusión conjunta de todas las variables pertenecientes a ese grupo es *Ridge*, si sustituimos la penalización *Ridge* por una penalización *Elastic Net* conseguiremos la capacidad de selección de variables dentro del grupo, esta estrategia de penalización es denominada LASSO-Grupal-Disperso (*Sparse Group* LASSO).

$$\boldsymbol{\beta}^{\substack{LASSO\ Grupal \\ Disperso}} = \underset{\{\boldsymbol{\beta}_j \in \mathbb{R}^{p_j}\}_{j=1}^{J}}{\operatorname{argmin}} \left\{ \sum_{i=1}^{n} \left(y_i - \sum_{j=1}^{J} z_{ij}^{T} \boldsymbol{\beta}_j \right)^2 + \lambda \sum_{j=1}^{J} \left[(1-\alpha) \left\| \boldsymbol{\beta}_j \right\|_2 + \alpha \left\| \boldsymbol{\beta}_j \right\|_1 \right] \right\}$$

En la Gráfica 19 podemos ver las regiones de penalización donde la aplicación de la penalización *Ridge* en el grupo β_1 ha sido sustituida por la penalización *Elastic Net*, más claramente podemos observarlo en la Gráfica 20 con las intersecciones con los planos de los ejes.



Gráfica 19. Superficie de penalización LASSO-Grupal (izqu) y LASSO-Grupal-Disperso (drch). Tomado de Hastie et al. (2015), página 64 [07].



Gráfica 20. Intersecciones en el plano de los ejes de la región de penalización sobre variables en LASSO-Grupal-Disperso.

El problema de optimización LASSO-Grupal-Disperso, al igual que LASSO-Grupal es convexo y su solución óptima está especificada por las ecuaciones de subgradiente cero.

ESTIMADOR LASSO-GRUPAL-SOLAPADO (*OVERLAP GROUP* LASSO)

Hay situaciones en las que las variables puede pertenecer a más de un grupo, por ejemplo $Z_1 = \{X_1, X_2, X_3\}$ y $Z_2 = \{X_3, X_4, X_5\}$, en caso de replicar la variable solapada supondría incrementar la posibilidad de que dicha variable sea incluida en el modelo, por otro lado replicar el coeficiente asignado a dicha variable supondría que en el caso de que el coeficiente de uno de los grupos que contiene a la variable fuese cero forzosamente el otro grupo o grupos tendrían un coeficiente de grupo igual a cero, es decir, los grupos con solape de variable entrarían o saldrían del modelo de forma conjunta.

Jacob et al. (2009) [01] propusieron un enfoque variable replicado o LASSO-Grupal-Solapado. En el ejemplo anterior los posibles conjuntos de coeficientes distintos de cero para el LASSO-Grupal-Solapado son $\{1,2,3\}$, $\{3,4,5\}$ y $\{1,2,3,4,5\}$. En general, los conjuntos de posibles coeficientes distintos de cero siempre corresponden a grupos, o las uniones de grupos. También definieron una penalización implícita en las variables originales que produce el enfoque de la variable replicada como su solución.

Definiendo $v_j \in \mathbb{R}^p$ como un vector que es cero en todas las posiciones excepto en las posiciones correspondiente a miembros del grupo j, siendo $\mathbf{v}_j \subseteq \mathbb{R}^p$ el subespacio de todos los posibles vectores. En términos de las variables originales $X = \left(X_1, X_2, \cdots X_p\right)$, el vector de coeficientes vendrá dado por la suma $\boldsymbol{\beta} = \sum_{j=1}^{J} v_j$.

Considerando el vector de coeficientes $\boldsymbol{\beta}$ los grupos definidos, así como las uniones de grupos para aplicar la penalización LASSO-Grupal, tenemos que el LASSO-Grupal-Solapado respondería al problema de optimización

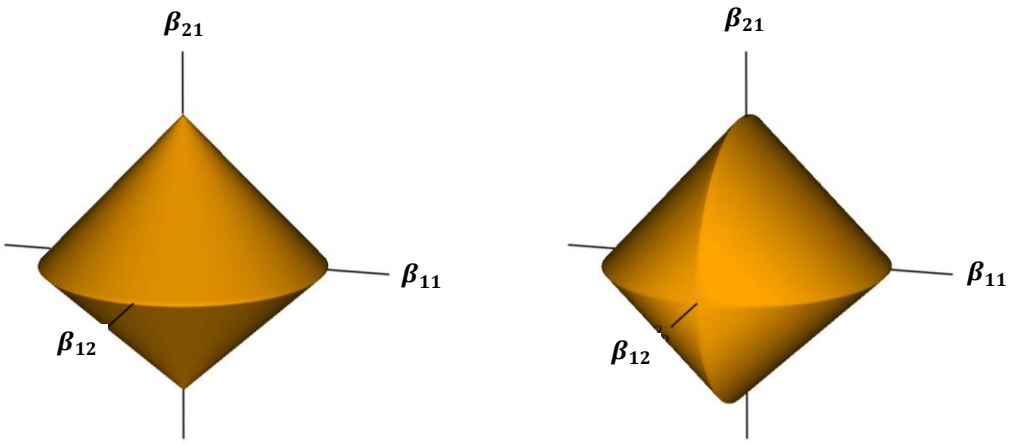
$$\boldsymbol{\beta}^{LASSO\,Grupal\;Solapado} = \underset{\{\boldsymbol{v}_j \in \mathbf{v}_j\}_{j=1}^{J}}{\operatorname{argmin}} \left\{ \left\| \boldsymbol{Y} - \boldsymbol{X} \left(\sum_{j=1}^{J} \boldsymbol{v}_j \right) \right\|_2^2 + \lambda \sum_{j=1}^{J} \left\| \boldsymbol{v}_j \right\|_2 \right\}$$

Expresado en términos del vector de coeficientes β tenemos

$$\Omega_v(\beta) = \inf_{\substack{\{\boldsymbol{v}_j \in \boldsymbol{v}_j\}_{j=1}^{J} \\ \boldsymbol{\beta}=\sum_{j=1}^{J} \boldsymbol{v}_j}} \left\{ \sum_{j=1}^{J} \|\boldsymbol{v}_j\|_2 \right\}$$

$$\boldsymbol{\beta}^{\substack{LASSO_{Grupal} \\ Solapado}} = \underset{\boldsymbol{\beta}\in\mathbb{R}^{\boldsymbol{p}}}{\operatorname{argmin}} \{\|\boldsymbol{Y} - \boldsymbol{X\beta}\|_2^2 + \lambda\Omega_v(\beta)\}$$

Veamos la región de penalización compuesta por las distintas penalizaciones en base a las normas aplicadas a cada coeficiente. Supongamos tres variables $\{X_1, X_2, X_3\}$ que forman parte de dos grupos con una de las variables compartida en ambos grupos $\{X_1, X_2\}$ y $\{X_2, X_3\}$. En la Gráfica 21 observamos la diferencia de contorno en la región de penalización respecto de un LASSO-Grupal sin solapamiento $\{X_1, X_2\}$ y $\{X_3\}$

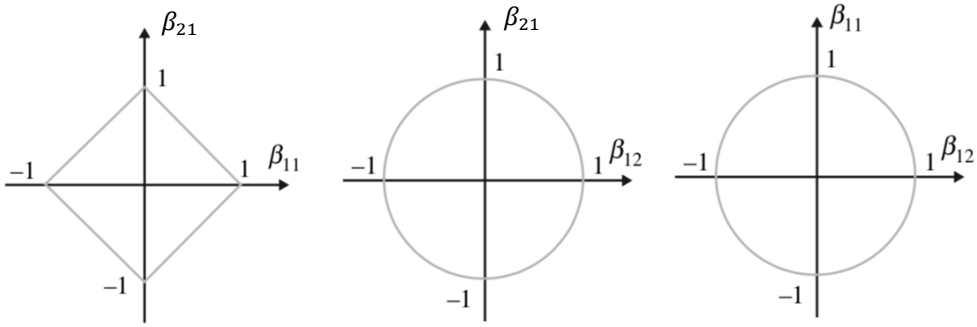


Gráfica 21. Superficie de penalización LASSO-Grupal (izqu) y LASSO-Grupal-Solapado (drch). Tomado de Hastie et al. (2015), página 67 [08].

Los grupos y coeficientes considerados en el ejemplo de la Gráfica 21 serían para LASSO-Grupal $Z_1 = \{X_1, X_2\} \rightarrow \beta_1 = \beta_{11} + \beta_{12}$ y el segundo grupo $Z_2 = \{X_3\} \rightarrow \beta_2 = \beta_{21}$, así para LASSO-Grupal-Solapado tenemos $Z_1 = \{X_1, X_2\} \rightarrow \beta_1 = \beta_{11} + \beta_{12}$ y el segundo grupo $Z_2 = \{X_2, X_3\} \rightarrow \beta_2 = \beta_{21} + \beta_{12}$

En la variable solapada β_{12} un valor alejado de cero implicaría en las otras variables que comparten grupo con ésta β_{21} y β_{11} una menor penalización (más similar a *Ridge*, L_2).

En las intersecciones con los planos de los ejes mostradas en la Gráfica 22 se observa más claramente la penalización correspondiente *Ridge* L_2 para $\{\beta_{21}, \beta_{12}\}$ y para $\{\beta_{11}, \beta_{12}\}$ y penalización LASSO L_1 para $\{\beta_{21}, \beta_{11}\}$ ya que las variables X_1 y X_3 no comparten agrupamiento entre ellas.



Gráfica 22. Intersecciones en el plano de los ejes de la región de penalización sobre variables en LASSO-Grupal-Solapado.

ESTIMADOR LASSO-FUSIONADO (*FUSED* LASSO)

Un inconveniente de LASSO es el hecho de que ignora la ordenación de las variables, para situaciones en las que el orden de las variables o su distancia relativa son relevantes en el modelo porque por ejemplo variables cercanas deben tener coeficientes similares. El método LASSO-Fusionado (Tibshirani et al., 2005) [01] contempla este requerimiento ya que penaliza tanto los coeficientes como la diferencia entre los coeficientes adyacentes.

El LASSO-Fusionado, que es especialmente útil en sistema dispersos $p \gg n$, fue motivado originalmente por el problema de analizar datos de espectroscopía de masas de proteínas y datos de expresión génica, donde se sabe que las variables espacialmente más próximas son conjuntamente relevantes o irrelevantes.

Tibshirani et al., (2005) [02] tomaron prestado el término fusión de Land y Friedman (1996) [01] que propusieron una penalización del tipo $\sum_{j=1}^{n}\left|\beta_j - \beta_{j-1}\right|^{\alpha} \leq s_2$ para varios valores de α denominada fusión variable (*variable fusion*), especialmente $\alpha = 0, 1, 2$. Sin embargo, estos no consideraron la aplicación conjunta de las dos penalizaciones (LASSO $\sum_{j=1}^{n}\left|\beta_j\right|$ y fusión variable $\sum_{j=1}^{n}\left|\beta_j - \beta_{j-1}\right|$).

El LASSO-Fusionado se expresa mediante la doble penalización como

$$\boldsymbol{\beta}^{LASSO_{Fusionado}} = \underset{\boldsymbol{\beta}\in\mathbb{R}^p}{\operatorname{argmin}} \left\{ \sum_{i=1}^{n}\left(y_i - \sum_{j=1}^{p} x_{ij}\boldsymbol{\beta}_j \right)^2 + \lambda_1 \sum_{j=1}^{p}\left|\beta_j\right| + \lambda_2 \sum_{j=2}^{p}\left|\beta_j - \beta_{j-1}\right| \right\}$$

$$\boldsymbol{\beta}^{LASSO_{Fusionado}} = \underset{\boldsymbol{\beta}\in\mathbb{R}^p}{\operatorname{argmin}} \left\{ \|\boldsymbol{Y} - \boldsymbol{X}\boldsymbol{\beta}\|_2^2 + \lambda_1 \sum_{j=1}^{p}\left|\beta_j\right| + \lambda_2 \sum_{j=2}^{p}\left|\beta_j - \beta_{j-1}\right| \right\}$$

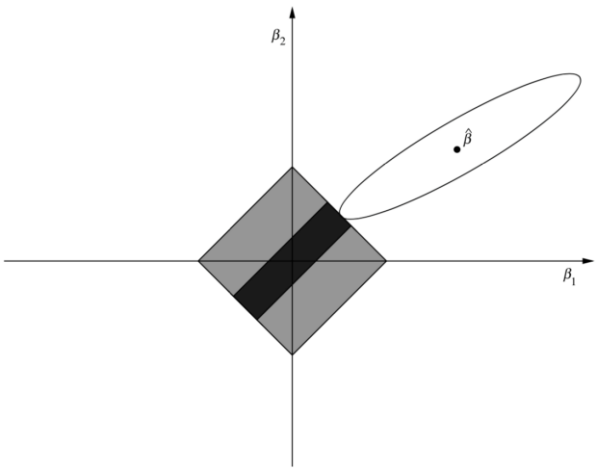
con $\lambda_1;\ \lambda_2 \geq 0$.

La primera penalización es el LASSO que favorece la dispersión (*sparsity*) en los coeficientes y la segunda penalización potencia que los coeficientes próximos (vecinos) sean idénticos.

Esta penalización de la vecindad puede ser generalizada para distintos conceptos, por ejemplo, pixeles adyacentes en una imagen, de esta forma podemos plantear una generalización de la segunda penalización de la forma

$$\lambda_2 \sum_{j\sim j'}^{p} |\beta_j - \beta_{j'}|$$

En cuanto a la descripción de la región de penalización podemos ver en la Gráfica 23 como esta región, al estar condicionada por dos penalizaciones, implica que el cumplimiento de ambas será más restrictivo y por consiguiente formado por la intersección entre ellas. Si la restricción LASSO determina el rombo $\sum_{j=1}^{p}|\beta_j| \leq s_1$, y la restricción fusión determina la banda lateral que cumple $\sum_{j=1}^{n}|\beta_j - \beta_{j-1}| \leq s_2$, la restricción LASSO-Fusionado es la región indicada en negro, lo que significa que se "sacrifica" la prioridad de anulación de coeficientes a favor la igualación de coeficientes próximos entre ellos.



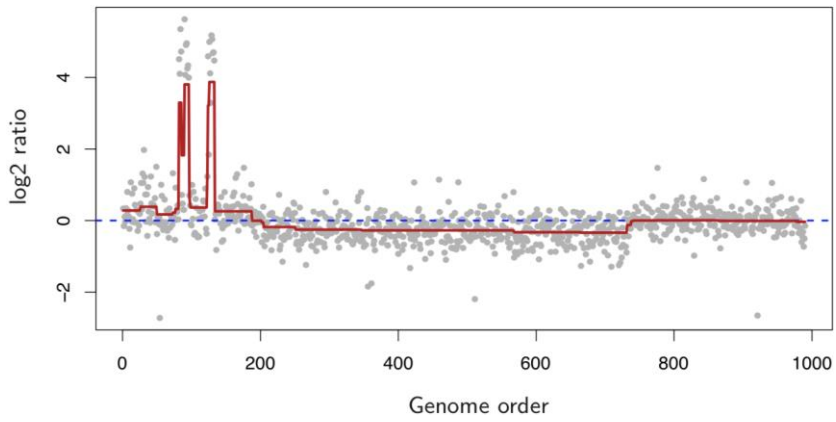
Gráfica 23. Región de penalización LASSO-Fusionado.
Tomado de Tibshirani et al. (2005), página 93 [03].

Veamos su aplicación en un caso especial particularmente útil que surge cuando consideramos la matriz de predicción $\boldsymbol{X} = \boldsymbol{I}_n$ (la matriz de identidad $N \times N$). Este es un caso especial del LASSO-Fusionado, usado para aproximar una secuencia $\{y_i\}_{i=1}^{n}$.

$$\boldsymbol{\beta}^{LASSO_{Fusionado}} = \underset{\boldsymbol{\beta}\in\mathbb{R}^n}{\operatorname{argmin}} \left\{ \sum_{j=1}^{n} (y_i - \beta_0 - \beta_j)^2 + \lambda_1 \sum_{j=1}^{n} |\beta_j| + \lambda_2 \sum_{j=1}^{n-1} |\beta_{j+1} - \beta_j| \right\}$$

Un ejemplo de esta aplicación podemos observarlo en Friedman et al. (2001) [06] en la Gráfica 24 que muestra los resultados de un experimento de hibridación genómica comparativa (CGH en sus siglas en inglés) donde se han tomado el logaritmo base 2 del número de copias relativas de cada gen[15] en una muestra de cáncer respecto una muestra de control, como observamos los datos presentan bastante ruido. El eje horizontal representa la localización cromosómica de cada gen.

La idea subyacente es que, en las células cancerosas, los genes a menudo se amplifican (duplican) o se eliminan, y es interesante detectar estos eventos. Además, estos eventos tienden a ocurrir en regiones contiguas. La estimación de la señal suavizada del aproximador de señal de LASSO-Fusionado se muestra en rojo oscuro (con valores elegidos adecuadamente para λ_1 y λ_2). Las regiones significativamente distintas de cero pueden usarse para detectar ubicaciones de ganancias y pérdidas de genes en el tumor.



Gráfica 24. LASSO-Fusionado aplicado a datos de CGH, cada punto representa el número relativo de copias de gen en una muestra de tumorosa respecto de una sana (en una escala $\boldsymbol{log_2}$).
Tomado de Friedman et al. (2001), página 667 [07].

[15] Realmente desde un punto de vista biológico se trata de segmentos de un cromosoma más que de genes individuales.

CAPÍTULO 2

AGRUPAMIENTO DISPERSO

Cuando tenemos una muestra de alta dimensión $p \gg n$ se podría esperar que los verdaderos *clusters* subyacentes presentes en los datos difieran solo con respecto a una pequeña fracción de las características. El agrupamiento disperso agrupa las observaciones utilizando un subconjunto reducido de características o variables, como hemos comentado. Por consiguiente, la identificación y asignación de *clusters* se realizará de acuerdo con un limitado conjunto de variables, siendo la mayoría, como es propio de los sistemas dispersos igual a cero.

En el presente capítulo vamos a describir los métodos de regularización que permiten identificar y seleccionar a aquellas características que son las relevantes en la identificación y asignación del *cluster* al que pertenece cada elemento de la muestra.

Al igual que en el capítulo anterior describiremos un recorrido por los métodos que se han utilizado en este campo específico del análisis multivariante (el agrupamiento), comenzando por métodos aplicables a sistemas perfectamente definidos o sistemas compatibles determinados $n > p$, que no son de alta dimensión.

La irrupción de la potencia de tratamiento de datos a nivel tecnológico también llegó al campo del agrupamiento incorporando volúmenes de datos de alta dimensión que para su análisis requerían de la asunción de sistemas dispersos, y en consecuencia la aplicación de aquellas técnicas de regularización que permiten bajo determinados criterios identificar el pequeño conjunto de características que define y asignar correctamente el *cluster* al que pertenece cada elemento muestral.

Comenzamos esbozando el planteamiento de la aplicación de la regularización en cuanto a selección de variables.

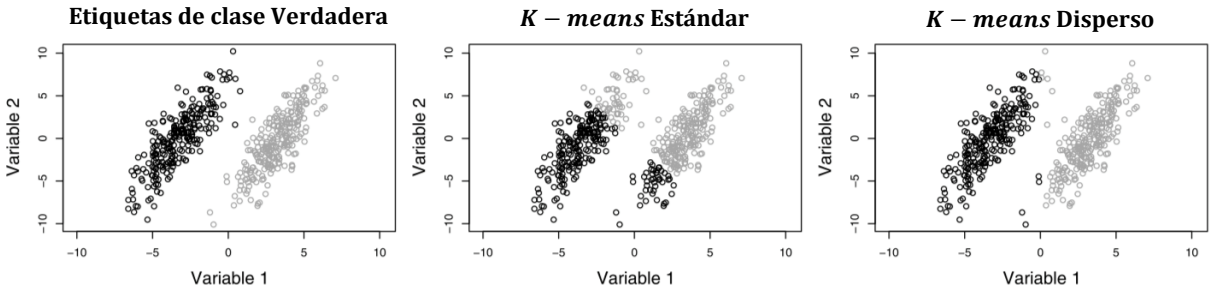
Sea $\boldsymbol{X}$ una matriz de datos $n \times p$, con n observaciones y p características. Supongamos que deseamos agrupar las observaciones y sospechamos que los verdaderos grupos subyacentes difieren solo con respecto a algunas de las características. Este método es más útil para la configuración de alta dimensión donde $p \gg n$, pero también se puede usar cuando $p < n$.

El agrupamiento disperso tiene una serie de ventajas. Si los *clusters* subyacentes difieren solo en términos de algunas de las características, entonces podría resultar en una identificación más precisa de estos *clusters* que el agrupamiento en *clusters* estándar. También produce resultados interpretables, ya que se puede determinar con precisión qué características son responsables de las diferencias observadas entre los grupos o *clusters*. Además, se requieren menos funciones para asignar una nueva observación a un *cluster* preexistente.

Por consiguiente nos encontramos igualmente, como en el caso de la regresión, con una consecuencia de los sistemas dispersos que es las obtención de modelos de menor complejidad, y por lo tanto más simples, al permitirnos "sacar" del modelo a variables que estarían presentes en modelos de agrupamiento estándar por acción de factores externos, como puede ser la presencia de ruido, lo que llevaría a un sobreajuste que además nos aleja del modelo correcto de agrupamiento que pretendemos describir.

Es decir, no únicamente en sistemas de alta dimensión $(p \gg n)$ la consideración de dispersión (conjunto de variables $(k < p)$ con valor cero) puede permitirnos obtener unos modelos de agrupamiento más correctos y mejor definidos, como se puede observar en el siguiente ejemplo.

Generamos 500 observaciones independientes a partir de una distribución normal bivariada $n = 500 > p = 2$. Un cambio de media en la primera característica define las dos clases. Los datos resultantes, así como los *clusters* obtenidos mediante el agrupamiento estándar de *2-means* y como alternativa un agrupamiento disperso de *2-means*, se pueden ver en la Gráfica 25. A diferencia del agrupamiento estándar de *2-means*, el agrupamiento disperso de *2-means* significa que automáticamente identifica un subconjunto de las características para agrupar las observaciones. Aquí utiliza solo la primera característica y concuerda bastante bien con las verdaderas etiquetas de clase.



Gráfica 25. Ejemplo bidimensional, dos clases difieren solo con respecto a la primera característica. *Sparse 2-means clustering* selecciona solo la primera característica y, por lo tanto, produce un resultado superior.
Tomado de Witten y Tibshirani (2010), página 714 [01].

Un elemento clave que ha de ser considerado como punto de partida en análisis de agrupamiento es la referencia por la que vamos a diferenciar los datos de la muestra, y van a ser distinguidos unos de otros, la disimilitud.

Los métodos de agrupamiento requieren algún concepto de la disimilitud entre pares de observaciones. Supongamos que $d(\boldsymbol{x}_i, \boldsymbol{x}_{i'})$ denota cierta medida de disimilitud entre las observaciones $\boldsymbol{x}_i$ y $\boldsymbol{x}_{i'}$, que son las filas i e i' de la matriz de datos $\boldsymbol{X}$, consideramos que $d(\boldsymbol{x}_i, \boldsymbol{x}_{i'}) = \sum_{j=1}^{p} d_{i,i',j}$, donde $d_{i,i',j}$ indica la disimilitud entre las observaciones i e i' a lo largo de la característica j.

Podemos tomar d como la distancia euclídea, $d_{i,i',j} = \left(X_{ij} - X_{i'j}\right)^2$, sin embargo, son posibles otras medidas de disimilitud, como la diferencia absoluta $d_{i,i',j} = \left|X_{ij} - X_{i'j}\right|$, distancia Mahalanobis o cualquiera que consideremos en función de las características del análisis, distancia Minkowski, etc.

Hagamos en primer lugar un recorrido por las estrategias que varios autores han contemplado respecto la necesidad de técnicas de agrupamiento específicas buscando la selección de variables definitorias del modelo en un número lo más reducido posible, es decir, modelos dispersos.

REDUCCIÓN DE LA DIMENSIONALIDAD

Los primeros métodos se orientaban a la reducción de la dimensionalidad lo que se traduce en una disminución de la complejidad del modelo, permitiendo describir a éste con un número reducido (menor) de variables.

Componentes principales. PCA

Una forma de reducir la dimensionalidad de los datos antes del agrupamiento es mediante una descomposición de matriz. Se puede aproximar la matriz $n \times p$ de datos $\boldsymbol{X}$ como $\boldsymbol{X} \approx \boldsymbol{AB}$ donde $\boldsymbol{A}$ es una matriz $n \times q$ y $\boldsymbol{B}$ es una matriz $q \times p$, $q \ll p$. Posteriormente, se pueden agrupar las observaciones utilizando $\boldsymbol{A}$ como matriz de datos, en lugar de $\boldsymbol{X}$.

Por ejemplo, Ghosh y Chinnaiyan (2002) [01] y Liu et al. (2003) [01] proponen realizar análisis de componentes principales (PCA) para obtener una matriz $\boldsymbol{A}$ de dimensionalidad reducida; seguidamente, las n filas de $\boldsymbol{A}$ se pueden agrupar habiendo reducido de esta forma la dimensionalidad.

Factorización matricial no negativa

Tamayo et al. (2007) [01] sugieren descomponer $\boldsymbol{X}$ utilizando la factorización matricial no negativa (Lee y Seung, 1999) [01], básicamente es el mismo procedimiento que la factorización por componentes principales, pero con el condicionante o restricción que la descomposición factorial no permite entradas negativas y en consecuencia todos los elementos de $\boldsymbol{A}$ y $\boldsymbol{B}$ son positivos, lo cual facilita una descomposición con dispersión de factores, aunque presenta la desventaja de requerir parámetros de ajuste en procesos de ensayo y error que implican altos consumos de tiempo de computación.

Una vez obtenida la descomposición asimismo se procede posteriormente a aplicar algún método de agrupamiento para las filas de $\boldsymbol{A}$.

Sin embargo, estos enfoques tienen varios inconvenientes, en primer lugar, el agrupamiento resultante no es disperso en las características, ya que cada una de las columnas de $\boldsymbol{A}$ es una función del conjunto completo de características p. Además, no hay garantía de que $\boldsymbol{A}$ contenga la o las características que estamos interesados en detectar a través del agrupamiento.

Combinación de modelos de distribución

Podemos encontrar una descripción general del agrupamiento basado en combinación de modelos en McLachlan, Bean y Peel (2002) [01] y Fraley y Raftery (2002) [01].

La idea básica consiste en modelar las filas de $\boldsymbol{X}$ como observaciones multivariadas independientes extraídas de un modelo combinado de K componentes.

Tomemos $\boldsymbol{x}_1, \boldsymbol{x}_2, \cdots, \boldsymbol{x}_n$ para denotar n observaciones p-dimensionales. Con un enfoque basado generalmente en un modelo combinado de distribuciones normales para el agrupamiento de estos datos, suponemos que cada observación $\boldsymbol{x}_i$ proviene de una combinación de un número inicialmente especificado K de densidades normales multivariables en proporciones desconocidas $\pi_1, \pi_2, \cdots, \pi_k$. Es decir, $\boldsymbol{x}_i$ se toma como una realización de un vector aleatorio $\boldsymbol{X}$ que tiene la función de densidad de probabilidad combinada $f(\boldsymbol{x}; \boldsymbol{\Psi})$ definida por

$$f(\boldsymbol{x}; \boldsymbol{\Psi}) = \sum_{k=1}^{K} \pi_k \phi_k(\boldsymbol{x}; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$$

donde $\phi_k(\boldsymbol{x}; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$ denota la función de probabilidad, función de densidad normal p-variante con media $\boldsymbol{\mu}_k$ y matriz de covarianza $\boldsymbol{\Sigma}_k$ con $k = (1,2, \cdots, K)$.

Aquí el vector $\boldsymbol{\Psi}$ de parámetros desconocidos consta de la mezcla de proporciones π_k, de los elementos del componente medias $\boldsymbol{\mu}_k$, y los distintos elementos del componente matrices de covarianza $\boldsymbol{\Sigma}_k$ con $k = (1,2, \cdots, K)$.

Bajo el supuesto de que $\boldsymbol{x}_1, \boldsymbol{x}_2, \cdots, \boldsymbol{x}_n$ son observaciones independientes, tomando la función de verosimilitud

$$L(\boldsymbol{\Psi}) = \prod_{i=1}^{n} f_i(\boldsymbol{x}; \boldsymbol{\Psi}) = \prod_{i=1}^{n} \sum_{k=1}^{K} \pi_k \phi_k(\boldsymbol{x}_i; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$$

Obtenemos que

$$logL(\boldsymbol{\Psi}) = \sum_{i=1}^{n} log \left[\sum_{k=1}^{K} \pi_k \phi_k(\boldsymbol{x}_i; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) \right] = \sum_{i=1}^{n} log\, f(\boldsymbol{x}_i; \boldsymbol{\Psi}) \tag{1}$$

y la estimación de máxima verosimilitud del vector de parámetros $\boldsymbol{\Psi}$ la obtenemos

$$\partial logL(\boldsymbol{\Psi}) / \partial \boldsymbol{\Psi} = 0 \tag{2}$$

Las soluciones de (2) correspondientes a los máximos locales se pueden encontrar de forma iterativa mediante la aplicación del algoritmo de Maximización de Esperanza (EM) de Dempster, Laird, y Rubin (1977) [01]; véase también McLachlan y Krishnan (2007) [01].

Sin embargo, cuando consideramos sistemas de alta dimensión $p \approx n$ o $p \gg n$ surge un problema porque la matriz $p \times p$ de covarianza $\boldsymbol{\Sigma}_i$ no se puede estimar a partir de solo n observaciones. Las propuestas para superar este problema incluyen el enfoque del análisis factorial de McLachlan, Bean y Peel (2002) [02] y McLachlan, Peel y Bean (2003) [01], que asume que las observaciones se encuentran en un espacio de factores latentes de baja dimensión. Esto conduce a la reducción de la dimensionalidad, pero no a la dispersión.

El modelo base de agrupamiento se presta fácilmente a la selección de características. En lugar de buscar los parámetros de $\boldsymbol{\Psi}$, $\boldsymbol{\mu}_k$ y $\boldsymbol{\Sigma}_k$ que maximizan la probabilidad del logaritmo (1), se puede maximizar la probabilidad del logaritmo sujeto a una penalización que se elige para obtener la dispersión en las características. Este enfoque se toma en varios artículos, incluidos Pan y Shen (2007) [01], Wang y Zhu (2008) [01], y Xie, Pan y Shen (2008) [01]. Por ejemplo, si asumimos que las características de $\boldsymbol{X}$ están centradas para tener una media de cero, entonces Pan y Shen (2007) [02] proponen maximizar la verosimilitud penalizada

$$\sum_{i=1}^{n} log \left[\sum_{k=1}^{K} \pi_k \phi_k(\boldsymbol{x}_i; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) \right] - \lambda \sum_{k=1}^{K} \sum_{j=1}^{p} |\mu_{ij}| \tag{3}$$

donde $\boldsymbol{\Sigma}_1 = \boldsymbol{\Sigma}_2 = \cdots = \boldsymbol{\Sigma}_K$ se toma como una matriz diagonal. Es decir, se aplica una penalización L_1 (o LASSO) a los elementos de $\boldsymbol{\mu}_k$. Cuando el parámetro de ajuste no negativo λ es grande, algunos de los elementos de μ_k serán exactamente iguales a cero. Si, para algunas características $j, \mu_{kj} = 0$ para todos $k = 1,2,\cdots,K$, entonces el agrupamiento resultante no involucrará la característica j. Por lo tanto, esto produce un agrupamiento que es disperso en las características.

Hemos obtenido un modelo que al penalizar las distribuciones base (sus medias) que componen los elementos muestrales nos puede permitir obtener valores cero en todas las distribuciones base de alguna determinada característica de las p iniciales (no en sistemas de alta dimensión, sino cuando $n > p$), lo que nos lleva a poder "extraerla" del modelo.

Por lo tanto, en este caso sí se está produciendo una auténtica selección de variables (características) mediante la aplicación de una penalización, es decir, estamos realmente aplicando una regularización que nos lleva a obtener un modelo disperso (más disperso en función del número de características $0 \leq j \leq p$ que podamos anular mediante la elección adecuada del parámetro de penalización). Sin embargo, en sistemas de alta dimensión es necesario pasar previamente por la factorización en componentes principales para aplicar el método a estos componentes principales, lo cual, como hemos visto, no genera un sistema disperso ya que los componentes principales son combinaciones lineales de las variables presentes p.

Selección de variables en agrupamiento basado en modelos

Raftery y Dean (2006) [01] también presentan un método para la selección de características en la configuración del agrupamiento basada en modelos, utilizando un enfoque diferente. Refunden el problema de selección de variables como un problema de selección de modelos: se comparan los modelos que contienen subconjuntos de variables anidados. Los modelos anidados son dispersos en las características y, por lo tanto, esto produce un método para el agrupamiento disperso. El modelo Raftery y Dean (2006) [02] resultante selecciona variables (características), la cantidad de *clusters* y el modelo de agrupamiento en *clusters* simultáneamente.

Una propuesta relacionada se hace en Maugis, Celeux y Martin-Magniette (2009) [01].

Raftery y Dean (2006) [03] incluyen el procedimiento de selección de variables como parte del algoritmo de agrupamiento.

El factor Bayesiano para un modelo M_1 contra un modelo competidor M_2 es igual a las probabilidades posteriores para M_1 contra M_2 cuando las probabilidades de ambos en el modelo anterior son iguales. Se calcula como la relación de las probabilidades integradas para los dos modelos.

Para abordar el problema de selección de variables, lo reformulamos como un problema de selección de modelo. Tenemos un conjunto de datos Y, y en cualquier etapa de nuestro algoritmo de selección de variables, se divide en tres conjuntos de variables, $Y^{(1)}$, $Y^{(2)}$ e $Y^{(3)}$, de la siguiente manera:

- $Y^{(1)}$, el conjunto de variables de agrupamiento ya seleccionadas.
- $Y^{(2)}$, la(s) variable(s) que se considera para su inclusión en o exclusión del conjunto de variables de agrupamiento, serían las variables susceptibles de cambio de estatus.
- $Y^{(3)}$, las variables restantes.

El modelo M_1 especifica que dada $Y^{(1)}, Y^{(2)}$ es condicionalmente independiente de las membresías del *cluster* (definidas por las variables no observadas $\boldsymbol{z}$); es decir, $Y^{(2)}$ no proporciona información adicional sobre el agrupamiento. El modelo M_2 implica que $Y^{(2)}$ proporciona información adicional sobre la membresía de agrupamiento en *cluster*, después de que se haya observado $Y^{(1)}$.

$$
\begin{aligned}
M_1: \quad & p(Y|\boldsymbol{z}) \\
& = p\left(Y^{(1)}, Y^{(2)}, Y^{(3)}|\boldsymbol{z}\right) \\
& = p\left(Y^{(3)}\middle|Y^{(2)}, Y^{(1)}\right) p\left(Y^{(2)}\middle|Y^{(1)}\right) p\left(Y^{(1)}|\boldsymbol{z}\right)
\end{aligned}
$$

$$
\begin{aligned}
M_2: \quad & p(Y|\boldsymbol{z}) \\
& = p\left(Y^{(1)}, Y^{(2)}, Y^{(3)}|\boldsymbol{z}\right) \\
& = p\left(Y^{(3)}\middle|Y^{(2)}, Y^{(1)}\right) p\left(Y^{(1)}, Y^{(2)}|\boldsymbol{z}\right)
\end{aligned}
$$

Los modelos M_1 y M_2 se comparan mediante una aproximación al factor Bayesiano que permite que $p\left(Y^{(3)}\middle|Y^{(2)},Y^{(1)}\right)$ de alta dimensión se cancele de la relación. El factor Bayesiano, B_{12}, para M_1 contra M_2 basado en los datos Y está dado por

$$B_{12}=\frac{p(Y|M_1)}{p(Y|M_2)}=\frac{p\left(Y^{(3)}\middle|Y^{(2)},Y^{(1)}\right)p\left(Y^{(2)}\middle|Y^{(1)}\right)p\left(Y^{(1)}|\boldsymbol{z}\right)}{p(Y^{(3)}|Y^{(2)},Y^{(1)})p(Y^{(1)},Y^{(2)}|\boldsymbol{z})}=\frac{p\left(Y^{(2)}\middle|Y^{(1)}\right)p\left(Y^{(1)}|\boldsymbol{z}\right)}{p(Y^{(1)},Y^{(2)}|\boldsymbol{z})}$$

donde $p(Y|M_k)$ es la probabilidad integrada del modelo $M_k,\ k=1{,}2$, a saber

$$p(Y|M_k)=\int p(Y|\boldsymbol{\theta}_k,M_k)p(\boldsymbol{\theta}_k|M_k)\,d\boldsymbol{\theta}_k \tag{4}$$

$\boldsymbol{\theta}_k$ es el parámetro con valores vectoriales del modelo M_k y $p(\boldsymbol{\theta}_k|M_k)$ es su distribución previa (Kass y Raftery 1995) [01].

La relación B_{12} puede ser difícil de calcular, y utilizamos el criterio de información bayesiano (BIC) fácilmente calculable como base para una aproximación. Esto se define por

$$BIC=2\times log(máxima\ verosimilitud)-(nº\ de\ parámetros)\times log(n) \tag{5}$$

Las diferencias de menos de 2 entre los valores de BIC generalmente se consideran apenas dignas de mención, mientras que las diferencias mayores de 10 a menudo se consideran pruebas sólidas (Kass y Raftery, 1995) [02].

Raftery y Dean (2006) [04] proponen un algoritmo de búsqueda codicioso. En cada etapa, busca la variable para agregar la que más mejora el agrupamiento según lo medido por BIC, y luego evalúa si una de las variables del agrupamiento actual puede eliminarse. En cada etapa, se elige la mejor combinación de número de *clusters* y modelo de agrupamiento. El algoritmo se detiene cuando no es posible una mejora local.

Este procedimiento lo encontramos disponible en R, en el paquete ***clustvarsel*** de Scrucca y Raftery (2018) [01]. https://cran.r-project.org/web/packages/clustvarsel/clustvarsel.pdf

Como vemos, este método en caso de $n > p$ supone una selección de variables obteniendo un sistema disperso, por un mecanismo que es análogo a la selección por pasos (*Stepwise selection*) de variables.

Sin embargo, en el caso de sistemas de alta dimensión $p \gg n$, nuevamente es necesaria una factorización por componentes principales previa sobre los que se actuaría la selección, aunque este método resuelve un aspecto importante en la selección de componentes principales.

Chang (1983) [01] demostró que la práctica de reducir los datos a los componentes principales que explican la mayor variabilidad antes de la agrupación no está justificada en general. Mostró que los componentes principales con los valores propios más grandes no necesariamente contienen la mayor cantidad de información sobre la estructura del *cluster*, y que tomar un subconjunto de componentes principales puede conducir a una pérdida importante de información sobre los grupos en los datos.

La ventaja de este método de agrupamiento es que la selección de componentes principales no se realiza en base a la mayor variabilidad, sino en base a la selección automática los componentes principales que son más útiles al agrupamiento.

COSA. Agrupamiento de Objetos en Subconjuntos de Atributos

Friedman y Meulman (2004) [01] proponen agrupar objetos en subconjuntos de atributos (*Clustering Objects on Subsets of Attributes*, COSA). Si C_k denota los índices de las observaciones en el $k-$ésimo de *K clusters*. Entonces, el criterio COSA es

$$\min_{C_1,C_2,\cdots,C_K,\boldsymbol{w}} \left\{ \sum_{k=1}^{K} a_k \sum_{i,i' \in C_k} \sum_{j=1}^{p} \left(w_j d_{i,i',j} + \lambda w_j log w_j \right) \right\} \tag{6}$$

$$sujeto\ a \sum_{j=1}^{p} w_j = 1, \qquad w_j \geq 0 \qquad \forall j$$

En realidad, ésta es una versión simplificada de la propuesta de COSA, que permite diferentes ponderaciones de características dentro de cada *cluster*. Aquí, a_k es alguna función del número de elementos en el *cluster* k, $\boldsymbol{w} \in \mathbb{R}^p$ es un vector de pesos de características, y $\lambda \geq 0$ es un parámetro de ajuste. Se puede ver que este criterio está relacionado con una versión ponderada del agrupamiento $k - means$.

Desafortunadamente, esta propuesta tampoco produce realmente un agrupamiento disperso, ya que todas las variables tienen pesos distintos de cero para $\lambda > 0$. Se propone una extensión de (6) para generalizar el método a otros tipos de agrupamiento, como el agrupamiento jerárquico. El algoritmo de optimización propuesto es bastante complejo e involucra múltiples parámetros de ajuste.

Hasta aquí en el caso de sistemas $n > p$ hemos descrito métodos de agrupamiento orientados a la reducción de dimensionalidad (Componentes Principales PCA, Factorización Matricial No Negativa) y se han descrito métodos que aportan mejoras como facilidad en la dispersión/reducción de componentes principales (Combinación de Modelos), una mejor selección de los compontes principales de cara al agrupamiento (Selección de variables en agrupamiento basado en modelos) o implementar una doble ponderación (a *cluster* y a componentes principales) que permite regular el modelo de agrupamiento. Estas dos últimas técnicas conllevan una regularización al aplicar penalizaciones por complejidad.

Obviando que en este caso no se trata de sistema de alta dimensión, estos métodos no generan modelos realmente dispersos en cuanto a variables (p) ya que no se produce una reducción efectiva de variables (cada componente principal es una combinación lineal de variables).

Sin embargo, podemos también abordar sistemas $p \approx n$ y $p \geq n$, con los métodos descritos, pero con la necesidad de la aplicación previa de una factorización por componentes principales PCA, con lo que nos volvemos a situar en modelos no realmente dispersos en cuanto a características.

Es decir, obtenemos una reducción de dimensionalidad, pero no un modelo disperso.

ESTRUCTURA DEL AGRUPAMIENTO DISPERSO

Para abordar el agrupamiento disperso propiamente dicho (selección de un conjunto reducido de variables definitorias del modelo) Witten y Tibshirani (2010) [02] proponen una versión simplificada de (6), implementada en el paquete de R ***sparcl*** de Witten y Tibshirani (2013) [01]. Es un marco general que se puede aplicar para obtener versiones dispersas de varios métodos de agrupamiento. Los algoritmos resultantes son eficientes incluso cuando p es bastante grande y contemplan la regularización del modelo al aplicar penalizaciones encaminadas a mejorar la selección de variables.

Supongamos que deseamos agrupar n observaciones sobre dimensiones p; recordemos que $\boldsymbol{X}$ es de dimensión $n \times p$. Si $\boldsymbol{X}_j \in \mathbb{R}^n$ denota la característica j. Muchos métodos de agrupamiento pueden expresarse como un problema de optimización de la forma

$$\max_{\boldsymbol{\Theta} \in D} \left\{ \sum_{j=1}^{p} f_j(\boldsymbol{X}_j, \boldsymbol{\Theta}) \right\} \tag{7}$$

Donde $f_j(\boldsymbol{X}_j, \boldsymbol{\Theta})$ es una función que afecta solo a la característica $j - ésima$ de los datos, y $\boldsymbol{\Theta}$ es un parámetro restringido en un conjunto D.

El agrupamiento $k - means$ y el jerárquico son dos de estos ejemplos, (Con $k - means$, por ejemplo, f_j resulta ser la suma de cuadrados entre el *cluster* para la característica j, y $\boldsymbol{\Theta}$ es una partición de las observaciones en K conjuntos disjuntos). Definimos el agrupamiento disperso como la solución al problema.

$$\max_{\mathbf{w};\boldsymbol{\Theta} \in D} \left\{ \sum_{j=1}^{p} w_j f_j(\boldsymbol{X}_j, \boldsymbol{\Theta}) \right\} \tag{8}$$

$$sujeto\ a\ \|\boldsymbol{w}\|^2 \leq 1, \qquad \|\boldsymbol{w}\|_1 \leq s \qquad w_j \geq 0 \qquad \forall j$$

donde w_j es un peso correspondiente a la característica j y s es un parámetro de ajuste, $1 \leq s \leq \sqrt{p}$. Observaciones sobre (8):

1. Si $w_1 = w_2 = \cdots = w_p$ en (8), entonces el criterio se reduce a (7).

2. En L_1, o LASSO, la penalización resulta en dispersión para valores pequeños del parámetro de ajuste s: es decir, algunos de los w_j serán iguales a cero. La penalización de L_2 también cumple una función importante, ya que, sin ella, en la mayoría de los casos, un elemento de $\boldsymbol{w}$ sería distinto de cero en general. También se utilizó un criterio que involucra un objetivo lineal restringido tanto para una restricción L_1 como para una restricción L_2 que fue usado en Witten, Tibshirani y Hastie (2009) [01].

3. El valor de w_j puede interpretarse como la contribución de la característica j para el agrupamiento disperso resultante: un gran valor de w_j indica una característica que contribuye en gran medida, y $w_j = 0$ significa que la característica j no está involucrada en el agrupamiento en *cluster*.

4. En general, para que la formulación (8) resulte en un agrupamiento disperso no trivial, es necesario que $f_j(\boldsymbol{X}_j, \boldsymbol{\Theta}) > 0$ para algunos o todos j. Es decir, si $f_j(\boldsymbol{X}_j, \boldsymbol{\Theta}) \leq 0$, entonces $w_j = 0$. Si $f_j(\boldsymbol{X}_j, \boldsymbol{\Theta}) > 0$, entonces la restricción de no negatividad en w_j no tiene efecto.

Optimizamos (8) utilizando un algoritmo iterativo: manteniendo $\boldsymbol{w}$ fijo, optimizamos (8) con respecto a $\boldsymbol{\Theta}$, y manteniendo $\boldsymbol{\Theta}$ fijo, optimizamos (8) con respecto a $\boldsymbol{w}$. En general, no logramos un óptimo global de (8) utilizando este enfoque iterativo; sin embargo, tenemos la garantía de que cada iteración aumenta la función objetivo. La primera optimización normalmente implica la aplicación de un procedimiento de agrupamiento en *cluster* estándar a una versión ponderada de los datos. Para optimizar (8) con respecto a $\boldsymbol{w}$ con $\boldsymbol{\Theta}$ mantenido fijo, observamos que el problema puede reescribirse como

$$\max_{\boldsymbol{w}} \{\boldsymbol{w}^T \boldsymbol{a}\} \tag{9}$$

$$sujeto\ a\ \|\boldsymbol{w}\|^2 \leq 1, \qquad \|\boldsymbol{w}\|_1 \leq s \qquad w_j \geq 0 \qquad \forall j$$

donde $a_j = f_j(\boldsymbol{X}_j, \boldsymbol{\Theta})$. Esto se resuelve fácilmente mediante un umbral suave, como se detalla a continuación.

Proposición. La solución al problema convexo (9) es $\boldsymbol{w} = \frac{S(a_+,\Delta)}{\|S(a_+,\Delta)\|_2}$, donde x_+ denota la parte positiva de x y donde $\Delta = 0$, si eso resulta en $\|\boldsymbol{w}\|_1 \leq s$; de lo contrario, se elige $\Delta > 0$ para obtener $\|\boldsymbol{w}\|_1 = s$. Aquí, S es el operador de umbral suave, definido como $S(x, c) = sign(x)(|x| - c)_+$. Hemos asumido que hay un elemento máximo único de $\boldsymbol{a}$, y que $1 \leq s \leq \sqrt{p}$.

La solución del problema convexo contempla las condiciones Karush-Kuhn-Tucker (KKT), Dado el problema de optimización

$$\begin{cases} \min f(\boldsymbol{x}) \\ sujeto\ a \quad g_i(\boldsymbol{x}) \leq 0 \qquad i = 1,2,\cdots,m \\ sujeto\ a \quad h_j(\boldsymbol{x}) = 0 \qquad j = 1,2,\cdots,l \end{cases}$$

Con $f, h_i, g_i: A \to \mathbb{R}$ funciones de clase $\mathcal{C}^1(A)$ y $A \subseteq \mathbb{R}^n$ un conjunto abierto. Diremos que $\boldsymbol{x}^* = (x_1^*, x_2^*, \cdots, x_n^*)$ es un punto de Karush-Kuhn-Tucker para dicho problema si y solo si $\exists\ \lambda_1, \lambda_2, \cdots, \lambda_m, \mu_1, \mu_2, \cdots, \mu_m$ de forma que se cumplen la siguientes condiciones:

Estacionariedad

$$\nabla f(x^*) + \sum_{i=1}^{m} \mu_i \nabla g_i(x^*) + \sum_{j=1}^{m} \lambda_j \nabla h_j(x^*) = 0$$

Factibilidad primaria

$$g_i(x^*) \leq 0, \qquad i = 1,2,\cdots,m$$
$$h_j(x^*) = 0, \qquad j = 1,2,\cdots,l$$

Factibilidad dual

$$\mu_i \geq 0, \qquad i = 1,2,\cdots,m$$

Holgura complementaria

$$\mu_i g_i(x^*) = 0, \qquad i = 1,2,\cdots,m$$

Cuando $m = 0$, es decir, cuando no hay restricción de inecuación, la condición KKT se denomina condición de Lagrange y los multiplicadores KKT se denominan multiplicadores de Lagrange.

GAP ESTADÍSTICO.

Tibshirani et al. (2001) [01] propusieron un método para estimar el número óptimo de *clusters*. La técnica se puede emplear luego de aplicar cualquier algoritmo de agrupamiento. Puesto que forma parte del proceso de solución completa en los siguientes métodos de agrupamiento disperso que se describirán (*k-means* y jerárquico) se describe en este apartado el procedimiento propuesto para encontrar el número de *clusters* en una población que denominaron "GAP estadístico".

Supongamos que tenemos n observaciones independientes, cada una con p características. Sea $\boldsymbol{x}_i$ con elementos (x_{ij}), $i = 1, 2, \cdots, n$ y $j = 1, 2, \cdots, p$ y sea $d(\boldsymbol{x}_i, \boldsymbol{x}_{i'})$ la distancia entre las observaciones $\boldsymbol{x}_i$ y $\boldsymbol{x}_{i'}$. Consideremos como distancia el cuadrado de la distancia euclídea, es decir

$$d(\boldsymbol{x}_i, \boldsymbol{x}_{i'}) = \sum_{j=1}^{p} \left(x_{ij} - x_{i'j}\right)^2$$

Supongamos que como resultado de un análisis hemos obtenido k *clusters*, $C_1, C_2, \cdots, C_k$; donde C_r denota el conjunto de índices de las observaciones en el *cluster* r y $n_r = |C_r|$ es la cantidad de elementos del conjunto C_r. Sea

$$D_r = \sum_{i,i' \in C_r} d(\boldsymbol{x}_i, \boldsymbol{x}_{i'})$$

la suma de las distancias entre los elementos del *cluster* r tomados de por parejas (notemos que se cuenta dos veces cada distancia) y sea

$$W_k = \sum_{r=1}^{k} \frac{1}{2n_r} D_r$$

W_k es función de la cantidad de *clusters* k y mide la dispersión dentro de los *clusters*. Observemos que esta medida, bajo cualquier método razonable de agrupamiento, es monótona decreciente a medida que k crece, y se va contrayendo-acercándose cada vez más hacia el valor 0, lo cual es razonable ya que la varianza intra-*cluster* va disminuyendo cuando aumenta la cantidad de grupos. Cuando d es la distancia Euclídea, W_k es la suma de cuadrados dentro de *clusters*; una medida de la dispersión de las observaciones, dentro de los *clusters*.

La idea de la propuesta es comparar, para cada k, el valor de $log(W_k)$ con su esperanza bajo una distribución nula de referencia en la que asumimos que no hay *clusters*. Se define

$$Gap_n(k) = E_n^*[log(W_k)] - log(W_k)$$

donde E_n^* es la esperanza para muestras de tamaño n extraídas de la distribución de referencia. Un primer estimador del número de *clusters* en la distribución de la que proviene la muestra es el valor $\hat{k}$, que maximiza $Gap_n(k)$ después de tener en cuenta la distribución de referencia.

El problema entonces consiste en proponer una distribución de referencia apropiada y construir la distribución de muestreo del estadístico gap.

AGRUPAMIENTO DISPERSO K –*MEANS*

El agrupamiento $K-means$ minimiza la suma de cuadrados dentro del *cluster* (WCSS, *Within Cluster Sum of Square*). Es decir, busca dividir las n observaciones en K conjuntos, o *clusters*, de manera que el WCSS

$$\sum_{k=1}^{K}\frac{1}{n_k}\sum_{i,i'\in C_k}\sum_{j=1}^{p}d_{i,i',j} \tag{10}$$

es mínimo, donde n_k es el número de observaciones en el *cluster* k y C_k contiene los índices de las observaciones en el *cluster* k. En general, $d_{i,i',j}$ puede denotar cualquier medida de disimilitud entre las observaciones i e i' a lo largo de la característica j. Sin embargo, tomaremos $d_{i,i',j}=\left(X_{ij}-X_{i'j}\right)^2$; por esta razón, nos referimos a (10) como la suma de cuadrados dentro del *cluster*. Si definimos la suma de cuadrados entre *clusters* (BCSS) como

$$\sum_{j=1}^{p}\left(\frac{1}{n}\sum_{i=1}^{n}\sum_{i'=1}^{n}d_{i,i',j}-\sum_{k=1}^{K}\frac{1}{n_k}\sum_{i,i'\in C_k}d_{i,i',j}\right) \tag{11}$$

entonces, minimizar el WCSS es equivalente a maximizar el BCSS.

Se podría tratar de desarrollar un método para el agrupamiento disperso *k-means* optimizando un WCSS ponderado, sujeto a restricciones en los pesos y por consiguiente regularizando, es decir,

$$\max_{C_1,C_2,\cdots,C_K,\boldsymbol{w}}\left\{\sum_{j=1}^{p}w_j\left(-\sum_{k=1}^{K}\frac{1}{n_k}\sum_{i,i'\in C_k}d_{i,i',j}\right)\right\} \tag{12}$$

$$sujeto\ a\quad \|\boldsymbol{w}\|^2\le 1,\qquad \|\boldsymbol{w}\|_1\le s,\qquad w_j\ge 0\qquad \forall j$$

Aquí, s es un parámetro de ajuste. Como cada elemento de la suma ponderada es negativo, el máximo se produce cuando todos los pesos son cero, independientemente del valor de s. Esta no es una solución interesante. En su lugar, maximizamos un BCSS ponderado, sujeto a restricciones en los pesos. El criterio de agrupamiento disperso de *K-means* es el siguiente:

$$\max_{C_1,C_2,\cdots,C_K,\boldsymbol{w}} \left\{ \sum_{j=1}^{p} w_j \left(\frac{1}{n} \sum_{i=1}^{n} \sum_{i'=1}^{n} d_{i,i',j} - \sum_{k=1}^{K} \frac{1}{n_k} \sum_{i,i' \in C_k} d_{i,i',j} \right) \right\} \tag{13}$$

$$sujeto\ a \quad \|\boldsymbol{w}\|^2 \leq 1, \qquad \|\boldsymbol{w}\|_1 \leq s, \qquad w_j \geq 0 \qquad \forall j$$

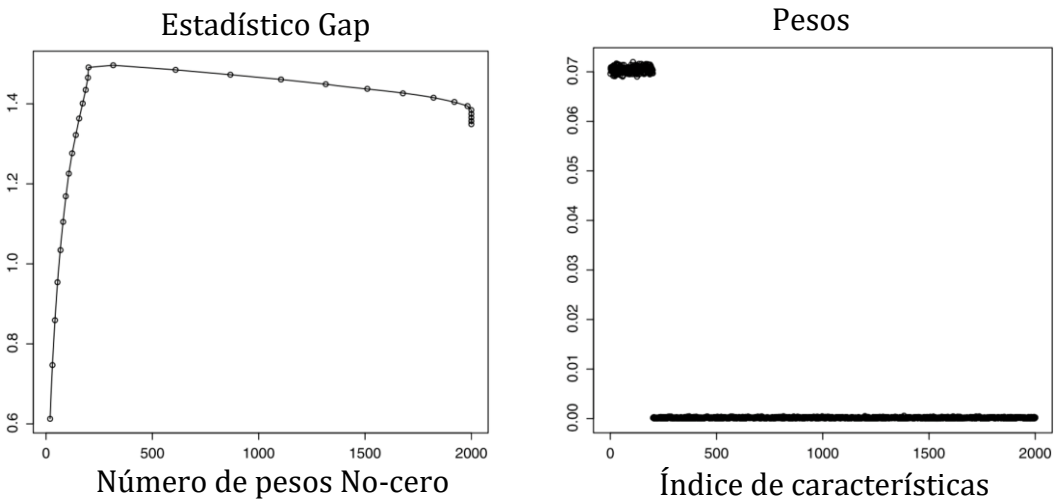
La regularización mediante los pesos hará que sean dispersos para una elección apropiada del parámetro de ajuste s, que debe satisfacer $1 \leq s \leq \sqrt{p}$. Tengamos en cuenta que si $w_1 = w_2 = \cdots = w_p$, entonces (10) simplemente se reduce al criterio estándar de agrupamiento *K-means*. Observamos que (11) y (12) son casos específicos de (7) y (8) donde $\boldsymbol{\Phi} = (C_1, C_2, \cdots, C_K)$,

$$f_j(\boldsymbol{X}_j, \boldsymbol{\Theta}) = \frac{1}{n} \sum_{i=1}^{n} \sum_{i'=1}^{n} d_{i,i',j} - \sum_{k=1}^{K} \frac{1}{n_k} \sum_{i,i' \in C_k} d_{i,i',j}$$

y ($\boldsymbol{\Phi} \in D$) donde D denota los conjuntos de todas las particiones posibles de las observaciones en *K clusters*.

El criterio (12) asigna un peso a cada característica, en función del aumento en BCSS que la característica puede aportar. Primero, consideramos el criterio con los pesos fijos $w_1, w_2, \cdots, w_p$. Reduciéndolo a un problema de agrupamiento, utilizando una medida ponderada de disimilitud. Segundo, consideramos el criterio con los *clusters* fijos $C_1, C_2, \cdots, C_K$. Luego, se asignará un peso a cada característica según el BCSS de esa característica. Las características con BCSS más grandes tendrán pesos más grandes.

Como ejemplo de aplicación Witten y Tibshirani (2010) [03] presentan un ejemplo con 6 clases de igual tamaño, donde $n = 120, p = 2000$ y 200 características diferentes entre clases, en la Gráfica 26 podemos observar el estadístico gap en función del número de características distintas de cero (izqu) así como el peso obtenido para las distintas características (drch), donde observamos claramente la dispersión de características obtenida por este método de regularización aplicado al agrupamiento *k-mean*



Gráfica 26. Aplicación del método *k-mean* disperso.
Gap estadístico (izqu). Peso asignado a las características (drch).
Tomado de Witten y Tibshinari (2010), página 717 [04].

AGRUPAMIENTO JERÁRQUICO DISPERSO

El agrupamiento jerárquico produce un dendrograma que representa un conjunto anidado de agrupaciones: según el lugar donde se corta el dendrograma, pueden aparecer entre 1 y n agrupaciones. Se podría aplicar un método para el agrupamiento jerárquico disperso cortando el dendrograma a cierta altura y maximizando una versión ponderada de la BCSS (suma de cuadrados intercluster) resultante. Sin embargo, no está claro dónde se debe cortar el dendrograma.

El agrupamiento jerárquico toma como entrada una matriz $n \times n$ de disimilitud $\boldsymbol{U}$. El agrupamiento en *cluster* puede usar cualquier tipo de enlace: completo (*complete*), promedio (*average*) o único (*single*). Si $\boldsymbol{U}$ es la matriz de disimilitud general $\{\sum_j d_{i,i',j}\}_{i,i'}$, entonces resulta el agrupamiento jerárquico estándar.

Para el agrupamiento disperso comenzamos exponiendo la matriz de disimilitud global $\{\sum_j d_{i,i',j}\}_{i,i'}$ en la forma en la forma (7), es decir,

$$\max_{\boldsymbol{\Theta} \in D} \left\{ \sum_{j=1}^{p} f_j(\boldsymbol{X}_j, \boldsymbol{\Theta}) \right\}$$

Puesto que la escala de la matriz de disimilitud en un factor no afecta la forma del dendrograma resultante, ignoramos las constantes de proporcionalidad y consideramos el criterio

$$\max_{\boldsymbol{U}} \left\{ \sum_j \sum_{i,i'} d_{i,i',j} U_{i,i'} \right\} \quad sujeto\ a \quad \sum_{i,i'} U_{i,i'}^2 \leq 1 \tag{14}$$

Observamos que (14) toma la forma (7) con $\boldsymbol{\Theta} = \boldsymbol{U}$, $f_j(\boldsymbol{X}_j, \boldsymbol{\Theta}) = \sum_{i,i'} d_{i,i',j} U_{i,i'}$ y $\boldsymbol{\Theta} \in D$ correspondiente a $\sum_{i,i'} U_{i,i'}^2 \leq 1$

Supongamos que $\boldsymbol{U}^*$ optimice (14). No es difícil demostrar que $U^*_{i,i'} \propto \sum_j d_{i,i',j}$ y, por lo tanto, realizar un agrupamiento jerárquico en $\boldsymbol{U}^*$ da como resultado agrupamiento jerárquico estándar. Entonces podemos pensar en el agrupamiento jerárquico estándar como resultado del criterio (14). Para obtener la dispersión en las características, modificamos (14) multiplicando cada elemento de la suma sobre j por un peso w_j, sujeto a restricciones en los pesos, aplicado un criterio de la forma (8)

$$\max_{\mathbf{w};\boldsymbol{\Theta}\in D}\left\{\sum_{j=1}^{p} w_j f_j(\boldsymbol{X}_j,\boldsymbol{\Theta})\right\}$$

$$sujeto\ a\ \ \|\boldsymbol{w}\|^2 \le 1, \qquad \|\boldsymbol{w}\|_1 \le s, \qquad w_j \ge 0 \qquad \forall j$$

que conduce a una matriz de disimilitud ponderada que es dispersa en las características (j).

$$\max_{\boldsymbol{w},\boldsymbol{U}}\left\{\sum_j w_j \sum_{i,i'} d_{i,i',j} U_{i,i'}\right\} \tag{15}$$

$$sujeto\ a \sum_{i,i'} U^2_{i,i'} \le 1, \qquad \|\boldsymbol{w}\|^2 \le 1, \qquad \|\boldsymbol{w}\|_1 \le s, \qquad w_j \ge 0 \qquad \forall j$$

La $\boldsymbol{U}^{**}$ que optimiza (15) es proporcional a $\left\{\sum_j d_{i,i',j}\, w_j\right\}_{i,i'}$.

Puesto que **w** es disperso para valores pequeños del parámetro de ajuste s, $\boldsymbol{U}^{**}$ involucra solo un subconjunto de las características, por lo que al realizar un agrupamiento jerárquico en $\boldsymbol{U}^{**}$ se obtiene un agrupamiento jerárquico disperso mediante el procedimiento de regularización con las restricciones indicadas

Tenemos pues en (15) *el criterio de agrupamiento jerárquico disperso*.

De ello se deduce directamente que (15) toma la forma (8), donde el parámetro de ajuste s se sitúa entre 1 y $\sqrt{p}$.

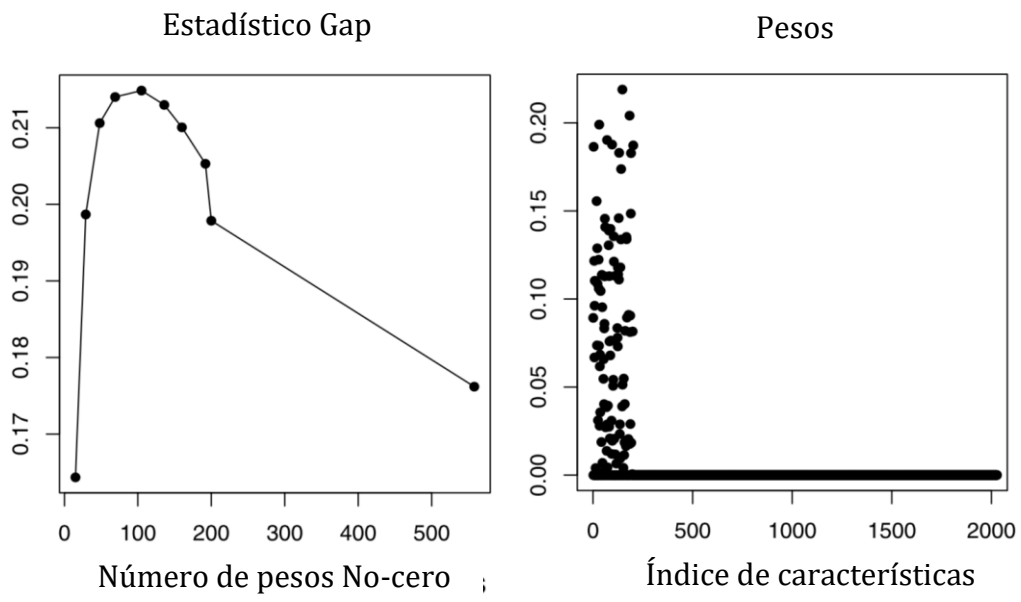
Observamos que (15) es bi-convexo en $\boldsymbol{U}$ y $\boldsymbol{w}$: con $\boldsymbol{w}$ fijo, es convexo en $\boldsymbol{U}$ y con $\boldsymbol{U}$ fijo, convexo en $\boldsymbol{w}$. Esto supone un inconveniente ya que no se garantiza la convergencia a un óptimo global.

La resolución del agrupamiento disperso jerárquico en base a lo anterior podemos observarla en el Anexo I.

Donde $\max_{\boldsymbol{w},\boldsymbol{u}}\{\boldsymbol{u}^T\boldsymbol{D}\boldsymbol{w}\}$ es básicamente el criterio de componentes principales dispersos (SPC) de Witten, Tibshirani y Hastie (2009) [02], con una restricción de no negatividad adicional en $\boldsymbol{w}$. Si $d_{i,i',j} \geq 0$, como suele ser el caso, la restricción de no negatividad se puede eliminar. De hecho, los pasos 1 y 2 del algoritmo para agrupamiento jerárquico disperso son esencialmente el algoritmo SPC de Witten, Tibshirani y Hastie (2009) [03].

Puesto que $\boldsymbol{u} \propto \boldsymbol{D}\boldsymbol{w}$ se puede reescribir como una matriz $\boldsymbol{U}$, $n \times n$, que es una combinación lineal ponderada de las matrices de disimilitud de las características. Cuando s es pequeño, entonces algunos w_j serán igual a cero, por lo que $\boldsymbol{U}$ dependerá solo de un subconjunto de las características. Finalmente, realizamos un agrupamiento jerárquico en $\boldsymbol{U}$ para obtener un dendrograma que se basa solo en un subconjunto de características.

Igual que en el caso del agrupamiento *k-mean* disperso anterior Witten y Tibshinari (2010) [05] aplicaron el procedimiento de agrupamiento jerárquico disperso a los datos anteriores obteniendo el resultado que se muestra en la Gráfica 27.



Gráfica 27. Aplicación del método jerárquico disperso.
Gap estadístico (izqu). Peso asignado a las características (drch).
Tomado de Witten y Tibshinari (2010), página 721 [06].

AGRUPAMIENTO CONVEXO

El método de agrupamiento de $k-means$ y también su generalización dispersa conducen a problemas que no son convexos en conjunto, por lo que es difícil garantizar que se haya logrado una solución global.

El agrupamiento $k-means$ presenta dos problemas importantes, en primer lugar, hemos de definir previamente el número de *cluster* k y en segundo lugar la sensibilidad al punto de inicio (centroides de partida) como se pone de manifiesto en Peña et al. (1999) [01]. Podemos ver un ejemplo en el Anexo II.

La sensibilidad a la inicialización se aborda mediante la implementación de algoritmos convexos.

Lashkari y Golland

Encontramos propuestas de algoritmos convexos para $k-means$ en Lashkari y Golland (2008) [01] que proponen como función a maximizar

$$l\left(\{q_j\}_{j=1}^{n};\mathcal{X}\right)=\frac{1}{n}\sum_{i=1}^{n}log\sum_{j=1}^{n}q_j f_j(\boldsymbol{x}_i)=\frac{1}{n}\sum_{i=1}^{n}log\left[\sum_{j=1}^{n}q_j e^{-\beta d_\phi(x_i,x_j)}\right]+const$$

siendo

- $\mathcal{X}=\{\boldsymbol{x}_1,\boldsymbol{x}_2,\cdots,\boldsymbol{x}_n\}\subset\mathbb{R}^d$
- $\{q_j\}_{j=1}^{n}\geq 0;\ \sum_{j=1}^{n}q_j=1$. Un conjunto de pesos o ponderación de cada *cluster*
- β factor de densidad del *cluster*. A mayor valor de β mayor número de *clusters*
- $d_\phi(\boldsymbol{x}_i,\boldsymbol{x}_j)$ una divergencia o distancia Bregman. Siendo ϕ una función diferenciable tal que $d_\phi(\boldsymbol{x}_i,\boldsymbol{x}_j)=\phi(\boldsymbol{x}_i)-\phi(\boldsymbol{x}_j)-\langle\boldsymbol{x}_i-\boldsymbol{x}_j,\nabla\phi(\boldsymbol{x}_j)\rangle$, con $\nabla\phi(\boldsymbol{x}_j)$ el vector gradiente de ϕ evaluado en $\boldsymbol{x}_j$. Baneerjee et al (2005) [01].

Nowozin y Bakir

En Nowozin y Bakir (2008) [01] la propuesta es la minimización de una función objetivo $\Omega(\gamma, \rho)$ condicionada por una serie de restricciones. Sea $k_{\boldsymbol{z}}(\cdot)$ un núcleo de suavizado no negativo centrado en $z \in \mathcal{Z}$, con $\mathcal{Z} \subseteq \mathcal{X}$. Sea $\{x_i\}_{i=1}^{N}$, $\boldsymbol{x}_i \in \mathcal{X}$

$$\min_{q,\gamma,\rho} \Omega(\gamma, \rho) \tag{16}$$

Sujeto a

$$\int_{\mathcal{Z}} q_{\boldsymbol{z}} k_{\boldsymbol{z}}(\boldsymbol{x}_i) d\boldsymbol{z} = \gamma_i : \alpha_i, \qquad i = 1, 2, \cdots, N \tag{17}$$

$$\rho \leq \gamma_i : \omega_i, \qquad i = 1, 2, \cdots, N \tag{18}$$

$$q_{\boldsymbol{z}} \geq 0 : \mu_{\boldsymbol{z}}, \quad \forall \boldsymbol{z} \in \mathcal{Z} \tag{19}$$

$$\int_{\mathcal{Z}} q_{\boldsymbol{z}} d\boldsymbol{z} = 1 : \ \sigma \tag{20}$$

donde $\alpha, \omega, \mu, \sigma$ son los multiplicadores de Lagrange para las restricciones respectivas.

- La restricción (17) evalúa una combinación convexa de respuestas para cada muestra. γ_i contiene la respuesta combinada para la muestra $\boldsymbol{x}_i$.

- La restricción (18) identifica - si $\nabla_p \Omega(\gamma, \rho) < 0$ - la respuesta más baja entre todas las muestras. El valor de la respuesta combinada más baja es ρ.

- Las restricciones (19) y (20) definen la combinación simplex de las funciones de respuesta.

Las funciones objetivo $\Omega(\gamma, \rho)$ propuestas son cuatro:

$$\Omega(\gamma, \rho) = -\rho$$

$$\Omega(\gamma, \rho) = -\frac{1}{N}\sum_{i=1}^{N} log(\gamma_i)$$

$$\Omega(\gamma, \rho) = -\rho + \frac{C}{N}\sum_{i=1}^{N}(\gamma_i - \rho)^2$$

$$\Omega(\gamma, \rho) = -\frac{1}{N}\sum_{i=1}^{N}\gamma_i + \frac{C}{N}\sum_{i=1}^{N}\left(\gamma_i - \frac{1}{N}\sum_{i=1}^{N}\gamma_i\right)^2$$

Hocking et al.

Hocking et al. (2011) [01] propusieron un planteamiento (algoritmo) de convexidad denominado ***Clusterpath***. Partiendo de la matriz de datos $\boldsymbol{X} \in \mathbb{R}^{n\times p}$ el óptimo global en base a la convexidad de la función se obtiene mediante

$$\min_{\boldsymbol{\alpha}\in\mathbb{R}^{n\times p}} f_q(\boldsymbol{\alpha}, \boldsymbol{X}) = \frac{1}{2}\|\boldsymbol{\alpha} - \boldsymbol{X}\|_F^2 + \lambda\sum_{i<j} w_{ij}\left\|\alpha_i - \alpha_j\right\|_q \tag{21}$$

donde:

- $\boldsymbol{\alpha}$ es la matriz de centroides
- $\|\cdot\|_q$, $q \in \{1,2,\infty\}$ es la norma L_q sobre $\mathbb{R}^p$, constituye la penalización entre centroid es
- λ es una constante de ajuste positiva
- $\|\boldsymbol{A}\|_F$ es la norma Frobenius $\|\boldsymbol{A}\|_F = \left(\sum_{i=1}^{n}\sum_{j=1}^{p}\left|A_{ij}\right|^2\right)^{1/2}$
- $\sum_{i<j} = \sum_{i}^{n-1}\sum_{j=i+1}^{n}$
- w_{ij} es el peso específico asignado a la relación entre dos centroides, con el que podremos controlar el camino geométrico de la solución, por ejemplo se puede determinar el peso en función de la distancia especificando dicho peso como $w_{ij} = e^{-\gamma\|X_i - X_j\|_2^2}$, que tendrá el efecto de decaer con la distancia, o bien asignando $w_{ij} = 1$ donde no se asigna prioridad a ningún centroide.

Lindsten et al.

Lindsten et al. (2011) [01] formularon la tarea de agrupamiento en *clusters* denominada ***SON*** (*Sum of Norms*) también como un problema de optimización convexo. Dados N puntos $\{x_j\}_{j=1}^{N}$ con $x_j \in \mathbb{R}^p$, sugirieron minimizar el criterio convexo.

$$\min_{\mu_1, \mu_2, \cdots, \mu_N} \left\{ \sum_{j=1}^{N} \|\boldsymbol{x}_j - \boldsymbol{\mu}_j\|^2 + \lambda \sum_{j=2}^{N} \sum_{i<j} \|\boldsymbol{\mu}_i - \boldsymbol{\mu}_j\|_q \right\} \tag{22}$$

donde λ es una constante de ajuste positiva, y $\boldsymbol{\mu}_i$ es el centroide asociado a $\boldsymbol{x}_i$. Lindsten et al. (2011) [02] consideran una norma de penalización L_q para las diferencias $\boldsymbol{\mu}_i - \boldsymbol{\mu}_j$

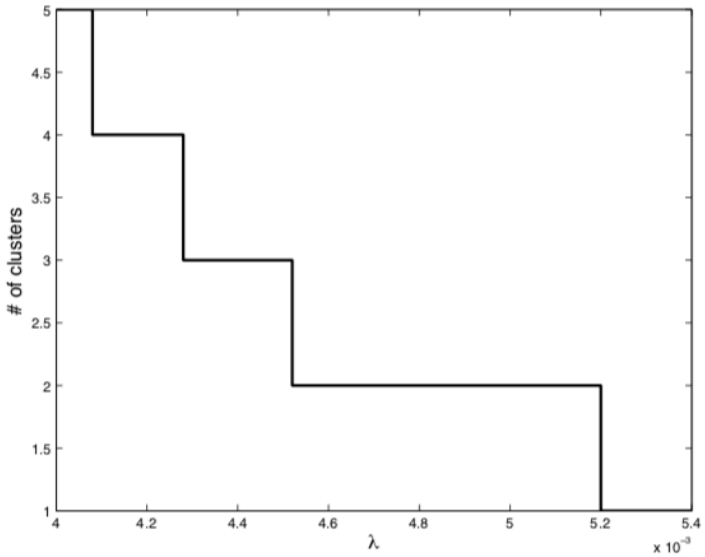
La función objetivo tiene cierta similitud con LASSO fusionado (Tibshirani et al., 2005) [04], cuando se usa la penalización L_1 en la definición (22), recuperamos un caso especial del *General Fused* LASSO (Hoefling, 2010 [01]; Tibshirani y Taylor, 2011 [01]).

Cuando $\lambda = 0$, el mínimo se alcanza cuando $\boldsymbol{x}_j = \boldsymbol{\mu}_j$, y cada punto ocupa un *cluster* único. A medida que aumenta λ, los centros de los *clusters* comienzan a unirse. Se dice que dos puntos $\boldsymbol{x}_i$ y $\boldsymbol{x}_j$ con $\boldsymbol{\mu}_i = \boldsymbol{\mu}_j$ pertenecen al mismo *cluster*. Para una λ suficientemente alta, todos los puntos se unen en un solo *cluster*.

Debido a que la función objetivo en la ecuación (21) y (22) es estrictamente convexa, posee un punto mínimo único para cada valor de λ. Por lo tanto, si trazamos la matriz de solución como una función de λ, entonces podemos identificar esos valores de λ dando $k - cluster$ para cualquier entero k entre 1 y N. En teoría, k puede disminuir en más de 1 cuando ciertos valores críticos de λ se superan. De hecho, cuando los puntos no están bien separados, observamos que muchos centroides se unirán abruptamente.

Continuando con el ejemplo del Anexo II, para aplicar el agrupamiento en *cluster* convexo, debemos elegir un valor para el parámetro de regularización λ. Una vez que este valor es fijo, el resultado del agrupamiento es independiente de la inicialización, debido a la convexidad de este método de agrupamiento. Sin embargo, el resultado dependerá, por supuesto, de λ.

El parámetro de regularización controlará la compensación entre el ajuste del modelo y el número de *clusters*. En la Gráfica 28, el número de *clusters* se representa como una función de λ. En el ejemplo del Anexo II cualquier λ entre $4.5 \cdot 10^{-3}$ y $5.1 \cdot 10^{-3}$ proporciona dos *clusters* y el "verdadero" S. Está claro que una de las principales dificultades para aplicar el agrupamiento convexo es encontrar un valor "bueno-adecuado" para λ.



Gráfica 28. Efecto de **λ** sobre el número de *clusters.*
Tomado de Lindsten et al. (2011), página 6 [03].

La elección de λ está relacionada con la decisión sobre la complejidad del modelo. Por lo tanto, métodos como la validación cruzada (ver, por ejemplo, Ljung (1999) [01]) pueden ayudar a encontrar un valor adecuado. Otra forma, sería resolver el problema para todos los valores de λ y dibujar un gráfico similar al de la Gráfica 28. A partir de este gráfico, vemos que el número de *clusters* permanece en dos para un amplio rango de valores de λ , lo que indica que hay evidencia de dos *clusters* en los datos.

Sin embargo, suponiendo que podamos resolver el problema para todos los valores de λ, una opción más sensata sería proporcionar el "conjunto de soluciones" completo al usuario. En otras palabras, en lugar de tomar una decisión difícil sobre λ (lo que implica, para un problema dado, una decisión difícil sobre el número de *clusters*), se dan todas las soluciones al usuario en función de λ. El hecho de si se debe o no tomar una decisión difícil sobre la cantidad de *clusters* se deja como una decisión a posteriori para el usuario.

Inevitablemente, es necesario que exista un equilibrio entre el ajuste del modelo y la complejidad de este, y es decisión del usuario decidir sobre este compromiso.

La idea de proporcionar un conjunto completo de soluciones, en lugar de una sola, aparece también en el agrupamiento jerárquico (véase, por ejemplo, Hastie et al. (2001) [01], Capítulo 14.3.12). La principal diferencia entre el agrupamiento convexo (con las rutas de solución en λ) y el agrupamiento jerárquico es que este último es intrínsecamente codicioso.

Para verlo tomemos, por ejemplo, un método aglomerativo (de abajo hacia arriba). Comenzando con un *cluster* para cada punto de datos, fusiona sucesivamente los dos *clusters* que minimizan la función objetivo al máximo y luego continúa de esa manera. Por lo tanto, una mala elección en una etapa temprana no se puede corregir y puede llevar a un mal resultado.

Este problema se ilustra en el ejemplo del Anexo IV, en el que el agrupamiento convexo se compara con el agrupamiento jerárquico y donde podemos ver la ventaja de este respecto al jerárquico.

Un efecto indeseado del agrupamiento convexo es que la penalización está influida por el tamaño de los *clusters*, lo que se traduce en una "preferencia" hacia *clusters* de distinto tamaño, veámoslo con un ejemplo en el Anexo V.

Una forma de eliminar la influencia de este problema es aplicar una ponderación basada en el núcleo a la regularización convexa.

Dado que la suma en el término de regularización se extiende sobre todos los pares de puntos, penalizará valores μ distintos incluso si los puntos de datos correspondientes están muy separados. Para evitar esto, podemos localizar la penalización de regularización agregando ponderaciones dependientes de los datos. El problema de optimización modificado es entonces,

$$\min_{\mu_1,\mu_2,\cdots,\mu_N}\left\{\sum_{j=1}^{N}\left\|\boldsymbol{x}_j-\boldsymbol{\mu}_j\right\|^2+\lambda\sum_{j=2}^{N}\sum_{i<j}\kappa(\boldsymbol{x}_i,\boldsymbol{x}_j)\left\|\boldsymbol{\mu}_i-\boldsymbol{\mu}_j\right\|_q\right\}$$

donde $\kappa(\boldsymbol{x}_i,\boldsymbol{x}_j)$ es un kernel local. Tengamos en cuenta que, dado que $\kappa(\boldsymbol{x}_i,\boldsymbol{x}_j)$ depende solo de los puntos de datos (fijos) $\{x_j\}_{j=1}^{N}$ y no de las variables de optimización $\{\mu_j\}_{j=1}^{N}$, no cambia la convexidad ni la dimensión del problema.

Por supuesto, se puede utilizar cualquier núcleo local (por ejemplo, gaussiano). Sin embargo, desde un punto de vista computacional, puede ser beneficioso usar un *kernel* con soporte limitado, ya que esto puede reducir significativamente el número de términos distintos de cero en la suma de regularización. Este puede ser, por ejemplo, un *kNN- kernel* simple (k vecinos más cercanos), es decir,

$$\kappa(\boldsymbol{x}_i, \boldsymbol{x}_j) = \begin{cases} 1 & Si\ x_i \in kNN(x_j)\ o\ \ x_j \in kNN(x_i) \\ 0 & En\ otro\ caso \end{cases}$$

donde $kNN(x)$ es el conjunto de los k vecinos más cercanos de x.

Sin embargo, en función de las características del problema a analizar cualquier función de ponderación puede ser implementada, por ejemplo $\kappa(x_i, x_j) = e^{\{-\|x_i - x_j\|^2\}}$.

Los beneficios de esta formulación son múltiples. El agrupamiento convexo admite un algoritmo iterativo simple y rápido que está garantizado para converger a un mínimo global único. En contraste, se ha demostrado que el problema clásico de $k-means$ es *NP-hard* (Aloise et al., 2009 [01]; Dasgupta y Freund, 2009 [01]). Además, el algoritmo codicioso clásico para resolver el agrupamiento de $k-means$ a menudo queda atrapado en mínimos locales subóptimos (Forgy, 1965 [01]; Lloyd, 1982 [01]; MacQueen, 1967 [01]), como hemos comentado.

Otro problema en el agrupamiento en *clusters* es determinar la cantidad de *clusters*. El agrupamiento jerárquico aglomerativo (Gower y Ross, 1969 [01]; Johnson, 1967 [01]; Lance y Williams, 1967 [01]; Murtagh, 1983 [01]; Ward, 1963 [01]) soluciona el problema al calcular una ruta de agrupamiento completa. Sin embargo, los enfoques de aglomeración pueden ser computacionalmente exigentes y tienden a caer en mínimos locales subóptimos, ya que los eventos de coalescencia no se revierten, como hemos visto en el ejemplo del Anexo IV. El agrupamiento convexo alternativo considerado aquí realiza un agrupamiento continuo al igual que el LASSO (Chen et al., 1998 [01]; Tibshirani, 1996 [02]) realizando una selección continua de variables.

CAPÍTULO 3

EJEMPLOS CON MÉTODOS DE REGULARIZACIÓN

En el capítulo 1 hemos descrito los métodos de reducción de la complejidad de un modelo y nos hemos centrado en los métodos de regularización, los cuales por sus características están especialmente indicados para el tratamiento de sistemas de variables de alta dimensión donde $p \gg n$, asumiendo que solo en un subconjunto de variables son distintas de cero las cuales pueden determinar o modelar de forma adecuada el sistema, es decir, estamos considerando sistemas dispersos (*sparse systems*).

En un recorrido temporal[16] hemos descrito las principales estrategias de regularización como son *Ridge*[17], LASSO, y *Elastic Net*, además de profundizar en otros métodos que mejoran deficiencias y corrigen la aplicación de los anteriores (LASSO-Adaptativo, LASSO-Relajado) o bien se adaptan adecuadamente a las especificidades de algunos sistemas (LASSO-Grupal, LASSO-Fusionado).

En el capítulo 2 dentro de las técnicas de análisis multivariante hemos recorrido la aplicación de las técnicas de regularización a distintas estrategias que se utilizan en el aprendizaje estadístico no supervisado, concretamente en el agrupamiento (*Clustering*).

En este capítulo nos detendremos en mostrar ejemplos de la aplicación de las técnicas de regularización de forma general y en las técnicas de agrupamiento, observando la diferencia en resultados respecto de la no aplicación de la regularización en un conjunto de datos creado de alta dimensión.

---

16 *Ridge* (1962 / 1970), LASSO (1996), *Elastic Net* (2005).

17 Recordar que *Ridge* sensu stricto no supone una selección de variables sino una contracción de las mismas, aunque dado que aplica una penalización L_2 implica que se trata de un método de regularización.

El paquete que utilizaremos en el caso de los métodos de regularización de forma general es el paquete de R denominado ***glmnet***. desarrollado por Jerome Friedman, Trevor Hastie, Robert. Tibshirani. [https://cran.r-project.org/web/packages/glmnet/glmnet.pdf].

El paquete que aplicaremos específicamente a las técnicas de agrupamiento será el paquete de R ***sparcl***, cuyos autores son Daniela M. Witten y Robert Tibshirani. [https://cran.r-project.org/web/packages/sparcl/sparcl.pdf].

REGULARIZACIÓN LASSO Y *ELASTIC NET*

Como hemos comentado vamos a mostrar la aplicación de los principales métodos de regularización de una forma práctica, para ello vamos a considerar (construir) un sistema de alta dimensión sobre el que aplicaremos los métodos descritos.

Generamos un sistema $p \gg n$, con $p = 110$ y $n = 100$

```
> set.seed(122)                                    # Fijación de una "semilla"
> x <- matrix(rnorm(100*110), ncol = 110)
> x <- scale(x, TRUE, TRUE)
```

Para la comprobación de la regularización en regresión lineal, generamos también de forma aleatoria una variable dependiente.

```
> set.seed(523)                                    # Fijación de una "semilla"
> y <- rnorm(100*1)
> y <-scale(y, TRUE, TRUE)
```

Finalmente componemos los datos de partida sobre los que trabajar[18]. *Datos* es una matriz (100×111) cuya primera columna contiene la variable dependiente y y el resto está formada por 110 variables independientes p.

```
> Datos <- cbind(y,x)
```

En primer lugar, vemos que para la estimación de una regresión lineal tenemos una matriz indeterminada (subdeterminada), por lo que para obtener el modelo de regresión o bien podemos aplicar una reducción de dimensionalidad mediante componentes principales PCA o podemos descartar (en este ejemplo) la diez últimas variables independientes que presenten la mínima correlación con respecto a la variable dependiente. Los métodos de selección de variables vistos en el capítulo 1 nos permitirán obtener también una selección de las principales variables, pero a un mayor coste computacional.

[18] Se describe en detalle esta parte del código con objeto de facilitar la reproducción exacta del ejemplo.

Obtenemos los coeficientes de todas las variables del modelo sin regularización (con penalización $\lambda = 0$), es decir, por Mínimos Cuadrados Ordinarios (MCO).

```
> Modelo_MCO <- glmnet(x, y,lambda = 0)
```

El modelo generado es de complejidad (número de variables) evidente. Tabla 2.

β_{01}	0.059594234	β_{29}	-0.182656336	β_{57}	-0.153037136	β_{85}	-0.048487428
β_{02}	-0.466128475	β_{30}	-0.302268836	β_{58}	-0.326088947	β_{86}	-0.555605108
β_{03}	-0.108073597	β_{31}	0.546828629	β_{59}	-0.118825227	β_{87}	0.016556770
β_{04}	0.037545625	β_{32}	0.343115582	β_{60}	-0.120853877	β_{88}	-0.067652150
β_{05}	-0.128995200	β_{33}	-0.082737381	β_{61}	0.032382140	β_{89}	-0.109823374
β_{06}	-0.210766725	β_{34}	-0.196201988	β_{62}	0.100200811	β_{90}	0.009618065
β_{07}	-0.091575165	β_{35}	-0.180114810	β_{63}	0.124132988	β_{91}	-0.070584130
β_{08}	-0.192246426	β_{36}	-0.110356714	β_{64}	0.370267216	β_{92}	0.139147073
β_{09}	0.046142143	β_{37}	-0.195559955	β_{65}	-0.380324452	β_{93}	-0.157286451
β_{10}	-0.199856609	β_{38}	0.107747258	β_{66}	-0.580250902	β_{94}	0.095819212
β_{11}	0.105681342	β_{39}	-0.071733551	β_{67}	-0.154958233	β_{95}	-0.074542703
β_{12}	-0.299206529	β_{30}	-0.463624352	β_{68}	-0.042678938	β_{96}	0.348917326
β_{13}	-0.084779619	β_{41}	0.046818390	β_{69}	-0.086366438	β_{97}	0.085170272
β_{14}	-0.199165595	β_{42}	-0.026509949	β_{70}	0.138468996	β_{98}	0.156678503
β_{15}	0.493370245	β_{43}	-0.464147328	β_{71}	-0.075962499	β_{99}	0.385706740
β_{16}	-0.178067204	β_{44}	-0.206760705	β_{72}	0.478653751	β_{100}	-0.300984270
β_{17}	0.353444600	β_{45}	0.095365509	β_{73}	-0.187284885	β_{101}	0.259840727
β_{18}	-0.027990182	β_{46}	0.255225248	β_{74}	0.380487532	β_{102}	0.068176280
β_{19}	-0.089323300	β_{47}	-0.119999700	β_{75}	0.181416945	β_{103}	-0.083182302
β_{20}	-0.139484882	β_{48}	0.260569122	β_{76}	-0.207171886	β_{104}	-0.171041462
β_{21}	-0.111308656	β_{49}	-0.035458471	β_{77}	0.011453723	β_{105}	-0.126112908
β_{22}	0.229148259	β_{50}	-0.071695412	β_{78}	0.128487383	β_{106}	0.181771115
β_{23}	0.326217833	β_{51}	0.176759805	β_{79}	-0.018732242	β_{107}	-0.125946587
β_{24}	0.227030534	β_{52}	0.053560066	β_{80}	-0.129839526	β_{108}	-0.172514223
β_{25}	-0.276080689	β_{53}	-0.271494666	β_{81}	-0.382217168	β_{109}	0.147687914
β_{26}	0.116934562	β_{54}	-0.099125075	β_{82}	0.290671660	β_{110}	-0.074316652
β_{27}	0.220867099	β_{55}	-0.057677579	β_{83}	0.177477460		
β_{28}	0.160577242	β_{56}	0.391416837	β_{84}	0.043106097		

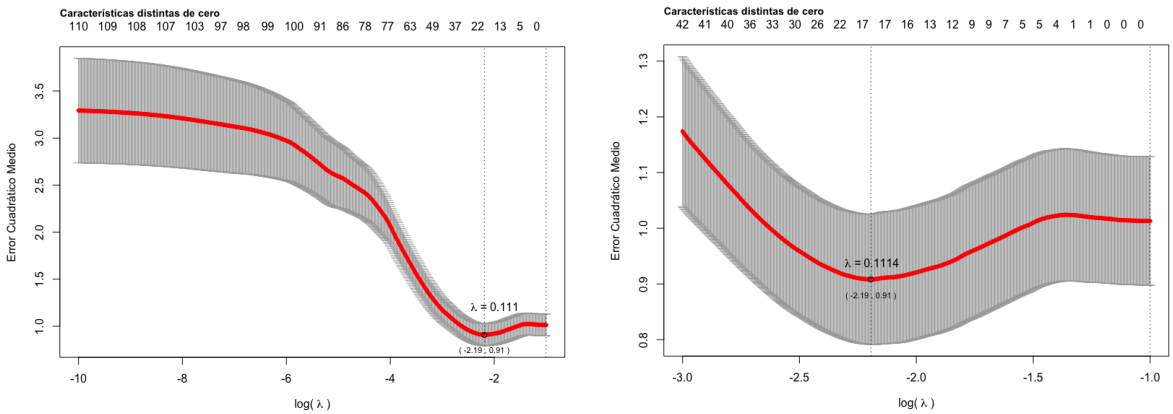
Tabla 2. Coeficientes regresión lineal

Partimos de unos datos en alta dimensión $p > n$ con $110 > 100$, y asumimos que se trata de un sistema disperso en el que es posible obtener un buen modelo considerando unas pocas variables distintas de cero. Comprobaremos si la aplicación del método de regularización LASSO nos permitirá seleccionar un subconjunto de variables características distintas de cero.

Una vez decidido el tipo de regularización a aplicar (LASSO), hemos de decidir el valor λ de penalización más conveniente mediante validación cruzada, a través de la función cv.glmnet().

```
> Penaliz_cv <- cv.glmnet(x, y, alpha = 1, lambda = rango_de_lambdas, standardize = TRUE,
                          nfolds = 10)
```

Obtenemos el valor $\lambda = 0.1114$ con un error cuadrático medio de 0.9083, como podemos ver en la Gráfica 29.



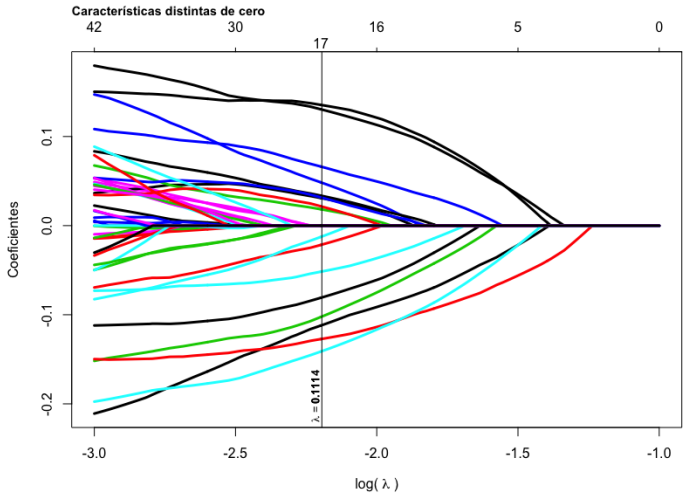
Gráfica 29. LASSO. Error Cuadrático Medio en función de $\boldsymbol{log(\lambda)}$. Estimación por validación cruzada. (izqu) rango $\boldsymbol{-10.0 < log(\lambda) < -1.0}$, (drch) rango $\boldsymbol{-3.0 < log(\lambda) < -1.0}$

Como dato relevante observamos que el número de variables distintas de cero para ese valor de λ (penalización) es 17, lo que supone una muy importante reducción de la complejidad del modelo que parte de 110 variables, es decir, solo un 15,5% de variables serán necesarias para modelar el sistema.

Una vez obtenido λ mediante la función glmnet obtenemos la curva de los coeficientes.

```
> glmnet(x, y, alpha = 1, lambda = rango_de_lambdas, standardize = TRUE)
```

y señalamos la λ que nos permite el mínimo error cuadrático medio, como podemos ver en la Gráfica 30.



Gráfica 30. LASSO. Coeficientes en función de $\boldsymbol{log(\lambda)}$.
Para $\boldsymbol{\lambda = 0.1114}$ obtenemos 17 coeficientes distintos de cero.

Los coeficientes obtenidos son los siguientes:

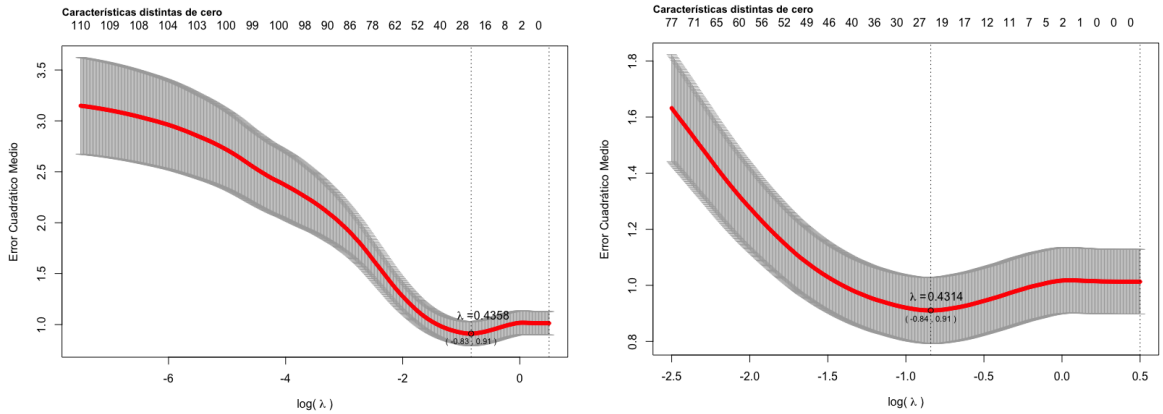
```
> Modelo_LASSO$beta
110 x 1 sparse Matrix of class "dgCMatrix"
```

V1		V29		V57		V85	
V2		V30		V58		V86	
V3		V31	0.04811198	V59	0.03139483	V87	
V4		V32		V60		V88	
V5		V33		V61		V89	
V6	-0.02132933	V34		V62		V90	
V7		V35		V63		V91	
V8		V36		V64	0.13526379	V92	
V9	0.06595912	V37		V65		V93	
V10		V38		V66		V94	
V11		V39	-0.11174191	V67		V95	
V12		V40		V68		V96	0.13025006
V13	-0.01226883	V41		V69		V97	
V14		V42		V70		V98	
V15		V43	-0.10206263	V71		V99	
V16		V44		V72		V100	
V17		V45		V73		V101	
V18		V46		V74		V102	
V19		V47		V75		V103	
V20		V48	-0.05151437	V76		V104	
V21		V49		V77		V105	
V22		V50		V78		V106	
V23		V51		V79		V107	
V24	0.03072409	V52	0.03337612	V80	-0.08063659	V108	-0.14091240
V25		V53		V81	-0.12713131	V109	
V26		V54	0.02141649	V82		V110	
V27		V55		V83			
V28	0.01817000	V56		V84			

Tabla 3. Coeficientes LASSO

Como hemos visto en el capítulo 1, en caso de sospechar colinealidad (donde LASSO presenta ineficiencias) o con objeto de intentar mejorar el error cuadrático medio podemos aplicar *Elastic Net*, sin embargo *Elastic Net* en nuestro ejemplo no mejora el error cuadrático medio ya que lo mantiene en 0,9106, ya que para un $\alpha = 0.25$ ofrece una curva muy similar a la anterior como podemos observar en la Gráfica 31.

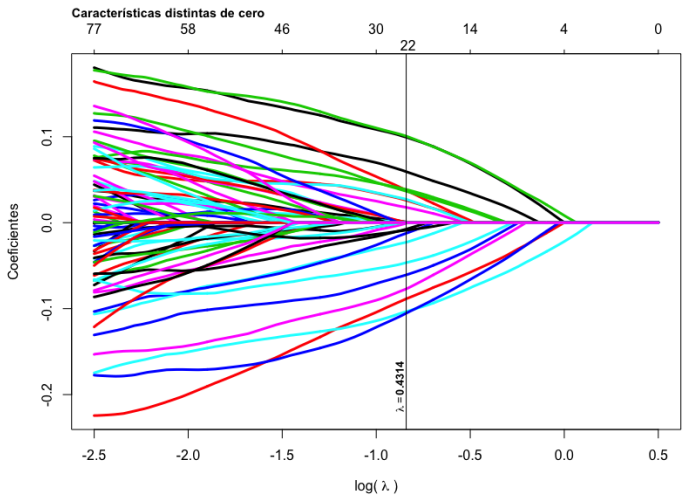
```
> Penaliz_cv <- cv.glmnet(x, y, alpha = 0.25, lambda = rango_de_lambdas, standardize = TRUE,
                          nfolds = 10)
```



Gráfica 31. *Elastic Net*. Error Cuadrático Medio en función de $\boldsymbol{log(\lambda)}$. Estimación por validación cruzada. (izqu) rango – $\mathbf{7.5 < log(\lambda) < 0.5}$, (drch) rango – $\mathbf{2.5 < log(\lambda) < 0.5}$

Con el valor de $\lambda = 0.4314$ obtenemos la Gráfica 32 de los coeficientes, donde observamos que el número de coeficientes distintos de cero son 22, lo que supone la incorporación al modelo de 5 variables adicionales al anterior modelo LASSO.

```
> glmnet(x, y, alpha = 0.25, lambda = rango_de_lambdas, standardize = TRUE)
```



Gráfica 32. *Elastic Net*. Coeficientes en función de $\boldsymbol{log(\lambda)}$.
Para $\boldsymbol{\lambda = 0.4314}$ obtenemos 22 coeficientes distintos de cero.

En la Tabla 4 podemos ver los coeficientes obtenidos (sombreados los nuevos coeficientes de variables), pero los valores de los nuevos coeficientes son prácticamente cero, por lo que en este caso la aportación de *Elastic Net* no ha sido significativa en la mejora del modelo.

```
> Modelo_Elastic_Net$beta

110 x 1 sparse Matrix of class "dgCMatrix"
```

V1	0.0007985174	V29		V57		V85	
V2		V30		V58		V86	
V3		V31	0.0376661456	V59	0.0386518518	V87	
V4		V32		V60		V88	
V5		V33		V61		V89	
V6	-0.0227383459	V34		V62		V90	
V7		V35		V63		V91	
V8	-0.0006773022	V36		V64	0.0994101473	V92	
V9	0.0593538997	V37		V65		V93	
V10		V38		V66		V94	
V11		V39	-0.0885778148	V67		V95	
V12		V40		V68		V96	0.1006787341
V13	-0.0123651313	V41		V69		V97	
V14		V42		V70		V98	
V15		V43	-0.0770115921	V71		V99	
V16		V44		V72		V100	
V17		V45		V73		V101	
V18		V46		V74		V102	
V19		V47		V75	-0.0087437339	V103	
V20		V48	-0.0464816825	V76		V104	-0.0075118840
V21		V49		V77		V105	
V22		V50		V78		V106	
V23		V51	0.0002674844	V79		V107	
V24	0.0260274063	V52	0.0366306425	V80	-0.0607247757	V108	-0.1053632368
V25		V53		V81	-0.1033993757	V109	
V26		V54	0.0271710879	V82		V110	
V27		V55		V83			
V28	0.0178226804	V56		V84			

Tabla 4. Coeficientes *Elastic Net.*

Con el ejemplo descrito hemos obtenido no solo una significativa reducción de la dimensionalidad, sino que obtenemos un modelo que tanto en LASSO como *Elastic Net* presenta un conjunto muy reducido de variables lo que confirma que se trata de un sistema disperso en alta dimensión y que puede ser descrito con una considerable reducción de la complejidad inicial (110 variables).

Variando los valores de α en *Elastic Net*, como podemos observar en la Tabla 5 que no obtenemos prácticamente reducción del error cuadrático medio, y el aumento (pequeño)[19] del número de variables que se produce a medida que nos acercamos a *Ridge* $(\alpha = 0)$.

Por consiguiente, podemos afirmar que el modelo obtenido, en este caso con LASSO, define de forma adecuada el sistema.

α	λ	Error Cuadrático Medio	Número de variables
1.00	0.1114	0.9083	17
0.95	0.1177	0.9080	17
0.75	0.1496	0.9067	17
0.55	0.2023	0.9059	17
0.35	0.3137	0.9067	19
0.25	0.4314	0.9106	22
0.15	0.6813	0.9205	27
0.05	1.4532	0.9424	43
0.00	1.6487	1.0453	110

Tabla 5. *Elastic Net.* De LASSO $(\boldsymbol{\alpha = 1})$ a *Ridge* $(\boldsymbol{\alpha = 0})$.

[19] Salvo en la proximidades de $\alpha \to 0$

REGULARIZACIÓN AGRUPAMIENTO JERÁRQUICO DISPERSO

Para ver mediante un ejemplo la regularización en el funcionamiento del agrupamiento jerárquico disperso generamos un conjunto de datos aleatorios con la misma matriz que el ejemplo anterior.

Dado que nos encontramos en métodos de agrupamiento (aprendizaje no supervisado) prescindimos del vector de variables dependientes.

Para inducir la diferenciación de agrupamiento en dos grupos (*clusters*), puesto que toda matriz x es homogénea en cuanto a todas sus variables con distribución normal, los 50 primeros elementos serán desplazados mediante la suma de 2, pero solo a las 25 primeras variables; de esta manera forzamos la diferenciación en dos grupos únicamente en las 25 primeras variables características de la matriz.

Las variables $p = 26, \cdots, 110$ serán homogéneas[20] en los 100 elementos muestrales (filas)

$$\begin{pmatrix} (x_{1,1}+2) & \cdots & (x_{1,25}+2) & x_{1,26} & \cdots & x_{1,110} \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ (x_{50,1}+2) & \cdots & (x_{50,25}+2) & x_{50,26} & \cdots & x_{50,110} \\ x_{51,1} & \cdots & x_{51,25} & x_{51,26} & \cdots & x_{51,110} \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ x_{100,1} & \cdots & x_{100,25} & x_{100,26} & \cdots & x_{100,110} \end{pmatrix}$$

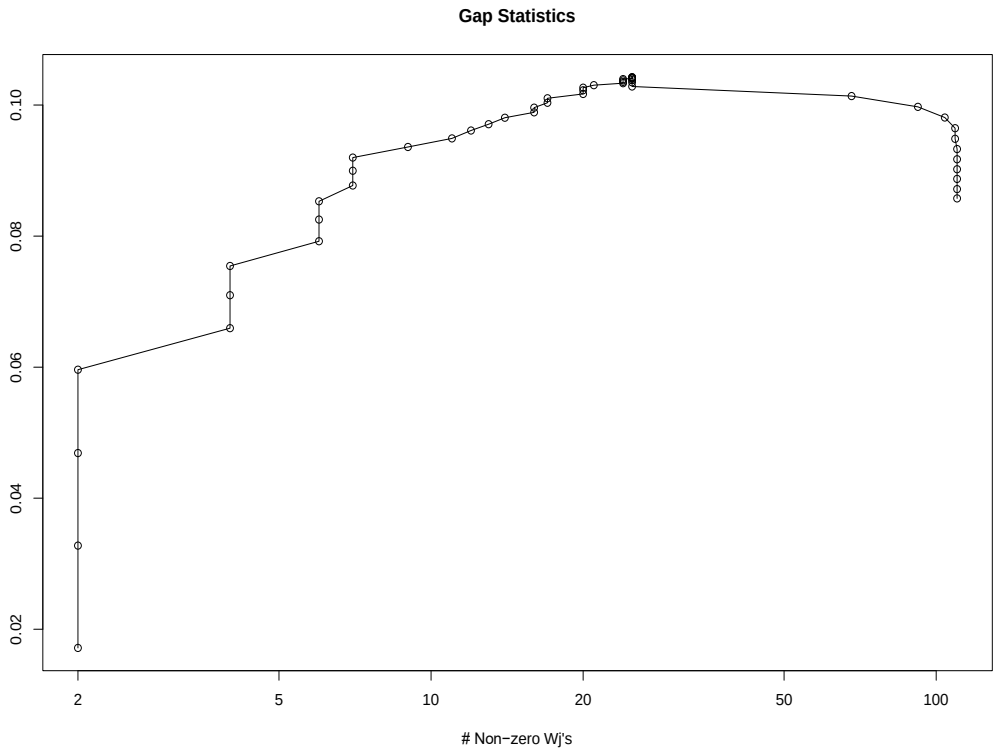
```
> set.seed(122)
> x <- matrix(rnorm(100*110), ncol = 110)
> x <- scale(x, TRUE, TRUE)

> y <- c(rep(1,50), rep(2,50))
> x[y==1,1:25] <- x[y==1,1:25]+2
```

[20] Podría considerarse en una situación real que estas variables corresponden a ruido.

Al tratarse de un método de regularización, como se ha descrito en el capítulo 2, en primer lugar, hemos de determinar los pesos significativos de cada variable, la regularización hará que la mayoría de los pesos asignados a las variable sean cero quedando únicamente un reducido grupo de variables con pesos distintos de cero (sistema disperso de alta dimensión).

El criterio que utilizaremos para determinar el vector de pesos más adecuado es el GAP estadístico, comentado en el capítulo 2.

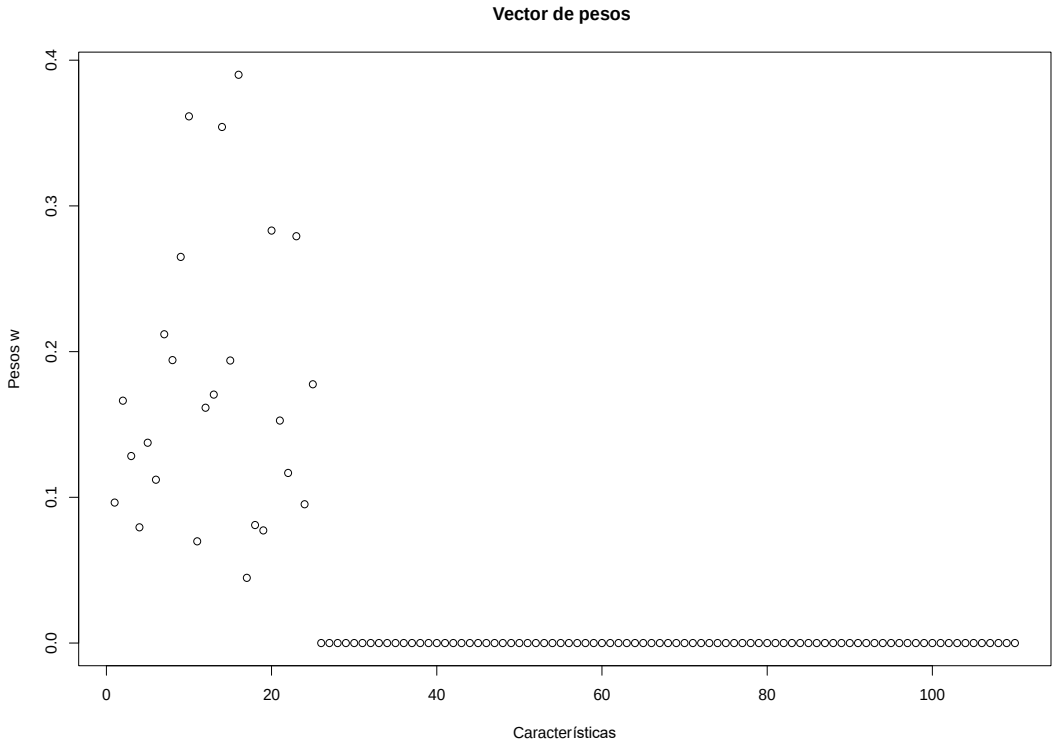
```
> Jerarq.out <- HierarchicalSparseCluster.permute(x, wbounds = seq(1.1, 6, by = 0.1 ),
                                                  nperms = 5)
```



Gráfica 33. GAP Estadístico en función del número de pesos No-Cero. Agrupamiento Jerárquico Disperso.

Vemos en la Gráfica 33 que el valor máximo del GAP se sitúa en algo más de 20 pesos distintos de cero, comprobando a que valor límite (*Wbound*) corresponde, `> Jerarq.out$bestw` nos devuelve un valor de 4.4, y comprobando en la tabla `> print(Jerarq.out)` que 4.4 corresponde a un valor de 25 pesos distintos de cero, el resultado es coincidente con los datos elaborados inicialmente.

De forma más visual podemos representar la gráfica del vector de pesos (Gráfica 34) donde en nuestro ejemplo se puede observar claramente la distribución de las variables a las que han sido asignadas pesos distintos de cero, siendo la mayoría (modelo disperso) pesos con valor de cero; a partir de la característica 25.



Gráfica 34. Vector de pesos W_j en función de las características. Agrupamiento Jerárquico Disperso.

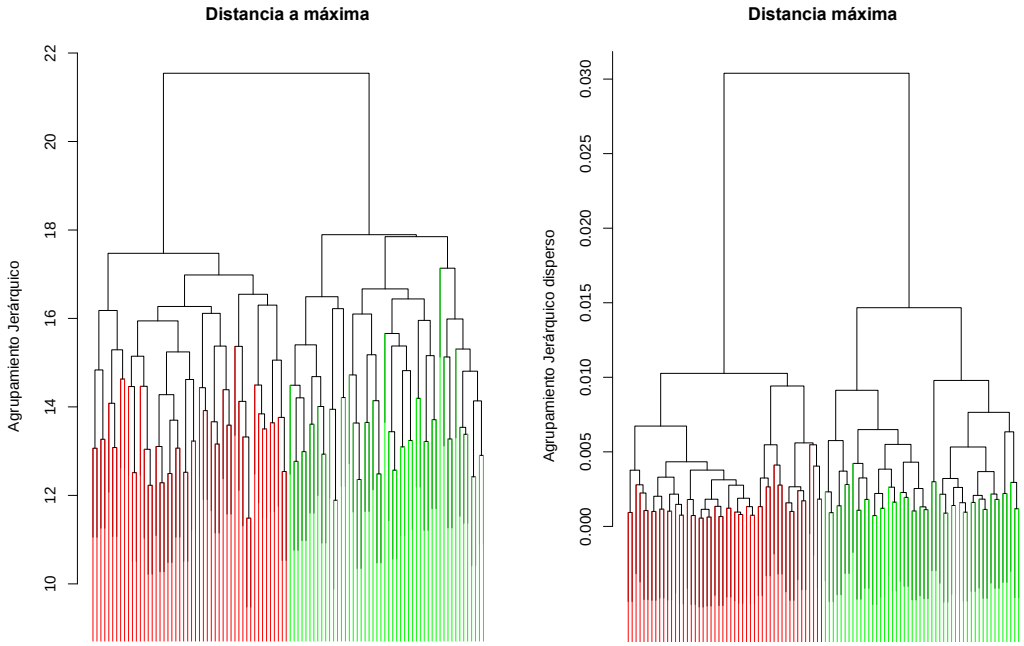
Una vez que se ha determinado el peso de las características, tenemos el conjunto de variables sobre las que vamos a aplicar el método que corresponda/decidamos de agrupamiento jerárquico, pero en este caso hemos reducido significativamente las variables a considerar.

Para verificar el funcionamiento lo adecuado es proceder a la aplicación de un agrupamiento jerárquico con todas las variables (estándar), lo que implica mayor complejidad y mayor coste computacional. Y confrontarlo con el agrupamiento jerárquico disperso, si el resultado es correcto habremos obtenido una significativa simplificación del modelo.

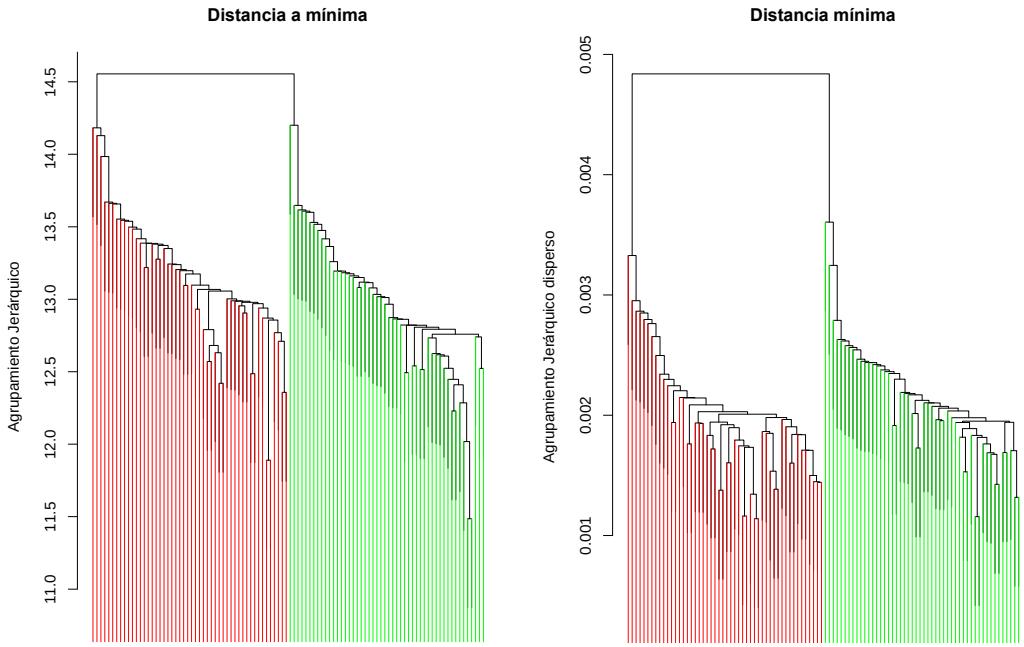
En las siguientes gráficas veremos a la izquierda el agrupamiento jerárquico estándar con hclust () y a la derecha el agrupamiento jerárquico disperso con HierarchicalSparseCluster (), señalaremos con el color correspondiente[21] (rojo, verde) a cada elemento en función de su

[21] Rojo los 50 primeros elementos y Verde los 50 últimos

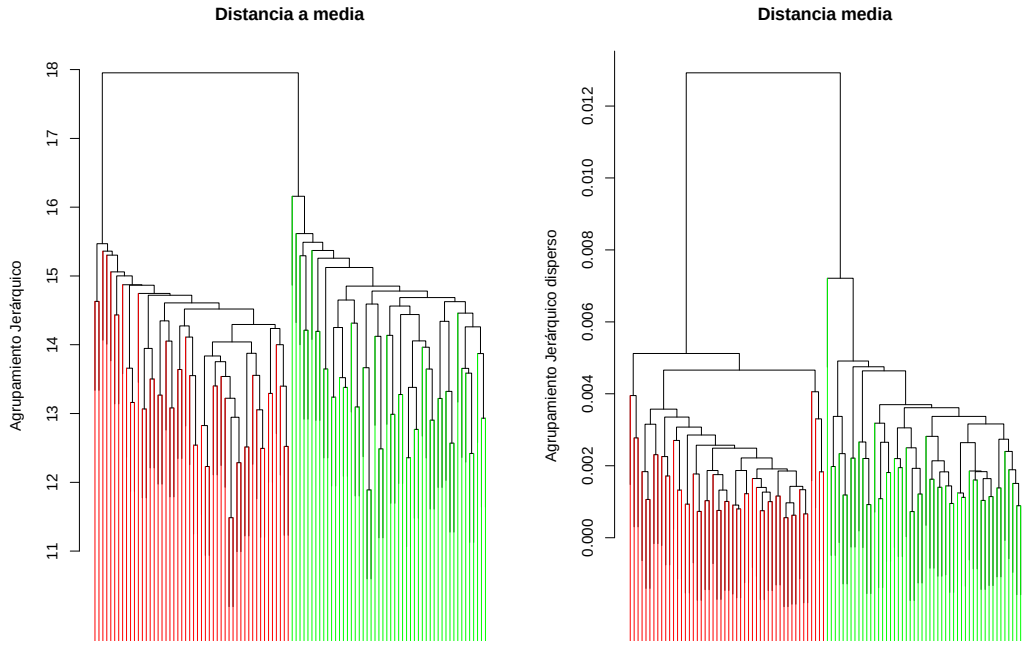
pertenencia real a cada grupo, mediante la función ColorDendrogram(). Realizaremos la comparación con los cuatro principales métodos, según el criterio de distancia para determinar la pertenencia a un grupo (Distancia Máxima, Distancia Mínima. Distancia Media, Distancia a Centroide).



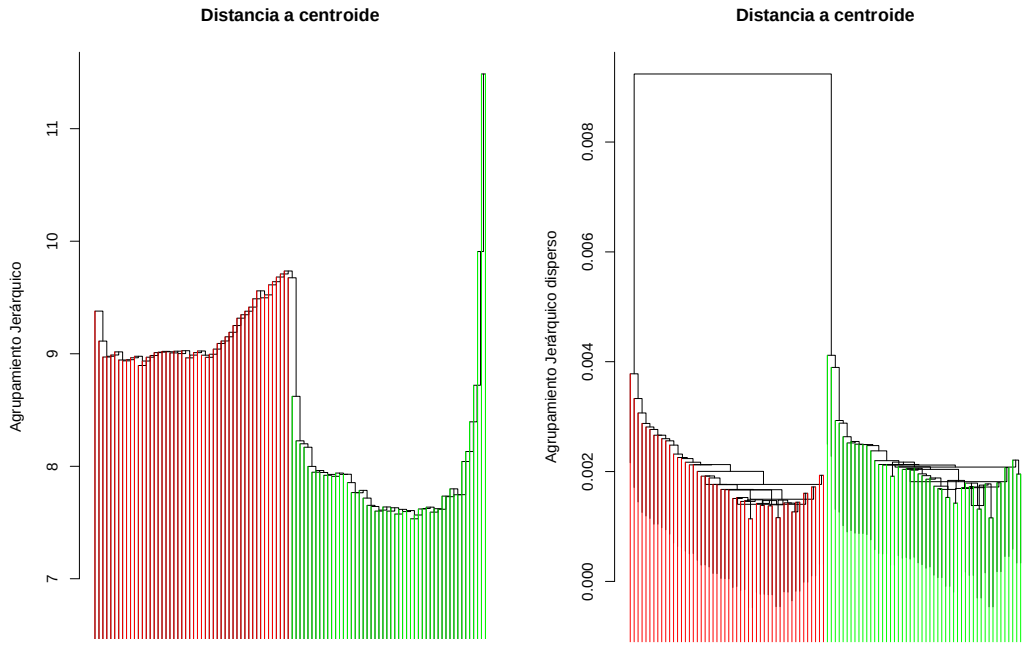
Gráfica 35. Agrupamiento jerárquico disperso versus agrupamiento jerárquico clásico. Distancia Máxima (*complete*).



Gráfica 36. Agrupamiento jerárquico disperso versus agrupamiento jerárquico clásico. Distancia Mínima (*single*).



Gráfica 37. Agrupamiento jerárquico disperso versus agrupamiento jerárquico clásico. Distancia Media (*average*).



Gráfica 38. Agrupamiento jerárquico disperso versus agrupamiento jerárquico clásico. Distancia a Centroide (*centroid*).

Como podemos comprobar, la eficiencia del agrupamiento jerárquico disperso en este ejemplo es clara, reduciendo un 77,27% las variables necesarias para asignar el agrupamiento de forma correcta en los cuatro métodos analizados.

REGULARIZACIÓN AGRUPAMIENTO *K-MEANS* DISPERSO

Implementamos el mismo conjunto del ejemplo anterior. Generamos un conjunto de datos aleatorios, formados por una matriz 100×110, que simularía una matriz de 100 muestras cada una compuesta de 110 características, y modificamos las 25 primeras características de forma que puedan obtenerse 2 grupos diferenciados.

Las variables $p = 26, \cdots, 110$ serán homogéneas[22] en los 100 elementos muestrales.

$$\begin{pmatrix} (x_{1,1}+2) & \cdots & (x_{1,25}+2) & x_{1,26} & \cdots & x_{1,110} \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ (x_{50,1}+2) & \cdots & (x_{50,25}+2) & x_{50,26} & \cdots & x_{50,110} \\ x_{51,1} & \cdots & x_{51,25} & x_{51,26} & \cdots & x_{51,110} \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ x_{100,1} & \cdots & x_{100,25} & x_{100,26} & \cdots & x_{100,110} \end{pmatrix}$$

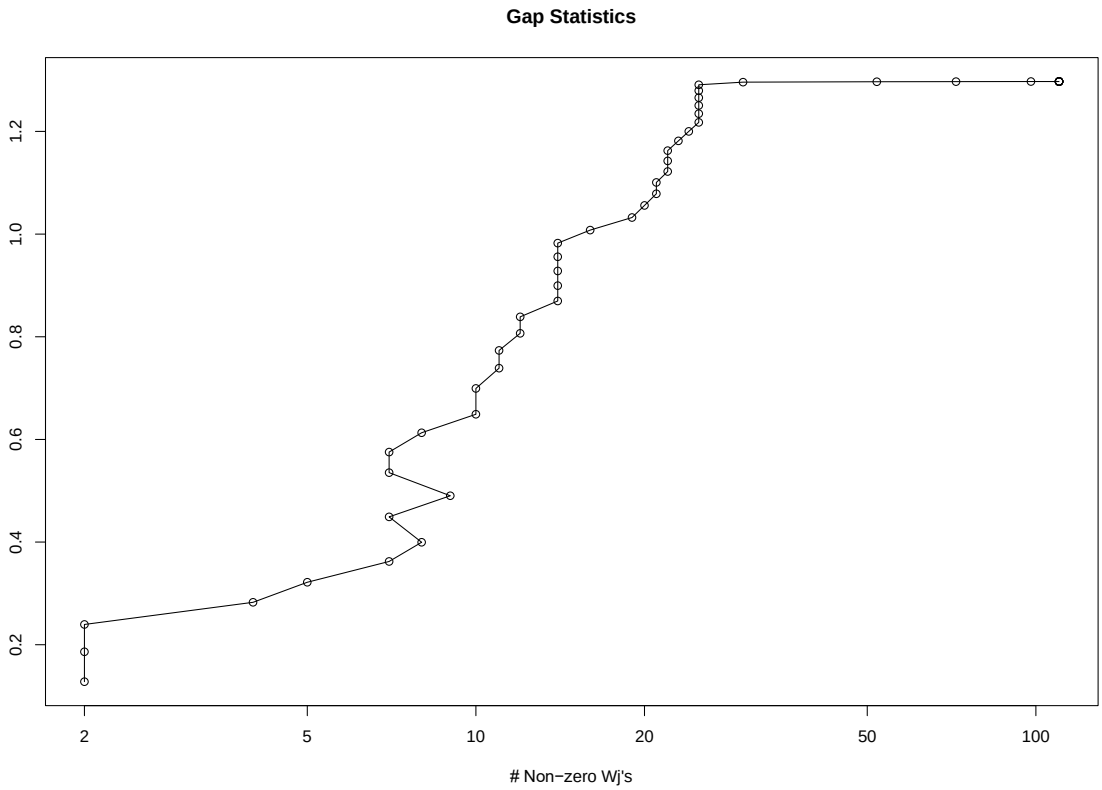
De igual forma que en anterior procedimiento debemos obtener el valor óptimo de penalización para los pesos (*wbound*) mediante la función KmeansSparseCluster.permute() que utilizaremos para determinar el vector de pesos más adecuado mediante GAP estadístico, comentado en el capítulo 2.

```
> KMean.perm <- KMeansSparseCluster.permute(x, K = 2, wbounds = seq(1.001,6, by=0.1),
                                            nperms = 5)
```

Vemos en la Gráfica 39 que el valor máximo del GAP se estabiliza a partir de los 20 pesos distintos de cero.

Comprobamos a que el mejor límite (*Wbound*) `> KMean.perm$bestw` nos devuelve un valor de 5.401, y comprobamos en la tabla `> print(KMean.perm)` que 5.401 corresponde a un valor de 110 pesos distintos de cero, es decir, todas las variables. Por lo que no conseguimos selección de variables mediante la asignación de pesos cero.

[22] Podría considerarse en una situación real que estas variables corresponden a ruido.



Gráfica 39. GAP Estadístico en función del número de pesos No-Cero. Agrupamiento K-*Mean* Disperso.

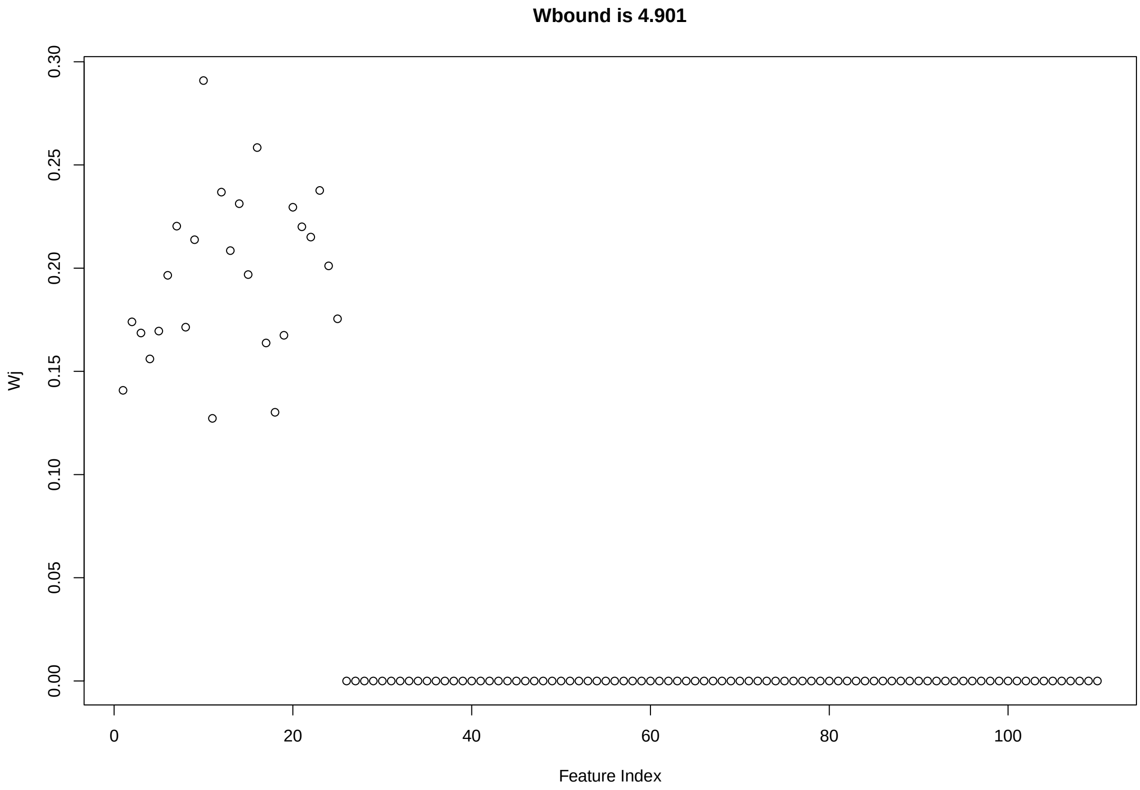
Sin embargo, de la observación de la gráfica, vemos que es monótona creciente, pero en su tramo final se encuentra estabilizada. En la tabla generada[23] comprobamos que los 25 pesos distintos de cero se mantienen desde un valor límite de 4.401 a 4.901 y es a partir de 5.001 cuando se produce la rápida incorporación de todas las variables con pesos distintos de cero.

Tomaremos pues el valor de 4.901 como valor razonable para un límite *wbounds* que nos permita la dispersión del modelo.

Para comprobar la distribución de pesos distintos de cero representamos la gráfica del vector de pesos (Gráfica 40) donde en nuestro ejemplo se puede observar claramente la distribución de las variables a las que han sido asignadas pesos distintos de cero, siendo la mayoría (modelo disperso) pesos con valor de cero a partir de la característica 25, como era de prever.

```
> KMean.out <- KMeansSparseCluster(x, K = 2, wbounds = 4.901)
```

[23] `> print(KMean.perm)`



Gráfica 40. Vector de pesos W_j en función de las características. Agrupamiento *K-Mean* Disperso.

Vemos claramente que los pesos corresponden a las 25 primeras características.

Comprobamos el resultado de asignación a cada *cluster* de cada elemento muestral mediante > print(KMean.out)

Wbound is 4.901 :

Number of non-zero weights: 25

Sum of weights: 4.900994

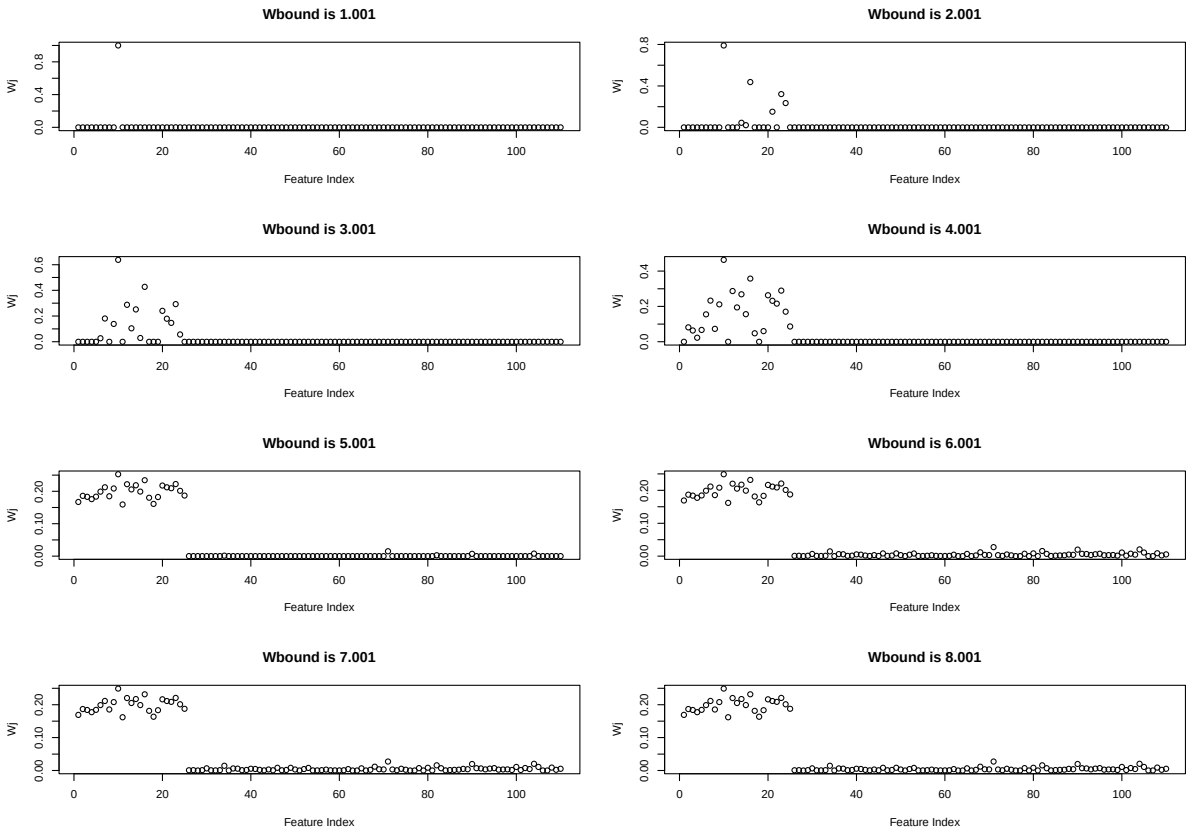
Clustering: 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2

Al igual que en el caso de agrupamiento jerárquico debemos comparar con respecto al agrupamiento *k-mean* estándar mediante la función > KMean_nosparse <- kmeans(x, 2) con el que obtenemos el mismo resultado.

```
Clustering vector:
 [1] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
1 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
2 2
```

Por lo tanto, ambos métodos asignan correctamente los elementos al *cluster* correspondiente (50 primeros al *cluster* 1 y 50 últimos al *cluster* 2), con la importante diferencia que el agrupamiento *k-mean* disperso lo ha hecho considerando solo 25 variables en lugar de 110.

La función `KMeansSparseCluster(x, K = 2, wbounds = seq(1.001, 9, by = 1))` nos permite comprobar un rango de límites (*wbound*) y sus resultados respecto de la distribución de pesos, donde se observa que el método diferencia claramente los elementos diferentes aunque es el ajuste del límite (la penalización de los pesos) el que determinará la asignación más correcta.



Gráfica 41. Distribución de los pesos para diferentes límites *wbound* (penalizaciones).

ANEXO I

Con objeto de plantear de forma simple el algoritmo a utilizar introducimos una notación adicional que resultará útil. Sea $D \in \mathbb{R}^{n^2 \times p}$ la matriz en la que la columna *j* consta de los elementos $\left\{d_{i,i',j}\right\}_{i,i'}$. Entonces, $\sum_j w_j \sum_{i,i'} d_{i,i',j} U_{i,i'} = \boldsymbol{u}^T \boldsymbol{D}\boldsymbol{w}$ donde $\boldsymbol{u} \in \mathbb{R}^{\boldsymbol{n^2}}$ se obtiene encadenando $\boldsymbol{U}$ en un vector. Por lo tanto, el criterio (15) es equivalente a

$$\max_{\boldsymbol{w},\boldsymbol{u}} \{\boldsymbol{u}^T \boldsymbol{D}\boldsymbol{w}\}$$

$$sujeto\ a\ \|\boldsymbol{u}\|^2 \leq 1, \qquad \|\boldsymbol{w}\|^2 \leq 1, \qquad \|\boldsymbol{w}\|_1 \leq s, \qquad w_j \geq 0 \qquad \forall j$$

Por lo tanto, algoritmo para el agrupamiento jerárquicos dispersos

Algoritmo para agrupamiento jerárquico disperso.

1. Inicialice $\boldsymbol{w}$ como $w_1 = w_2 = \cdots = w_p = \sqrt{p}$.
2. Iterar hasta la convergencia:
   a. Actualizar $\boldsymbol{u} = \frac{\boldsymbol{D}\boldsymbol{w}}{\|\boldsymbol{D}\boldsymbol{w}\|_2}$
   b. Actualizar $\boldsymbol{w} = \frac{S(\boldsymbol{a}_+,\Delta)}{\|S(\boldsymbol{a}_+,\Delta)\|_2}$ donde $\boldsymbol{a} = \boldsymbol{D}^T\boldsymbol{u}$ y $\Delta = 0$ si este resulta en $\|\boldsymbol{w}\|_1 \leq s$, en caso contrario, $\Delta > 0$ se elige tal que $\|\boldsymbol{w}\|_1 = s$

   Criterio de parada $\frac{\sum_{j=1}^{p}\left|w_j^r - w_j^{r-1}\right|}{\sum_{j=1}^{p}\left|w_j^{r-1}\right|} < 10^{-4}$, donde $\boldsymbol{w}^r$ indica el conjunto de pesos obtenidos en la iteración r

3. Reescribir $\boldsymbol{u}$ como una matriz $n \times n$, $\boldsymbol{U}$
4. Desarrollar el agrupamiento jerárquico sobre la matriz de disimilitud $n \times n$, $\boldsymbol{U}$

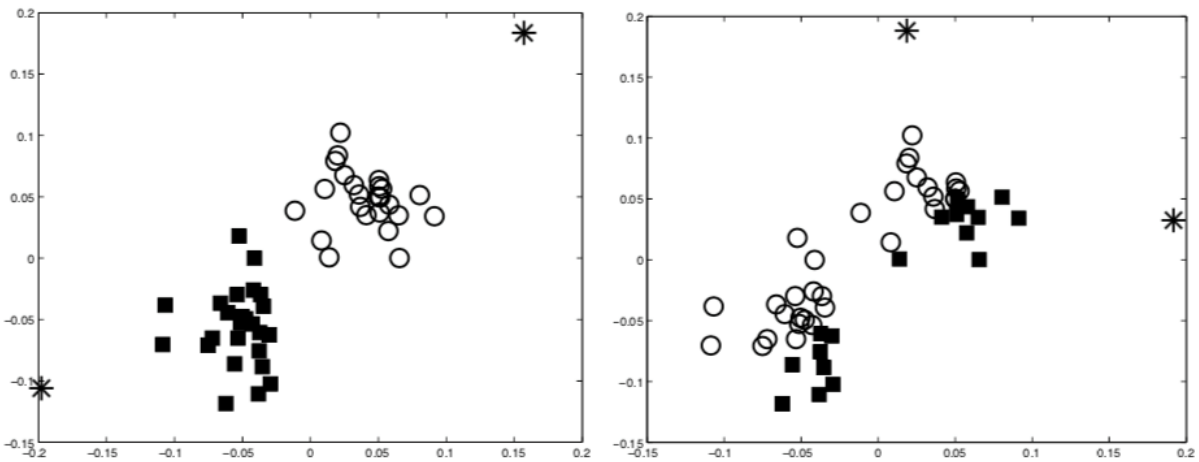
ANEXO II

Consideremos un simple problema de agrupamiento en 2D.

Sea x_j, $j = 1, 2, \cdots, 25$ formando el primer *cluster*. Estos x's se muestran como cuadrados rellenos en la gráfica de la izquierda de la Gráfica 42.
$\{x_j\}_{j=1}^{25}$ se generaron muestreando uniformemente de un círculo con radio 0,07 centrado en $[-0{,}05\ , -0.05]$.

Sea además x_j, $j = 26, 27, \cdots, 50$ forman el segundo *cluster*. Estos puntos se muestran como círculos en la gráfica de la izquierda de la Gráfica 42. $\{x_j\}_{j=26}^{50}$ se generaron muestreando uniformemente de un círculo con radio 0,07 centrado en $[0{,}05\ , 0.05]$.

Apliquemos el método de agrupamiento $k - means$ a $\{x_j\}_{j=1}^{50}$. Tomamos $k = 2$, usamos el algoritmo (Lloyd, 1982 [02]), ver Anexo III, y un máximo de 1000 iteraciones. Los dos asteriscos $(*)$ que se muestran en la gráfica de la izquierda de la Gráfica 42 muestran la inicialización de las dos medias del algoritmo de agrupamiento de $k - means$ (Lloyd, 1982) [03]. Para esta inicialización en particular, $k - means$ identifica los *clusters* "verdaderos". Sin embargo, si el algoritmo se inicializa de manera diferente, se obtiene un resultado erróneo, como se muestra en la gráfica de la derecha de la Gráfica 42.



Gráfica 42. Sensibilidad del algoritmo agrupamiento *k-mean* a la condición inicial. Asterisco (*) puntos iniciales. Tomado de Lindsten et al. (2011), página 3 [04].

La solución mínima siempre puede ser calculada por una búsqueda exhaustiva. Sin embargo, dado que el número de combinaciones crece como $\sum_{i=1}^{N/2} \frac{N!}{(N-i)!}$ (para $k = 2$), aproximadamente $2 \cdot 10^{39}$ posibles soluciones diferentes deberían verificarse en este ejemplo particular para calcular la minimización de S. Esto es claramente impracticable.

ANEXO III

Algoritmo 1.

$k-means$ *clustering* (algoritmo de Lloyd, (1982) [04])

Datos de inicio

- Conjunto de puntos $\{x_j\}_{j=1}^{N}$
- Número de *cluster* $k \leq N$
- Centroides iniciales $\{\theta_i\}_{i=1}^{k}$

Inicio

- Conjuntos indexados $\{S_i\}_{i=1}^{k}$

1: **Bucle**

2: **Actualizar conjunto de índices**: Para los centroides fijados $\{\theta_i\}_{i=1}^{k}$, calcular los conjuntos indexados $\{S_i\}_{i=1}^{k}$

$$S_i \leftarrow \left\{j: \left\|x_j - \theta_i\right\| \leq \left\|x_j - \theta_l\right\|, l = 1, 2, \cdots, k\right\}$$

3: **Actualizar centroides**: Para los conjuntos indexados fijados $\{S_i\}_{i=1}^{k}$, estimar los centroides $\{\theta_i\}_{i=1}^{k}$

$$\theta_i = \frac{1}{card\ S_i} \sum_{j \in S_i} x_j$$

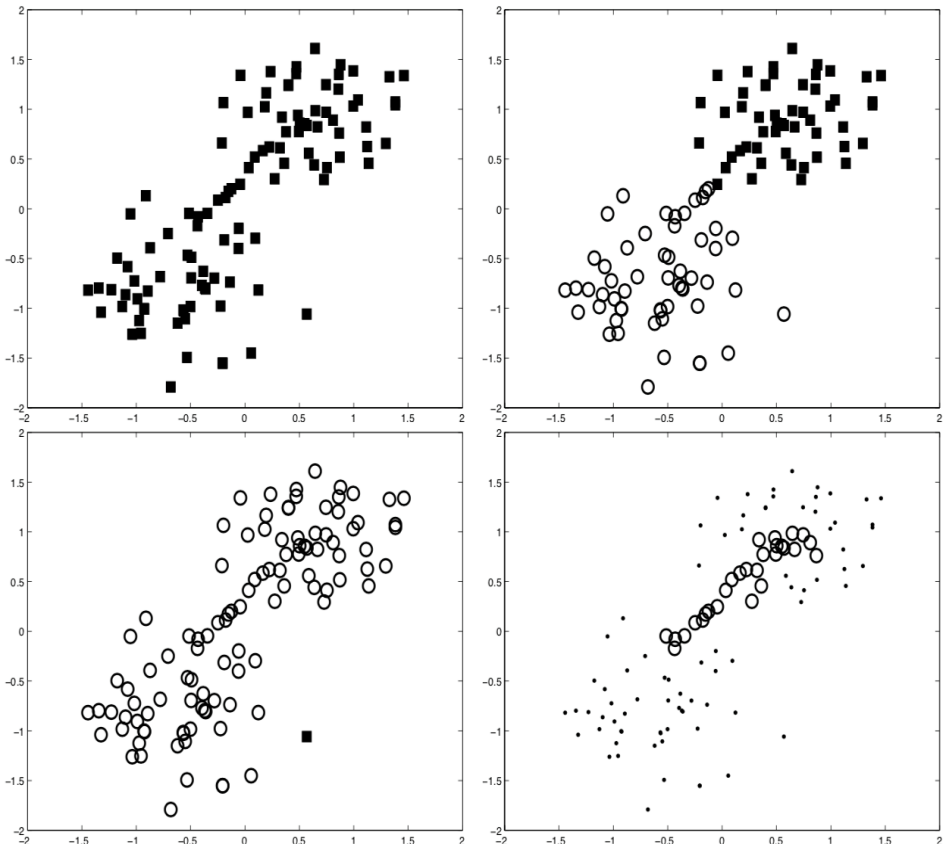
4: **Si** no hay cambios en la asignación de centroides desde la última iteración **o** el número máximo de iteraciones ha sido alcanzado **entonces**

5: **Volver**

6: **Fin de Si**

7: **Fin Bucle**

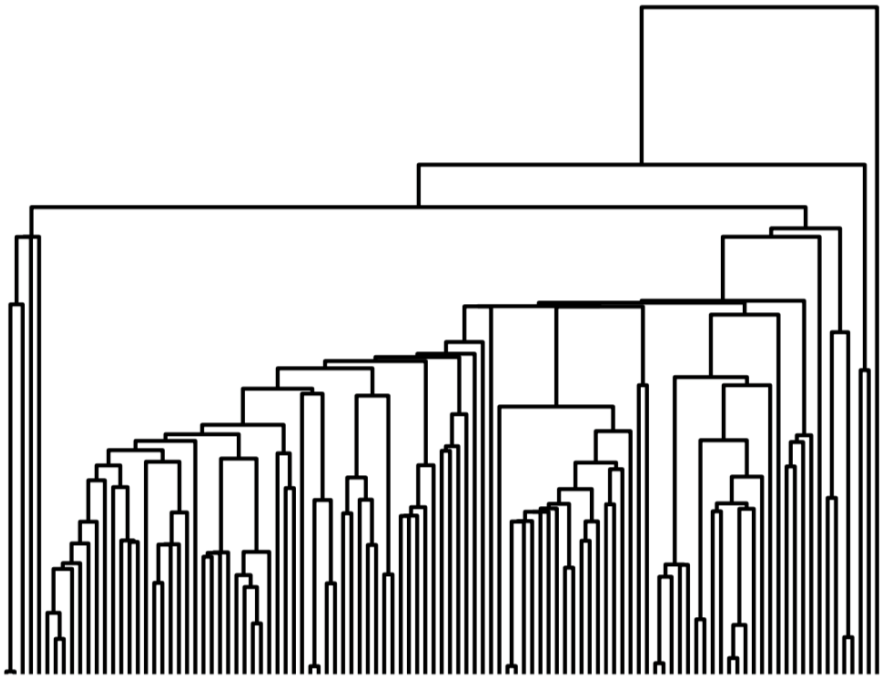
Consideramos $N = 107$ puntos, reflejados en la gráfica superior izquierda de la Gráfica 43.



Gráfica 43. Diferencias Agrupamientos jerárquicos.
- (Superior Izquierda) Conjunto de datos con dos *clusters* visibles.
- (Superior Derecha) *Cluster* convexo con $\boldsymbol{\lambda = 0.036}$.
- (Inferior Izquierda) Resultado del agrupamiento jerárquico para dos *clusters*.
- (Inferior Derecha) El mayor *cluster* –círculos- de un agrupamiento jerárquico con resultado para 32 *clusters.*

Tomado de Lindsten et al. (2011), página 7 [05].

Usamos un método aglomerativo de un solo enlace (*single linkage*). El diagrama de árbol, un dendrograma, que muestra la fusión de *clusters* se observa en la Gráfica 44.



Gráfica 44. Dendograma de un agrupamiento jerárquico.
Tomado de Lindsten et al. (2011), página 7 [06].

El algoritmo comienza con un *cluster* para cada punto de datos, ilustrado por las barras $N = 107$ en la parte inferior de la Gráfica 44. A medida que "nos movemos" hacia arriba en la figura, los *clusters* existentes se fusionan sucesivamente hasta que alcanzamos un mínimo de dos *clusters* en la parte superior.

Por lo tanto, el dendrograma proporciona un "conjunto de soluciones" completo. El usuario puede elegir "cortar" el árbol a cualquier profundidad deseada, produciendo una solución con el número correspondiente de *clusters*. Como se mencionó anteriormente, un problema con el agrupamiento jerárquico es que es codicioso. Si se comete un error al principio del algoritmo (es decir, en la parte inferior del dendrograma), este error no puede repararse a medida que avanzamos.

Este problema se ilustra en la Gráfica 43. La gráfica inferior izquierda muestra el resultado cuando el dendrograma se corta a un nivel que produce dos *clusters*. Uno de los *clusters* contiene un solo punto de datos, y los puntos restantes pertenecen al segundo *cluster*, lo que claramente no es óptimo con respecto al error dentro de la suma de cuadrados del *cluster*.

Para ver por qué obtenemos este resultado, optamos por detener la fusión de *clusters* en una etapa mucho más temprana, es decir, cuando tenemos hasta 32 *clusters*. El resultado de este nivel se muestra en la gráfica inferior derecha de la Gráfica 43. Para evitar el desorden de la figura, elegimos mostrar solo el *cluster* más grande. Los puntos de datos en este *cluster* se muestran como círculos, mientras que todos los puntos en los 31 *clusters* restantes se muestran como puntos. Dado que la fusión se basa solo en las propiedades locales, la proximidad del punto en el "centro" crea un *cluster* que se extiende en ambos *clusters* de puntos.

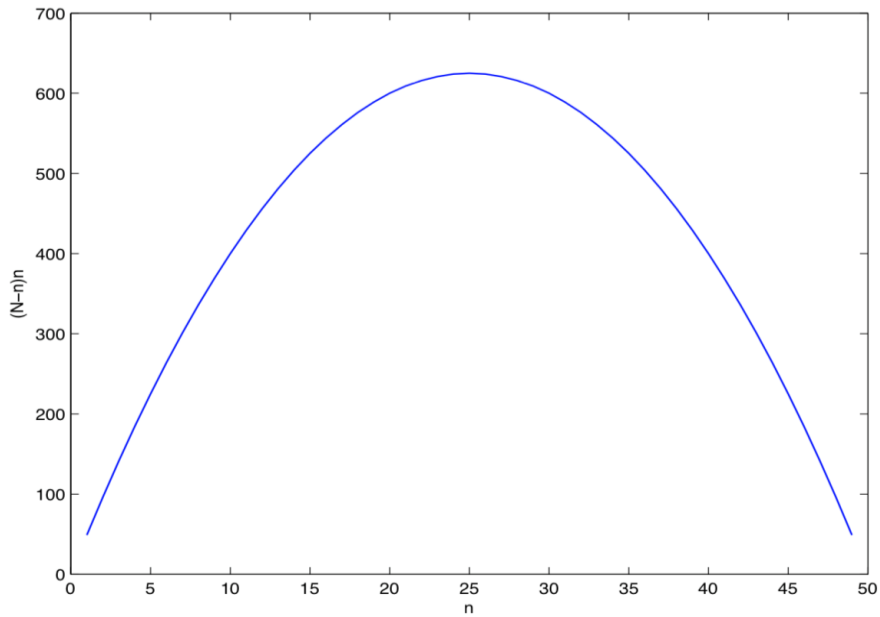
Los errores cometidos en esta etapa temprana no se pueden reparar mientras continuamos hacia dos *clusters*. Dado que no es un enfoque codicioso, el agrupamiento convexo funciona un poco diferente. Al resolver el problema de agrupamiento para todos los valores de λ, se puede crear una gráfica de la misma manera que en la Gráfica 43. Esta gráfica contiene información similar a la del dendrograma en la Gráfica 44. Para cualquier valor dado de λ, podemos extraer una solución contenida en cierto número de *clusters*. Sin embargo, no es posible dibujar un dendrograma sobre los resultados del agrupamiento convexo. La razón es que, al pasar de un nivel al siguiente, no se realiza la fusión de los *clusters* existentes. En su lugar, podemos obtener una configuración completamente nueva, lo que significa que un error cometido en una etapa temprana (pequeña λ) de hecho puede repararse a medida que aumentamos λ. En la gráfica superior derecha de la Gráfica 43 se muestra el resultado del agrupamiento convexo para $\lambda = 0.036$.

ANEXO V

Supongamos que λ se elige de tal manera que la solución al problema de agrupamiento convexo contenga $k = 2$ *clusters*. Para simplificar, supongamos que los puntos de datos se ordenan de modo que $\{x_j\}_{j=1}^{n}$ pertenezca al *cluster* 1 y $\{x_j\}_{j=n+1}^{N}$ pertenezca al *cluster* 2. Además, sea r la distancia entre los dos centroides de los *clusters* (bajo la norma elegida). Ahora, puesto que el término de regularización en $\lambda \sum_{j=2}^{N} \sum_{i<j} \|\mu_i - \mu_j\|_q$ controla el número de *clusters*, su valor debe permanecer constante mientras $k = 2$. Este término debería ser independiente de cuantos elementos de datos se asignen a cada *cluster*, es decir, independiente de n. Sin embargo, se puede verificar que el término de regularización en este caso viene dado por

$$\lambda \sum_{j=2}^{N} \sum_{i<j} \|\mu_i - \mu_j\|_q = \lambda r (N - n) n$$

el cual es dependiente de n, alcanzando su máximo en $n = [N/2]$ y su mínimo en $n = 1$ o $n = N - 1$, como podemos ver en la Gráfica 45.



Gráfica 45. Valor del término de regularización para diferentes tamaños de *clusters*. $(\mathbf{N} - \mathbf{n})\mathbf{n}$ vs. $\mathbf{n}$, para $\mathbf{N} = \mathbf{50}$.
Tomado de Lindsten et al. (2011), página 8 [07].

BIBLIOGRAFÍA

Akaike, H. (1974). “A new look at the statistical model identification.” IEEE Trans. Automatic Control, 19(6), 716–723. Cita [01].

Aloise, D., Deshpande, A., Hansen, P. y Popat, P. (2009), “NP-hardness of Euclidean sum-of-squares clustering,” Machine Learning, 75, 245–248. Cita [01].

Bakin, S. (1999). Adaptive regression and model selection in data mining problems, Technical report, PhD. thesis, Australian National University, Canberra. Cita [01].

Banerjee, A., Merugu, S., Dhillon, I. y Ghosh, J. (2005). Clustering with Bregman divergences. *Journal of machine learning research*, *6*(Oct), 1705-1749. Cita [01].

Beale, E., Kendall, M. y Mann, D. (1967). ‘The discarding of variables in multivariate analysis’, Biometrika 54(3/4), 357–366. Cita [01].

Bertsimas, D., King, A. y Mazumder, R. (2016). Best subset selection via a modern optimization lens. The annals of statistics, 44(2), 813-852. Cita [01], [02].

Breiman, L. (1995). Better subset regression using the nonnegative garrote. Technometrics, 37(4), 373-384. Cita [01], [02].

Breiman, L. (1996). Heuristics of instability and stabilization in model selection. The annals of statistics, 24(6), pp. 2350-2383. Cita [01].

Chang, W. (1983). On using principal components before separating a mixture of two multivariate normal distributions. Journal of the Royal Statistical Society: Series C (Applied Statistics), 32(3), 267-275. Cita [01].

Chen, S., Donoho, D. y Saunders, M. (1998), “Atomic Decomposition by Basis Pur- suit,” SIAM Journal on Scientific Computing, 20, 33–61. Cita [01].

Dasgupta, S. y Freund, Y. (2009), “Random projection trees for vector quantization,” IEEE Trans. Inf. Theor., 55 (7), 3229–3242. Cita [01].

Dempster, A., Laird, N. y Rubin, D. B. (1977). Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society: Series B (Methodological)*, *39*(1), 1-22. Cita [01].

Draper, N. y Smith H., (1966). Applied regression analysis. New York, London, Sydney. Cita [01].

Efron, B., Hastie, T., Johnstone, I. y Tibshirani, R. (2004). Least angle regression. The Annals of statistics, 32(2), 407-499. Cita [01], [02].

Efroymson, M. (1960). Multiple regression analysis. In Mathematical Methods for Digital Computers (eds. A. Ralston and H. S. Wilf), vol. 1, 191–203. Wiley. Cita [01].

Efroymson, M. (1966). Stepwise regression a backward and forward look. Florham Park, New Jersey. Cita [01].

Fan, J. y Li, R. (2001). Variable selection via nonconcave penalized likelihood and its oracle properties. Journal of the American statistical Association, 96(456), 1348-1360. Cita [01], [02], [03].

Fan, J. y Peng, H. (2004). Nonconcave penalized likelihood with a diverging number of parameters. The Annals of Statistics, 32(3), 928-961. Cita [01], [02].

Forgy, E. (1965), “Cluster analysis of multivariate data: efficiency versus interpretability of classifications,” Biometrics, 21, 768–780. Cita [01].

Fraley, C. y Raftery, A. E. (2002). Model-based clustering, discriminant analysis, and density estimation. *Journal of the American statistical Association*, *97*(458), 611-631. Cita [01].

Frank, L. y Friedman, J. (1993). A statistical view of some chemometrics regression tools. Technometrics, 35(2), 109-135. Cita [01].

Friedman, J., Hastie, T. y Tibshirani, R. (2001). The elements of statistical learning (Vol. 1, No. 10). New York: Springer series in statistics. Cita [01], [02], [03], [04], [05], [06], [07].

Friedman, J. y Meulman, J. (2004). Clustering objects on subsets of attributes (with discussion). *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, *66*(4), 815-849. Cita [01].

Friedman, J., Hastie, T. y Tibshirani, R. (2010). Regularization paths for generalized linear models via coordinate descent. Journal of statistical software, 33(1), 1. Cita [01], [02].

Fu, W. (1998). Penalized regressions: the bridge versus the lasso. Journal of computational and graphical statistics, 7(3), 397-416. Cita [01].

Ghosh, D. y Chinnaiyan, A. (2002). Mixture modelling of gene expression data from microarray experiments. *Bioinformatics*, *18*(2), 275-286. Cita [01].

Goldberger, A. (1991). A Course in Econometrics. Harvard, University Press, Cambridge. Cita [01],

Gower, J. y Ross, G. (1969), "Minimum Spanning Trees and Single Linkage Cluster Analysis," Journal of the Royal Statistical Society. Series C (Applied Statistics), 18, 54–64. Cita [01].

Hastie, T., Tibshirani, R. y Friedman, J. (2001). The Elements of Statistical Learning: Data Mining, Inference, and Prediction. Springer Series in Statistics. Springer New York Inc., New York, NY, USA. Cita [01].

Hastie, T., Tibshirani, R. y Wainwright, M. (2015). Statistical learning with sparsity: the lasso and generalizations. Chapman and Hall/CRC. Cita [01], [02], [03], [04], [05], [06], [07], [08].

Hocking, R. y Leslie, R. (1967). 'Selection of the best subset in regression analysis', Technometrics 9(4), 531–540. Cita [01].

Hocking, T., Joulin, A., Bach, F. y Vert, J. (2011, June). Clusterpath an algorithm for clustering using convex fusion penalties. In *28th international conference on machine learning* (p. 1). Cita [01].

Hoefling, H. (2010), "A Path Algorithm for the Fused Lasso Signal Approximator," Journal of Computational and Graphical Statistics, 19, 984–1006. Cita [01].

Hoerl, A., (1962). Application of ridge analysis to regression problems: Chemical Engineering Progress, 55 (1), pp. 69–78. Cita [01].

Hoerl, A. y Kennard, R. (1970a). Ridge Regression: Biased Estimates for Non-orthogonal Problems. Technometrics, 12(1), pp. 55–67. Cita [01], [02].

Hoerl, A. y Kennard, R. (1970b), Ridge Regression: Applications to Non-orthogonal Problems: Technometrics, 12(1), pp. 69–82. Cita [01].

Huang, J., Ma, S. y Zhang, C.-H. (2008), Adaptive Lasso for sparse highdimensional regression models, Statistica Sinica 18, 1603–1618. Cita [01], [02].

Izenman J. (2008). Modern Multivariate Statistical Techniques: Regression, Classification, and Manifold Learning, p. 141. Cita [01], [02], [03].

Jacob, L., Obozinski, G. y Vert, J. (2009). Group lasso with overlap and graph lasso. In Proceedings of the 26th annual international conference on machine learning. ACM. pp. 433-440. Cita [01].

Johnson, S. (1967), "Hierarchical clustering schemes," Psychometrika, 32, 241–254. Cita [01].

Kass, R. y Raftery, A. (1995). Bayes factors. *Journal of the american statistical association*, *90*(430), 773-795. Cita [01], [02].

Lance, G. y Williams, W. (1967), "A General Theory of Classificatory Sorting Strategies: 1. Hierarchical Systems," The Computer Journal, 9, 373–380. Cita [01].

Land, S. y Friedman, J. (1996) Variable fusion: a new method of adaptive signal regression. Technical Report. Department of Statistics, Stanford University, Stanford. Cita [01].

Lashkari, D. y Golland, P. (2008). Convex clustering with exemplar-based models. In *Advances in neural information processing systems* (pp. 825-832). Cita [01].

Lee, D. y Seung, H. (1999). Learning the parts of objects by non-negative matrix factorization. *Nature*, *401*(6755), 788. Cita [01].

Lin, Y. y Zhang, H. (2006). Component selection and smoothing in smoothing spline analysis of variance models. Annals of Statistics, 34(5), 2272-2297. Cita [01].

Lindsten, F., Ohlsson, H. y Ljung, L. (2011). *Just relax and come clustering!: A convexification of k-means clustering*. Linköping University Electronic Press. Cita [01], [02], [03], [04], [05], [06], [07].

Liu, J., Zhang, J., Palumbo, M. y Lawrence, C. (2003). Bayesian clustering with variable and transformation selections. *Bayesian statistics*, *7*, 249-275. Cita [01].

Ljung, L (1999). System identification, Theory for the user. System sciences series. Prentice Hall, Upper Saddle River, NJ, USA, second edition. Cita [01].

Lloyd, S. (1982), "Least squares quantization in PCM," Information Theory, IEEE Transactions on, 28, 129 – 137. Cita [01], [02], [03], [04].

MacQueen, J. (1967), "Some methods for classification and analysis of multivariate observations," in Proc. Fifth Berkeley Symp. on Math. Statist. and Prob., Univ. of Calif. Press, vol. 1, pp. 281–297. Cita [01].

Mallows, C. (1973). "Some Comments on C." Technometrics, 15, 661–675. Cita [01].

Maugis, C., Celeux, G. y Martin-Magniette, M. L. (2009). Variable selection for clustering with Gaussian mixture models. *Biometrics*, *65*(3), 701-709. Cita [01].

McLachlan, G., Bean, R. y Peel, D. (2002). A mixture model-based approach to the clustering of microarray expression data. *Bioinformatics*, *18*(3), 413-422. Cita [01], [02].

McLachlan, G., Peel, D. y Bean, R. (2003). Modelling high-dimensional data by mixtures of factor analyzers. *Computational Statistics & Data Analysis*, *41*(3-4), 379-388. Cita [01].

McLachlan, G. y Krishnan, T. (2007). *The EM algorithm and extensions* (Vol. 382). John Wiley & Sons. Cita [01].

Meinshausen, N. y Bühlmann, P. (2006). High-dimensional graphs and variable selection with the lasso. The annals of statistics, 34(3), 1436-1462. Cita [01], [02].

Meinshausen, N. (2007). Relaxed lasso, Computational Statistics and Data Analysis 52(1): 374–393. Cita [01], [02], [03], [04].

Murtagh, F. (1983), "A Survey of Recent Advances in Hierarchical Clustering Algorithms," The Computer Journal, 26, 354–359. Cita [01].

Nowozin, S. y Bakir, G. (2008). A decoupled approach to exemplar-based unsupervised learning. In *Proceedings of the 25th international conference on Machine learning* (pp. 704-711). ACM. Cita [01].

Pan, W. y Shen, X. (2007). Penalized model-based clustering with application to variable selection. *Journal of Machine Learning Research*, *8*(May), 1145-1164. Cita [01], [02].

Peña, J., Lozano, J. y Larranaga, P. (1999). An empirical comparison of four initialization methods for the k-means algorithm. *Pattern recognition letters*, *20*(10), 1027-1040. Cita [01].

Peña, D. (2002). Regresión y diseño de experimentos. Alianza Editorial. pp. 577 – 578. Cita [01].

Raftery, A. y Dean, N. (2006). Variable selection for model-based clustering. *Journal of the American Statistical Association*, *101*(473), 168-178. Cita [01], [02], [03], [04].

Rosset, S., Zhu, J. y Hastie, T. (2004). Boosting as a regularized path to a maximum margin classifier. Journal of Machine Learning Research, 5(Aug), 941-973. Cita [01], [02].

Schwarz, G. (1978). "Estimation the Dimension of a Model." Annals of Statistics, 6, 461–464. Cita [01].

Scrucca, L. y Raftery, A. E. (2018). clustvarsel: A Package Implementing Variable Selection for Gaussian Model-based Clustering in R. *Journal of Statistical Software*, *84*. Cita [01].

Stamey, T., Kabalin, J., McNeal, J., Johnstone, I., Freiha, F., Redwine, E. y Yang, N. (1989) Prostate specific antigen in the diagnosis and treatment of adenocarcinoma of the prostate ii: radical prostatectomy treated patients. J. Urol., 16, 1076–1083. Cita [01], [02], [03].

Tamayo, P., Scanfeld, D., Ebert, B., Gillette, M., Roberts, C. y Mesirov, J. (2007). Metagene projection for cross-platform, cross-species characterization of global transcriptional states. *Proceedings of the National Academy of Sciences*, *104*(14), 5959-5964. Cita [01].

Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. Journal of the Royal Statistical Society: Series B (Methodological), 58(1), 267-288. Cita [01], [02].

Tibshirani, R., Walther, G. y Hastie, T. (2001). Estimating the number of clusters in a data set via the gap statistic. Journal of the Royal Statistical Society: Series B (Statistical Methodology), 63(2), 411-423. Cita [01].

Tibshirani, R., Saunders, M., Rosset, S., Zhu, J. y Knight, K. (2005). Sparsity and smoothness via the fused lasso. Journal of the Royal Statistical Society: Series B (Statistical Methodology), 67(1), 91-108. Cita [01], [02], [03], [04].

Tibshirani, R. y Taylor, J. (2011), "The solution path of the generalized lasso," Annals of Statistics, 39, 1335–1371. Cita [01].

Tibshirani, R. (2011). Regression shrinkage and selection via the lasso: a retrospective. Journal of the Royal Statistical Society: Series B (Statistical Methodology), 73(3), pp. 273 – 282. Cita [01], [02], [03], [04].

Tibshirani, R. (2013). The lasso problem and uniqueness. Electronic Journal of Statistics, 7, 1456-1490. Cita [01], [02].

Wang, S. y Zhu, J. (2008). Variable selection for model-based high-dimensional clustering and its application to microarray data. *Biometrics*, *64*(2), 440-448. Cita [01].

Ward, J. (1963), "Hierarchical Grouping to Optimize an Objective Function," Journal of the American Statistical Association, 58, 236–244. Cita [01].

Witten, D., Tibshirani, R. y Hastie, T. (2009). A penalized matrix decomposition, with applications to sparse principal components and canonical correlation analysis. *Biostatistics*, *10*(3), 515-534. Cita [01], [02], [03].

Witten, D. y Tibshirani, R. (2010). A framework for feature selection in clustering. *Journal of the American Statistical Association*, *105*(490), 713-726. Cita [01], [02], [03], [04], [05], [06].

Witten, M. y Tibshirani, R. (2013). sparcl: Perform sparse hierarchical clustering and sparse k-means clustering. *R package version*, *1*(3). Cita [01].

Xie, B., Pan, W. y Shen, X. (2008). Penalized model-based clustering with cluster-specific diagonal covariance matrices and grouped variables. *Electronic journal of statistics*, *2*, 168. Cita [01].

Yuan, M. y Lin, Y. (2006). Model selection and estimation in regression with grouped variables. Journal of the Royal Statistical Society: Series B (Statistical Methodology), 68(1), 49-67. Cita [01], [02].

Yuan, M. y Lin, Y. (2007), On the non-negative garrotte estimator, Journal of the Royal Statistical Society, Series B 69(2), 143–161. Cita [01].

Zhao, P. y Yu, B. (2006). On model selection consistency of Lasso. Journal of Machine learning research, 7(Nov), 2541-2563. Cita [01], [02].

Zou, H. y Hastie, T. (2005a). Regularization and variable selection via the elastic net. Journal of the royal statistical society: series B (statistical methodology), 67(2), 301-320. Cita [01], [02], [03], [04], [05], [06], [07].

Zou, H. y Hastie, T. (2005b). Addendum: regularization and variable selection via the elastic net. Journal of the Royal Statistical Society: Series B (Statistical Methodology), 67(5), 768-768. Cita [01].

Zou, H. (2006). The adaptive lasso and its oracle properties. Journal of the American Statistical Association, 101(476), 1418-1429. Cita [01], [02], [03], [04], [05], [06].