Una revisión del método Distance Weighted Discrimination

¿Una mejora de SVM en dimensiones altas ?

J. Antonio García Ramírez
Centro de Investigación en Matemáticas. Unidad Monterrey
jose.ramirez@cimat.mx

***Resumen*—El análisis estadístico de alta dimensión y tamaño de muestra pequeño (HDLSS) se está aplicando cada vez más en una amplia gama de contextos. En tales situaciones, se ve que el popular método de la Máquina de Vectores Soporte (SVM) sufre de "Acumulación de datos" en el margen, lo que puede disminuir la capacidad de generalización del modelo. Esto conduce al desarrollo de la *Distance Weighted Discrimination*, para encontrar un hiperplano separador . En el presente trabajo se revisa y reproduce, con detalle en la derivación y solución de la función de pérdida que se resuelve usando SOCP, del método desarrollado en [6] e implementado en el entorno R[9]. Basado en el trabajo e implementación de [12] se aplica y comparan resultados a conjuntos de datos reales y simulados (en medida de lo posible los mismos conjuntos de datos utilizados que en [6])**

Palabras clave: SVM, kernel, R (el ambiente de cómputo estadístico) y datos de alta dimensión con tamaño de muestra pequeño (data High Dimension Low Sample Size).

I. Introducción y motivación

Dentro del conjunto de herramientas estadísticas que Steve Marron ha desarrollado en los últimos años, y cuyo libro está en desarrollo [7], en lo que él define como *Análisis de datos orientado a objetos*, encuentran su lugar diferentes tipos y fuentes de datos que podríamos clasificar como complejos tales como texto, imágenes, video, audio y datos de alta dimensionalidad.
Un área emergente dentro de la estadística es el análisis de datos de alta dimensión d la mayoría de las veces mucho más grande que el tamaño de muestra n (conocido como HDLSS por sus siglas en inglés *High Dimension Low Sample Size*).

Ejemplos de áreas con el esquema HDLSS son: el análisis de microarreglos en genética (donde se presentan muy pocos casos y que se dispone de muchas expresiones de niveles de genes que son medidos en un espectro de alta dimensión) y en el análisis de imágenes en la medicina (donde una pequeña población de formas en tres dimensiones representadas por vectores de muchos parámetros y donde la segmentación de imágenes vuelve a presentarse [14]). En estas áreas los métodos clásicos del análisis multivariado son poco útiles pues el primer paso en el enfoque tradicional es hacer los datos esféricos (esto es multiplicarlos por el inverso de la raíz de la matriz de covarianzas, la cual no existe pues la matriz de covarianzas no es de rango completo[1]).

[1]Recordemos que tal matriz requiere de estimar $d(d+1)/2$ parámetros a partir de un limitado número de muestras n.

Entonces los datos de tipo HDLSS presentan una gran y fértil oportunidad para reinventar la mayoría de todos los tipos de inferencia estadística según [6], pág. 2.

En este trabajo revisamos, reproducimos y detallamos algunos aspectos del trabajo de Marron S. et al. [6], el cual se enfoca a la discriminación de dos clases con etiquetas $+1$ y -1 y es una novedosa vista del desempeño de las Máquinas de Vector Soporte (en lo que sigue denotado como SVM) propuesto por Vapnik en 1982 en configuraciones del tipo HDLSS por medio de proyectar los datos en el vector normal al hiperplano separador, esta vista refleja el fenómeno de acumulación de datos lo cual significa que muchas de estas proyecciones son la misma. Para el SVM, la acumulación de datos es común en contextos de HDLSS porque los vectores soporte tienden a ser numerosos en dimensiones altas y se amontonan en las vecindades del margen, esto afecta negativamente su desempeño de generalización.

La mayor contribución del trabajo presentado en [6] es un nuevo método de discriminación llamado *Distance Weighted Discrimination* (DWD), el cual evade el problema de acumulación de datos para mejorar la capacidad de generalización. El cálculo requerido por DWD presentado en el citado paper de 2007 usa programación de cono de segundo orden, en lo que sigue nos referimos a estos métodos como SOCP por sus siglas en inglés, desarrollado anteriormente a 2004 mientras que SVM utiliza los bien conocidos algoritmos de programación cuadrática.

La organización del presente trabajo es como sigue: En la sección II comenzamos revisando el el concepto de acumulación de datos, *data pilling*, así como su interpretación geométrica, en el caso de discriminación de dos clases con dos conjuntos de datos de dimensión elevada d donde las observaciones son independientes e idénticamente distribuidas de una distribución multivariada. Para continuar con una breve comparación entre DWD y SVM en cuanto a su objetivo e idea principal. En la sección III se detallan las deducciones (muy parecidas entre sí) de los problemas de optimización planteados por SVM y DWD, para DWD consideramos dos enfoques el dado en el 2007 por Marron et al. en [6] y otro más reciente de 2017 que generaliza e integra la noción de kernels en DVD y cuya implementación

[13] utilizamos para los experimentos de la sección IV, los cuales buscan reproducir los resultados de [6], por una parte con datos simulados y después con dos conjuntos de datos reales (uno del tipo HDLSS), a partir de lo observado en las simulaciones concluimos en la sección V con lo aprendido y verificado respecto a DWD y propuestas de trabajos futuros en el contexto de datos con alta dimensionalidad.

II. El fenómeno de acumulación de datos

II-A. Aspectos geométricos

El conocido fenómeno que se presenta al trabajar en dimensiones grandes, llamado maldición de la dimensionalidad [4] y por lo cual el tamaño de muestra para el análisis estadístico tradicional crece exponencialmente conlleva a que en dimensiones altas todas las posibles muestras de entrenamiento de un método de aprendizaje estadístico (en particular la discriminación o clasificación) pueblan escasamente el espacio de todas las entradas. En contextos HDLSS, donde el tamaño de muestra del que se dispone es pequeño se afecta el desempeño de generalización del SVM debido al fenómeno de acumulación de datos.

Para presentar el fenómeno de acumulación de datos consideremos un ejemplo de discriminación de dos clases, que se muestra en la figura 1 y la figura 2, donde se generó una muestra balanceada de tamaño 40 (20 etiquetas '+1' para las primer clase y el mismo número para la etiqueta '-1') proveniente de una distribución normal multivariada con matriz de correlación la identidad y con vector de medias cero (excepto en la primer coordenada donde se tienen los valores 2.2 para la primer clase y -2.2 para la segunda) en la figura 1 se muestra el caso de normalidad multivariado en dimensión 2 y en la figura 2 el caso de normal multivariado en dimensión 82.

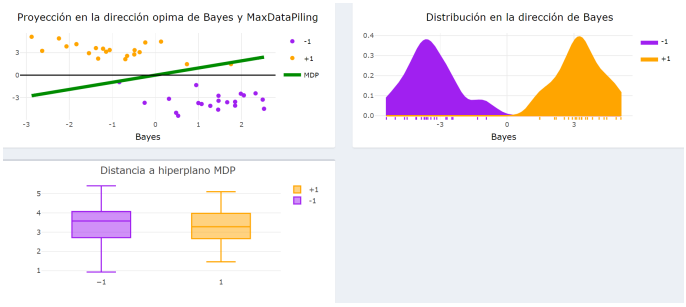

Figura 1. Ejemplo de acumulacion de datos ($d = 2$) en configuración no HDLSS. Proyección de la muestra (arriba izquierda) el eje x corresponde a la dirección óptima de Bayes, el eje y a un vector cualquiera ortogonal a x y la línea verde corresponde a la dirección de MDP w. Distribución de la proyección de la muestra en la dirección óptima de Bayes (arriba derecha) y su distancia al hiperplano separador (abajo izquierda).

En ambas simulaciones el eje x representa la dirección normal al hiperplano de separación óptimo de Bayes, pues los métodos de discriminación cuyo vector normal se acerca a esta dirección deberían tener buenas propiedades de generalización (ver [6] pág. 5) es decir que nuevos puntos serán clasificados de manera correcta, el eje y es simplemente una dirección ortogonal a x y la línea verde corresponde a la dirección de máxima acumulación, o bien MDP del inglés *maximal data pilling*, que es una de las direcciones que en configuraciones HDLSS posee la propiedad de que las dos clases se apilan completamente en solo dos puntos, uno para cada clase.

Existen varios ejemplos simples y extremos de direcciones donde se presenta la acumulación de datos en configuraciones HDLSS, para comenzar consideremos un vector ortogonal al subespacio generado por los datos [2] porque sus proyecciones son siempre cero, otro ejemplo importante corresponde al vector normal del hiperplano separador pues estas proyecciones determinan el margen que involucra SVM.

Continuando con las figuras 1 y 2, la dirección MDP es calculada como $w = \hat{\Sigma}^{-1}(\bar{x}^+ - \bar{x}^-)$, donde $\bar{x}^+$ y $\bar{x}^-$ representan respectivamente el vector de medias de la clase '+1' y '-1' y la inversa se considera como la inversa generalizada de la matriz de covarianzas [3] y como podemos ver en la parte superior izquierda de las figuras 1 y 2 los puntos de ambos conjuntos tienden a amontonarse en la dirección de w y mientras se incrementa d los puntos en cada clase aparentan ser colineales.

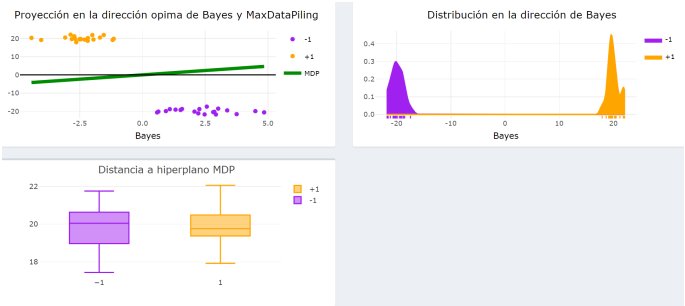

Figura 2. Ejemplo de acumulacion de datos ($d = 82$) en configuración HDLSS. Mismo orden que en la figura 1.

La acumulación de datos no es una propiedad útil en la clasificación pues el vector de dirección correspondiente depende fuertemente de aspectos muy particulares de la *realización* del conjunto de prueba del que se dispone. En configuraciones HDLSS con una cantidad razonable de ruido una realización diferente de la muestra provoca diferentes vectores w resultando en pobres propiedades de generalización.

En el código anexo al presente trabajo (carpeta 'DWD1') se incluye un Dashboard desarrollado con el framework Shinydashboard [11] , donde en su segundo tab contiene una simulación que genera las instancias mostradas en las

[2]Esto se garantiza pues la matriz de muestra X es de rango incompleto

[3]Recordando que dicha matriz es de rango incompleto, notemos su parecido con la recta que determina la discriminación lineal de Fisher (FLDA) sin embargo esta última no garantiza una dirección de acumulación de datos

figuras 1 y 2 pero d en el rango [2,1000] donde los efectos de acumulación son más notorios conforme d se incrementa.

Un resultado interesante contenido en [3] es que las configuraciones HDLSS bajo ciertas consideraciones multivariadas con tamaño de muestra fijo, tienden asintóticamente ($d \to \infty$) a tener una estructura geométrica fundamentalmente rígida mientras que la aleatoriedad aparece como rotaciones de la estructura.

II-B. Relación entre DWD y SVM

Por lo mencionado anteriormente un área de oportunidad en la mejora de la generalización de SVM en instancias HDLSS es la de permitir que más puntos del conjunto de entrenamiento tengan impacto en la dirección w o bien en la determinación del margen, lo que se traduce en reemplazar el criterio maximin basado en el margen (de SVM) por un criterio que sea función de las distancias r_i de los datos al hiperplano separador, una manera simple de implementar lo anterior es determinar una dirección w **que optimice la suma de los inversos de las distancias** dando más importancia a puntos cercanos al hiperplano y menos a los que están alejados o alternativamente (el enfoque dual) podemos prestar atención a la dirección normal al hiperplano separador entre las combinaciones convexas de los puntos en las clases, pero ahora las combinaciones se eligen para minimizar la suma de los inversos de las distancias. De esta manera todos los puntos reciben un peso positivo.

Hasta aquí hemos hablado de casos linealmente separables pero esto no es en general cierto, al igual que SVM el método de DWD aborda este problema incorporando penalizaciones por violación (datos que se localizan en el lado equivocado del hiperplano separador) lo cual conlleva a la selección y afinación de otro parámetro.

Como nota mencionamos que cuando $d >> n$ los datos consisten en un subespacio n dimensional y la idea de trabajar en este espacio es impráctica primero porque los nuevos datos se espera que aparezcan fuera de este subespacio y segundo en el contexto de microarreglos el interés recae en solo algunos subconjuntos de genes específicos.

III. Formulación del problema de optimización de DWD

En esta sección se comienza detallando la derivación del problema de optimización que plantea SVM utilizandola como base para la derivación del problema de optimización concerniente a DWD propuesta por Marron en [6]. Finalmente comentamos acerca del enfoque del algoritmo de SOCP y contrastandolo contra la generalización de [12] qué es mejor en tiempo de ejecución e incluye el caso de trabajar con kernels.

Fijemos la notación utilizada por [6]. Consideremos un conjunto de datos de entrenamiento de n d-vectores con entradas x_i, y con sus respectivas etiquetas $y_i \in \{+1, -1\}$. Denotemos por X la matriz $d \times n$ cuyas columnas son las x_i's y y el $n-$vector que contiene las y_i's. Las dos clases de los ejemplos de la sección anterior están en X. Las cantidades n_+ y n_- corresponden a $n_+ = \sum_{i=1}^{n} 1_{y_i=+1}$ y $n_+ = \sum_{i=1}^{n} 1_{y_i=-1}$ teniendo que $n = n_+ + n_-$.

Denotamos por Y a la matriz $n \times n$ con componentes y en la diagonal. Entonces si escogemos $w \in R^d$ como el vector normal del hiperplano separador y $\beta \in R$ como el intercepto, los residuos del $i-$esimo punto estan dados por:

$$\bar{r}_i = y_i(x_i^t w + \beta)$$

O en notación de matrices:

$$\bar{r} = Y(X^t w + \beta e) = YX^t w + \beta y$$

Donde $e \in R^n$ denota el vector de unos. Cuando los datos no sean linealmente separables denotaremos por $\epsilon \in R_+^n$ al vector de errores que servirá como penalizador, así los nuevos residuos serán

$$r = YX^t w + \beta y + \epsilon$$

Cuando los datos x_i se colocan en su adecuado lado del hiperplano separador $\epsilon_i = 0$ y tenemos que $\bar{r}_i = r_i$.

III-A. El problema de optimización de SVM

Consideremos primero el caso en que los dos clases son linealmente separables. Entonces siguiendo la derivación de [6] y complementando con la de que se puede consultar en [4] tenemos que el problema es el de maximizar el mínimo residuo, lo cual puede lograrse si introducimos una nueva variable δ y maximizando $\bar{r} \geq \delta e$, sin pérdida de generalidad podemos restringir a δ a la unidad y minimizar la norma de w o equivalentemente a la mitad de la norma de w al cuadrado para tener una solución convexa a maximizar. Si permitimos perturbaciones ϵ_i y penalizamos con la 1-norma de ϵ en la función objetivo a los datos mal clasificados tenemos que la función objetivo de minimización está dada por:

$$\min_{w,\beta,\epsilon} (1/2) w^t w + C e^t \epsilon \tag{1}$$

Sujeto a

$$YX^t + \beta y + \epsilon \geq e, \epsilon_i \geq 0 \tag{2}$$

Donde C es un parámetro de penalización que regularmente debe afinarse. La función de costo anterior, después de efectuar los productos punto, tiene un lagrangiano dado por :

$$\begin{aligned} L_{p-SVM} = \frac{||w||^2}{2} + C \sum_{i=1}^{n} \epsilon_i \\ - \sum_{i=1}^{n} \alpha_i [y_i(x_i^t w + \beta) - (1 - \delta_i)] - \sum_{i=1}^{n} \mu_i \epsilon_i \end{aligned} \tag{3}$$

Derivando lo anterior con respecto a w,β y ϵ_i e igualando a cero tenemos que :

$$w = \sum_{i=1}^{n} \alpha_i y_i x_i$$

$$0 = \sum_{i=1}^{n} \alpha_i y_i$$

$$\alpha_i = C - \mu_i, \forall i$$

Por las restricciones de positividad $\alpha_i, \mu_i, \epsilon_i \geq 0$ y sustituyendo lo anterior en (2) tenemos el problema dual:

$$L_{D-SVM} = \sum_{i=1}^{n} \alpha_i - (1/2) \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j y_i y_j x_i^t x_j \quad (4)$$

Lo anterior proporciona una cota inferior de (2)[4], si maximizamos L_{D-SVM} sujeto a $0 \leq \alpha_i \leq C$ y $\sum_{i=i}^{n} \alpha_i y_i = 0$ y utilizando las derivadas igualadas a cero, las condiciones de Karush-Khun-Ticher incluyen:

$$\alpha_i[y_i(x_i^t w + \beta) - (1 - \epsilon_i)] = 0$$

$$\mu_i \epsilon_i = 0$$

$$y_i(x_i^t w + \beta) - (1 - \epsilon_i) \geq 0$$

Lo cual garantiza la unicidad la solución del dual y del primal.

Sin embargo el dual es más fácil de optimizar con técnicas de programación cuadrática convexa y acepta una solución de la forma $\hat{w} = \sum_{i=1}^{n} \hat{\alpha}_i y_i x_i$, con coeficientes no cero α_i para ciertas observaciones las cuales corresponden a los vectores soporte los cuales se encuentran en la frontera del margen ($\hat{\epsilon}_i = 0$) y el resto ($\epsilon_i > 0$) tienen $\hat{\alpha}_i = C$ y β puede obtenerse de la primer condición de KKT.

Existe una analogía mecánica para la elección del hiperplano separador en el caso de SVM la cual consiste en imaginar que cada vector soporte ejerce una fuerza normal repulsiva al hiperplano, cuando la magnitud de las fuerzas son escogidas de manera apropiada el hiperplano está en equilibrio y solo los vectores soporte ejercen acción en el.

Notemos que en la función objetivo del primal y el dual existen productos punto de vectores consigo mismo lo que permite extender la idea de SVM incluyendo kernels.

Ya que hemos revisado la formulación de SVM la utilizaremos para replicar la deducción del problema de optimización de DWD, primero como la construyó Marron en [6] y posteriormente la generalización de [12].

[4]Porque el problema primal corresponde a minimizar

III-B. El problema de optimización de DWD (Clásico, 2007)

Originalmente, en el 2007, Marron et al. [6] deducen, como a continuación lo detallamos, el problema de optimización de DWD. Aunque en el mencionado trabajo se menciona que en teoría se puede elegir cualquier función convexa para formar la función de costo de DWD, en particular Marron elige la función $f(x) = \frac{1}{x}$ lo que le permite plantearlo como un problema SOCP.

Se elige como criterio de penalización que la suma de los recíprocos de los residuos perturbados por el vector de penalización ϵ sea minimizado (a diferencia de maximizar el mínimo como en SVM), entonces se plantea:

$$\min_{r,w,\beta,\epsilon} \sum_{i} (1/r_i) + Ce^t\epsilon \quad (5)$$

Sujeto a

$$r = YX^t w + \beta y + \epsilon$$

$$(1/2)w^t w \leq 1/2$$

$$r_i \geq 0, \epsilon_i \geq 0$$

Donde de nuevo $C = C_{DWD} > 0$ es un parámetro de penalización que debe afinarse. La condición de que la norma de w sea la unidad se relaja a que sea a lo más la unidad, esto hace el problema convexo y si los datos son linealmente separables la solución óptima tiene norma igual a 1.

La parte ingeniosa y delicada de la derivación consiste en eliminar los recíprocos utilizando restricciones de conos de segundo orden. Los conos de segundo orden se definen como las hipersuperficies que son de la forma:

$$S_{m+1} := \{(\eta, u) \in R^{m+1} : \eta \geq ||u||\}$$

Notemos en lo anterior que u es un vector de dimensión m. Considerando los conos mencionados se definen $r_i = \rho_i - \sigma_i$ donde $\rho_i = (r_i + 1/r_i)/2, \sigma_i = (1/r_i - r_i)/2$. Entonces $\rho_i^2 - \sigma_i^2 = 1$ o $(\rho_i, \sigma_i, 1) \in S_3$ y $\rho_i + \sigma_i = 1/r_i$. Entonces sustituyendo lo anterior en (5) la función de costo a minimizar se convierte en:

$$\min_{\phi,w,\beta,\epsilon,\rho,\sigma,\tau} Ce^t\epsilon + e^t\rho + \epsilon^t\sigma \quad (6)$$

Sujeto a

$$YX^t w + \beta y + \epsilon - \rho + \sigma = 0$$

$$\phi = 1$$

$$\tau = e$$

Con $(\phi, w) \in S_{d+1}, \epsilon_i \geq 0, (\rho_i, \sigma_i, \tau_i) \in S_3, i = 1, 2, \ldots, n$. Si bien la sustitución anterior es fácil, es importante notar que la condición $\phi = 1$ es el segundo detalle importante, pues esto transforma el problema de buscar un vector de norma menor a la unidad w en R^d en buscar un punto en el cono embebido en S_{d+1} lo cual en principio podría parecer extraño[5] sin embargo

[5]Pues incrementa la dimensión del espacio de búsqueda.

notemos que en la formulación anterior todas las restricciones son lineales. El problema anterior, y en general los problemas SOCP, tiene un dual amable el cual es:

$$\max_{\alpha} -||XY\alpha|| + 2e^t\sqrt{\alpha} \tag{7}$$

Sujeto a

$$y^t\alpha = 0$$

$$0 \leq \alpha \leq Ce$$

El dual anterior se asemeja mucho al problema dual correspondiente a SVM el cual maximiza $-(1/2)||XY\alpha||^2 + e^t\alpha$ Hasta aquí el planteamiento de la solución a DWD que planteo Marron en el 2007 [6], luce sencilla sin embargo la solución del problema no es fácil.

Resolver el primal o el dual que plantea DWD directamente es ineficiente porque en efecto la variable w es de dimensión $d >> n$ además de que el primal tiene $2n + 1$ ecuaciones y $3n + d + 2$ variables, Marron planteó lo siguiente para hacer el cómputo más sencillo: Primero consideramos la descomposición QR de X, con $Q \in R^{d\times n}$ con columnas ortonormales y $R \in R^{n\times n}$ triangular superior, entonces al reemplazar X por R se tiene en el problema primal $YR^t\hat{w}$, con $(\phi, \hat{w}) \in S_{n+1}$, entonces el número de restricciones depende de n no de d.

De lo anterior notemos que como $X^t = R^tQ^t$ cualquier solución factible $(\phi, \hat{w}, \beta, \epsilon, \rho, \sigma, \tau)$ del nuevo problema proporciona una solución factible de $(\phi, w, \beta, \epsilon, \rho, \sigma, \tau)$ en el problema original y como $w = Q\hat{w}$ entonces $||w|| = ||\hat{w}||$ y las soluciones tienen el mismo valor objetivo.

Existe también una interpretación mecánica de DWD, la cual es análoga a la de SVM. Se considera que cada punto ejerce una fuerza de $1/r^2$, cuyo potencial es $1/r$, entonces el hiperplano está en equilibrio si las fuerzas que actúan en él están en equilibrio (en este caso son más vectores que los vectores soportes de SVM).

Los problemas primal y dual de SVM son problemas de programación cuadrática convexa y se pueden resolver eficientemente por métodos primal-dual de punto interior los cuales para garantizar una precisión de ϵ requieren de $O(\sqrt{n}\ln(1/\epsilon))$ iteraciones, donde cada iteración requiere de resolver un sistema de ecuaciones cuadrado de dimensión n, en la práctica se requiere de entre 10 y 50 iteraciones. Para los problemas de tipo SOCP el primal y el dual de DWD, Marron afirma que existen eficientes métodos primal-dual de punto interior para resolverlos inclusive él utilizó la implementación en Matlab SDPT3 de [10] para el paquete de R 'DWD' [5], archivado desde el 2014 en el CRAN de R, sin embargo en [12] encontramos una implementación más reciente y otra perspectiva del problema.

A continuación daremos una reseña sobre lo que la implementación SDPT3 realizaba para resolver el problema de DWD: de nuevo teóricamente para alcanzar una precisión de ϵ se requiere de $O(\sqrt{n}\ln(1/\epsilon))$ iteraciones pero esta vez cada iteración requiere de resolver un sistema lineales de $n \times n$ con un costo de $O(n^3)$ [6], lo cual explica el porqué del desuso de la implementación [5]. En la siguiente subsección se verá la implementación actual de la solución general al problema de optimización que plantea DWD, incluyendo kernels y su respectiva programación en el paquete [13] aún disponible en CRAN (The Comprehensive R Archive Network).

III-C. El problema de optimización de DWD (Moderno, 2017) y el caso del Kernel

En el 2017 se publicó una nueva estrategia para resolver el problema de DWD. Como Marron señaló en el 2007 en [6] en principio cualquier función convexa podría utilizarse para asignar los pesos (en el caso de Marron $1/r$) de las distancias de los puntos al hiperplano separador, retomando esta idea Wang y Zou definen una función, $V_q(.)$ de pérdida general que considera cualquier potencia de la función de pérdida de Marron $(1/r)$, como sigue :

$$V_q(x) = \begin{cases} 1-u, & si \quad u \leq \frac{q}{q+1} \\ \frac{1}{u^q}\frac{q^q}{(q+1)^{q+1}}, & si \quad u > \frac{q}{q+1} \end{cases}$$

Después de probar que $V_q(.)$ tiene un gradiente Lipschitz continuo, en [12] se deriva un algoritmo del tipo mayorización de la minoría para resolver el problema de DWD con una función objetivo idéntica a la de Marron en [6] pero reemplazando $1/r$ por $V_q(x)$. En las secciones siguientes empleamos la implementación de este algoritmo dado en [12].

Una de las aportaciones de [12], aparte de una implementación del problema generalizado de DWD fue el de incluir la teoría necesaria para utilizar kernels en DWD. En [6] podemos encontrar que para considerar el caso de los kernels dicen lo siguiente, de manera poco formal y más bien algorítmica. Suponen una matriz M formada por el resultado de usar un kernel con las observaciones $M = K(x_i, x_j)$ y la factorizan como R^tR, proceden a reemplazar YX^tw por $YR^t\hat{w}$, al igual que en la observación que realizan para disminuir la dimensión. Ahora existen dos casos: Si $RY\alpha \neq 0$, se restringe a que $\hat{w}$ sea de la forma $RY\hat{\alpha}$ en el primal lo cual es equivalente a sustituir $YR^t\hat{w}$ por $YR^tRY\hat{\alpha}$ y $(\phi, \hat{w}) \in S_{n+1}$ reemplazado por $\hat{\alpha}^tTR^tRY\hat{\alpha} \leq 1$

Y la justificación de Marron et al. en [6] para la extensión a kernels de DWD recae en que de lo anterior se tiene:

$$YRRY = YMY = YK(x_i, x_j)Y = Y\Phi(X)^t\Phi(X)Y$$

Con lo cual se puede clasificar una observación nueva con el signo de $w^t\Phi(x) + \beta = \hat{\alpha}Y\Phi(X)^t\Phi(x) + \beta = \sum_i \hat{\alpha}_i y_i K(x_i, x) + \beta$.
Es decir que Marron et al. simplemente enchufaron una matriz de Kernel.

[6]Considerando la reducción de d a n variables utilizando la descomposición QR de X de la que ya se habló.

Por otro lado en [12] dan un tratamiento formal al problema de que DWD incluya Kernels, una prueba detallada se encuentra en la sección *4 Kernel distance-weighted discrimination in reproducing kernel Hilbert space and Bayes risk consistency* del paper. De manera general lo que hacen para fundamentar el uso de Kernels en DWD es considerar un kernel K de la misma manera que Marron en el 2007 y utilizar el teorema de representación de Wahba para probar que el error de clasificación de kernel DWD se aproxima al error de discriminación de Bayes, por lo que kernel DWD trabaja tan bien como la regla de Bayes (asintóticamente hablando).

III-D. Sobre la elección del parámetro C (en DWD) y problemas abiertos

Para situaciones no HDLSS Marron et al. [6] recomiendan afinar el parámetro C utilizando validación cruzada. En los casos HDLSS se recomienda escoger una constante grande y dividirla entre una distancia típica, en particular los autores consideran una distancia típica como la mediana de las distancias de puntos entre clases:

$$d_t = mediana\{||x_i - x_j|| : y_i = +1, y_j = -1\}$$

Sin embargo mencionan que elecciones más cuidadosas de este parámetro serán exploradas en papers próximos y lo plantean como un problema abierto. Sin embargo ellos usaron validación cruzada para sus experimentos.

Un problema abierto que plantean [6] es el de probar que DWD provee un clasificador que sea consistente con el riesgo de Bayes y si su versión Kernel lo es, resuelto en [12] como lo mencionamos en párrafos anteriores.

Otro problema que sí permanece abierto planteado en [6] es el de combinar la reducción de dimensión (o el problema de selección de variables) con las ideas de DWD.[7]

IV. Experimentos

En esta sección se buscó reproducir los resultados contenidos en [6] (en medida de lo posible), en general nuestros experimentos otorgan conclusiones similares con excepción del caso práctico de genes (contexto en el que fue desarrollado DWD) donde nuestras diferencias son significativas.

Si bien en [6] se reportan los resultados de la comparación entre la regla de discriminación por diferencia de medias, regresión logística regularizada, SVM y DWD nosotros reemplazamos la regla de discriminación por diferencia de medias (a pesar de que esta es la estimación máximo verosímil del discriminador óptimo de Bayes para distribuciones Gaussianas) por el método de separación basado en el vector que define la dirección de máxima acumulación (lo llamamos MDP) para considerarlo como una cota inferior de desempeño en dimensiones altas y con muestras no gaussianas. También incluimos en nuestras comparaciones el método Adaboost.

Las simulaciones y los ejemplos reales se ejecutaron en un equipo con la versión 3.4.4 del ambiente R,con 8 gb de memoria RAM procesador i7HQ y en los casos en que fue necesario (en particular los casos con dimensiones mayores a 400) se ejecutaron en una máquina virtual en la nube de Microsoft Azure con características similares pero con un solo núcleo. Para los experimentos se utilizaron las implementaciones de los paquetes e1071 [8], kerndwd [13], fastadabbost [1] y glmnet [2] para SVM, DWD, Adaboost y la regresión logística regularizada, el método de clasificar con el vector de MDP se implentó.

Para cada simulación reportamos los valores de los parámetros usados, en este punto existen importantes diferencias con respecto a los resultados de [6]

Consideramos dos conjuntos de experimentos los realizados sobre datos simulados y sobre dos conjuntos de datos reales (el código se anexa en el archivo 'simulaciones.R' y 'simulacionesReeales.R' respectivamente).

IV-A. Resultados de la simulación

La primer simulación consistió en generar una muestra aleatoria para entrenamiento de tamaños $n_- = n_+ = 25$ y tamaño de conjunto de prueba 200 igualmente balanceado.

Se varió la dimensión en el conjunto $\{10, 40, 100, 400, 1600\}$. Las observaciones corresponden a normales multivariadas con matriz de covarianza la identidad pero para el grupo +1 la media tiene un 2.2 en la primer coordenada y cero en las demás de manera análoga el grupo -1 tiene vector de medias -2.2 en la primer coordenada y cero en las demás. Esta simulación corresponde al caso sencillo pues por un resultado de análisis multivariado sabemos que las muestras condicionadas en su primer componente son normales y distinguir dos normales con media 2.2 y -2.2 se puede considerar un caso de separación linealmente separable.

El experimento anterior se replicó 100 veces. Para la regresión logística consideramos un parámetro de penalización de $\frac{1}{100}$ a diferencia de [6] donde menciona que este parámetro no vale la pena de ser afinado nosotros encontramos que sí lo vale pues su desempeño mejora considerablemente.

En este escenario, figura 3, confirmamos que DWD tiene un mejor desempeño que SVM, DWD se comporta en dimensiones altas como el clasificador con MDP. Notemos sin embargo que el desempeño obtenido por Adaboost y RLR son mucho mejores.

[7] Lo cual suena muy prometedor pues resuelven el problema de seleccionar variables en dimensiones altas lo cual en contextos genéticos es bastante útil.

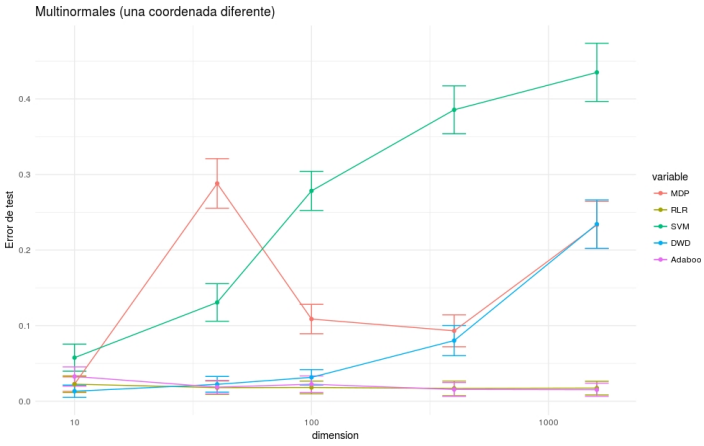

Figura 3. Resultados obtenidos de la primer simulación.Los puntos corresponden al error promedio y las bandas a la desviación estándar de las 100 ejecuciones por cada dimensión. Los parámetros para SVM, DWD y adaboost fueron fijados a 100 y para la regresión logística penalizada (RLR) fue de 0.001. El eje x está en escala logarítmica base 10.

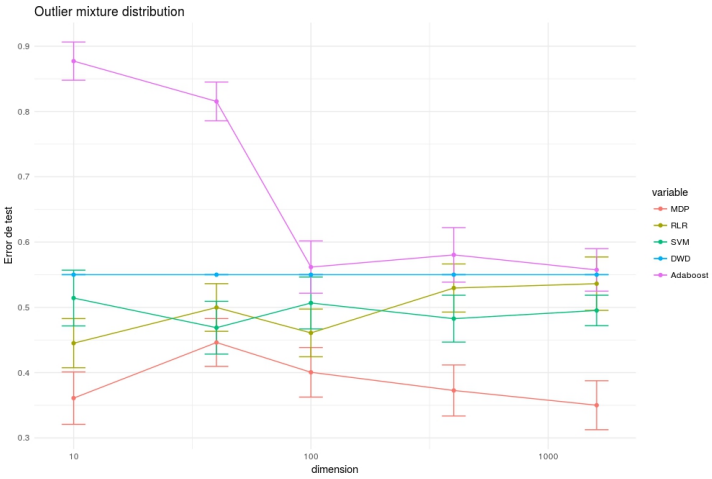

Figura 4. Resultados de la segunda simulación. Mismos valores de parámetros que en la figura 3. Nótese que los resultados son equivalentes a lanzar una moneda, pues el error promedio en el conjunto prueba es mayor a 0.5

Para la segunda simulación, figura 4, se simuló una muestra donde el 80 % de la muestra de entrenamiento y de prueba son análogas a las anteriores, el 20 % restante se divide, en partes iguales en las dos clases, en observaciones con la misma distribución que las primeras pero la diferencia radica en que la media en su primer componente se cambia de 2.2 a 100 y 500 y de -2.2 a -100 y -500. Este conjunto de datos busca identificar qué tan robustos o sensibles son los métodos de clasificación a outliers.

En este contexto es importante destacar el punto de que el nuevo 20 % de outliers no son vectores de soporte en SVM. Es importante notar las diferencias de desempeño de los clasificadores a lo largo de las dimensiones no es notoria para RLR y SVM pues sus intervalos se enciman. Lo sorprendente es **la estabilidad de DWD en cualquiera de las dimensiones simuladas**. Y como era de esperarse si se clasifica con la dirección de máximo apilamiento los resultados son mejores pues los outliers están demasiado lejos de los centros de cada clase por lo que son fáciles de reconocer. Notemos el desempeño de Adaboost el cual parece indicar que el método es más sensible en dimensiones bajas a outliers que en las altas[8]. En particular al mirar el error de clasificación en el conjunto de prueba para SVM es menor al de DWD, pues puede que estos valores atípicos sean vectores de soporte, sin embargo en dimensiones altas DWD y SVM tienen errores parecidos.

[8]O bien quizá su parámetro, el número de árboles con el que entrena, no es el adecuado.

Para la tercer simulación consideramos una muestra de datos de esferas anidadas, las cuales son altamente no gaussianas (a diferencia de los dos ejemplos anteriores). Con los mismos tamaños de muestra para el conjunto de entrenamiento y de prueba se simularon observaciones donde para las primeras $d/2$ provienen de una normal multivariada con centro en el vector 50 y matriz de covarianza la identidad para la etiqueta '+1' y mismo centro pero matriz de covarianza 49 veces la identidad para la etiqueta '-1', y para cada observación las restantes $d/2$ entradas son los cuadrados de las primeras.

Los resultados se muestran en la figura 5. En dimensiones bajas (<100) SVM parece distinguir mejor que DWD el tipo de observación (y a qué esfera pertenece) sin embargo para dimensiones altas (100>) su desempeño es similar al de DWD. Notemos que el criterio de discriminación de MDP es inútil en configuraciones de datos como estas (debido a la dependencia entre las componentes de las observaciones).

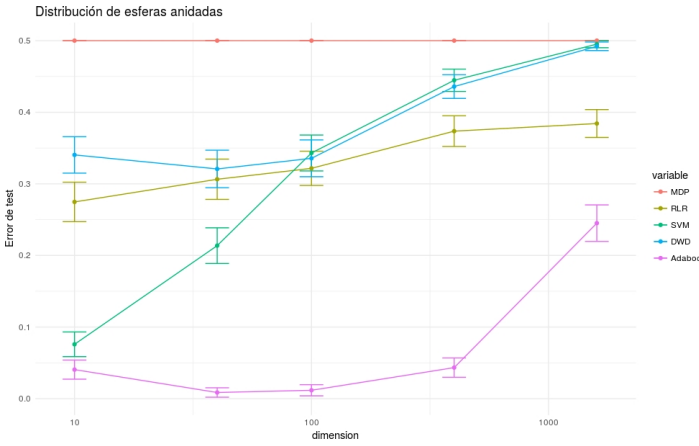

Figura 5. Resultados de la tercera simulación.Mismos valores de parámetros que en la figura 3.

Hasta este punto hemos comprobado que en conjuntos de datos gaussianos y en esferas anidadas DWD se desempeña de manera muy parecida a SVM pero que DWD es menos sensible a valores atípicos. Pero a diferencia de los resultados de [6] la regresión logística regularizada es un buen competidor a ambos métodos sobre todo en la simulación con valores atípicos y que Adaboost se desempeña mejor de los métodos ejemplificados (cuando la dimensión a tratar es mayor a 100).

En la siguiente sección se reportan los resultados obtenidos con los mismos métodos con datos reales.

IV-B. Resultados con datos reales

El primer experimento con datos reales se efectuo con el mismo conjunto de datos usado en [6] sobre cancer de mama[9].

A pesar de no ser una configuración HDLSS la meta de este estudio es clasificar $n = 569$ tumores malignos o benignos basados en $d = 30$ caracteristicas (observaciones y respuestas de las células) que resumen la información del tumor.

En la figura 6 se muestran los errores al entrenar y probar los cinco métodos utilizando k-fold validation[10] (con k=10) para cada valor del parámetro del conjunto $\{0.10, 63.79, 245.18, 544.24, 961.00\}$ para RLR, SVM y DWD (MDP no requiere de parámetros por lo cual su representación es una línea constante) en vista de que el parámetro considerado para Adaboost no es continuo se grafican los resultados para los parámetros $\{1, 65, 246, 545, 962\}$.

En este punto es en donde se encontraron diferencias notorias en cuanto a los resultados de [6] pues en este esquema DWD mejora el desempeño de SVM, **no así en nuestros resultados**. Esto puede deberse a variadas razones como que el software (algoritmo implementado) que se utiliza para cada uno de los métodos es diferente (la de Marron, concerniente a DWD está hecha en Matlab por ejemplo) o bien a los simuladores de números pseudoaleatorios o inclusive por la interpretación de los valores de los parámetros del software. En contrapunto de los resultados anteriores RLR se desempeña mal en este conjunto de datos (lo cual es ad hoc pues no se tiene el supuesto de normalidad de los datos).

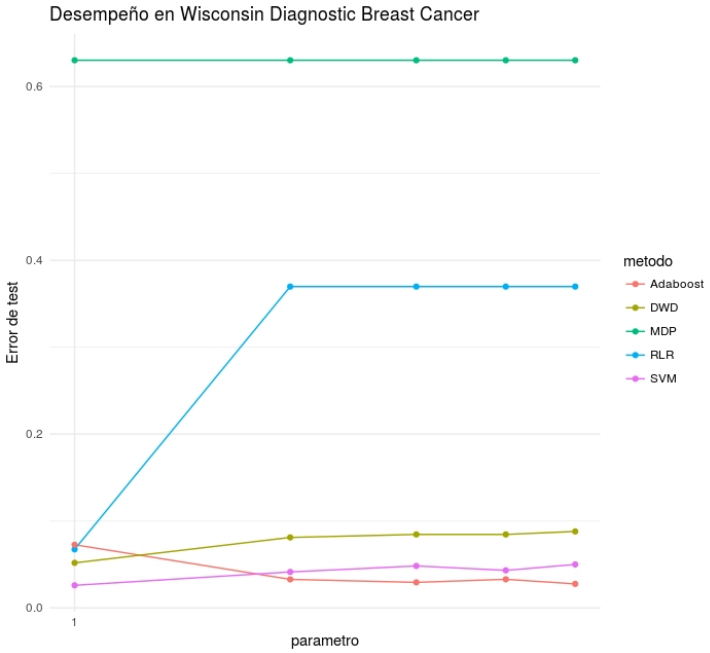

Figura 6. Evaluación de los cinco métodos en una instancia no HDLSS, en datos reales de cáncer de mama.

Finalmente, nuestro último experimento intentó replicar al de [6] respecto a análisis de microarreglos de genes. En el citado paper se utiliza una instancia que comprende $n = 136$ casos y la respuesta de $d = 456$ **genes previamente seleccionados de otro conjunto de genes** [11].

La anterior es una muestra tipo HDLSS sin embargo no son datos accesibles y abiertos. Por lo que para emularla se recurrio al conjunto de datos 'Gene expression cancer RNA-Seq' el cual es un conjunto de datos abierto[12]. Este conjunto de datos contiene 801 registros de pacientes con tipos de tumor BRCA, KIRC, COAD, LUAD y PRAD, además de sus secuencias de RNA (20,531 características).

Así que con el fin de efectuar un experimento parecido al de [6] con genes, el conjunto de datos de RNA se filtro para contener solo dos categorías 'COAD' y 'LUAD', quedando 219 registros de los cuales se tomaron aleatoriamente 136, de igual manera se escogieron de manera aleatoria 912 columnas (para contrarrestar el hecho de que nuestros genes no fueron

[9]Disponible en https://archive.ics.uci.edu/ml/datasets/Breast+Cancer+Wisconsin+%28Diagnostic%29

[10]Siguiendo el mismo proceso de [6]

[11]Ver [6] pág. 23

[12]Disponible en https://archive.ics.uci.edu/ml/datasets/gene+expression+cancer+RNA-Seq#

preseleccionados).

Los resultados de este experimento se muestran en la figura 7, donde al igual que en el ejercicio anterior se realizó validación cruzada k-fold (con $k = 5$). Los parámetros para RLR, SVM, DWD fueron {0.00, 0.71, 1.43, 2.14, 2.86, 3.57, 4.29, 5.00, 5.71, 6.43, 7.14, 7.86, 8.57, 9.29 , 10.00} mientras que para Adaboost se usaron los valores { 1 12, 23, 34, 45 , 57, 68, 79, 90, 101} para el número de árboles a entrenar.

En contra de lo que esperábamos obtener para los valores de parámetros mencionados, SVM se desempeño mejor que DWD en esta configuración HDLSS.

Esto lo atribuimos al hecho de que en nuestra muestra de columnas seleccionamos genes cuya respuestas no son suficientemente informativas para diferenciar los grupos a diferencia del trabajo de [6].

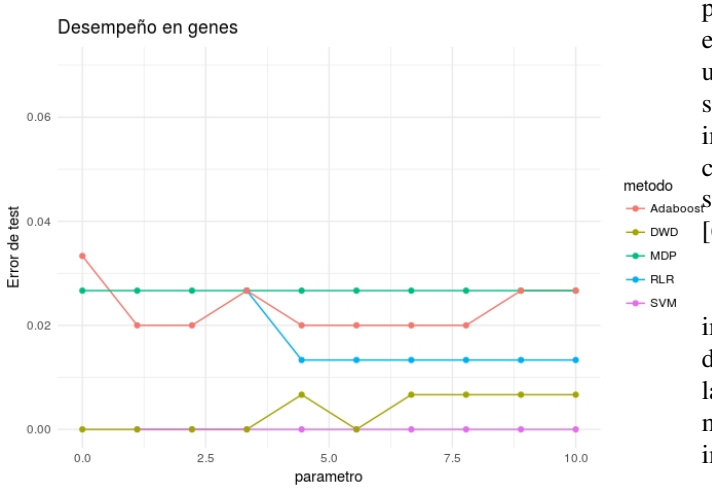

Figura 7. Evaluación de los cinco métodos (variando los parametros) en una instancia HDLSS, en datos reales de secuencias de RNA.

V. Conclusiones

Como primer conclusión tenemos que la principal fortaleza de DWD es que su desempeño es cercano al de SVM, cuando SVM es mejor, sin embargo DWD es sumamente estable a datos atípicos (como lo verificamos en nuestra segunda simulación). En configuraciones que no son del tipo HDLSS ambos métodos tienen desempeños parecidos en dimensiones bajas.
Verificamos con datos reales que DWD compite en desempeño inclusive con algoritmos como Adaboost (como lo muestra nuestro experimento con los datos de cáncer) y de manera general no encontramos discrepancia con los resultados contenidos en [6] excepto en que la regresión logística regularizada puede competir en altas dimensiones con DWD.

Esta revisión de DWD nos enseño aspectos valiosos en el cómputo estadístico, por un lado nos deja claro la diferencia entre modelación y cómputo, pues si bien Marron planteó su enfoque de DWD desde el 2007, tuvieron que pasar 10 años para una implementación robusta (como la que utilizamos en este proyecto). Y que no necesariamente el desarrollador de un método es quien lo implementa. Una lección aprendida es siempre revisar el trasfondo de los algoritmos que ya están implementados. Además de recordarnos lo importante que es contar con la experiencia de alguien en el campo, hecho que se ve reflejado en las diferencias de desempeño de DWD en [6] y en nuestro segundo experimento con datos reales.

Como nota cultural la revisión de DWD, históricamente es importante pues señala diferentes aspectos en el desarrollo de una nueva metodología. Primero la conceptualización, la implementación y en el caso de DWD la teorización matemática que lo generalice a más casos y la mejora de una implementación anterior.

En el caso particular de genes vimos que el desempeño de DWD no es mejor que el de SVM (en nuestra instancia de prueba) como lo comprobamos con nuestro experimento con datos de RNA, sin embargo este experimento en particular **resalta la necesidad de un criterio de selección de variables en dimensiones altas.**

Finalmente no podemos decir que DWD resuelva el problema clasificar en alta dimensión pero sí es una mejora con respecto a SVM en desempeño en algunas circunstancias y estabilidad a datos atípicos.

REFERENCIAS

[1] Chatterjee S. (2016), *fastAdaboost: a Fast Implementation of Adaboost*, R package version 1.0.0, https://CRAN.R-project.org/package=fastAdaboost

[2] Friedman J, Hastie T, Tibshirani R. (2010). *Regularization Paths for Generalized Linear Models via Coordinate Descent*. Journal of Statistical Software, 33(1), 1-22. http://www.jstatsoft.org/v33/i01/.

[3] Hall, P., Marron, J. S. and Neeman, A. (2004), *Geometric representation of high dimension low sample size data*, submitted to Journal of the Royal Statistical Society, Series B.

[4] Hastie, T., Tibshirani, R. and Friedman, J. (2001), *The Elements of Statistical Learning*, Springer Verlag, Berlin.

[5] Huang H., Lu X., Liu Y.,Marron J. S., Haaland P., *DWD: Distance Weighted Discrimination and Second Order Cone Programming*, archivado el 19-04-2014 en https://cran.r-project.org/web/packages/DWD/index.html

[6] Marron, J.S., Todd, M.J., Ahn, J. (2007) *Distance-Weighted Discrimination*, Journal of the American Statistical Association, 102(408), 1267–1271. https://faculty.franklin.uga.edu/jyahn/sites/faculty.franklin.uga.edu.jyahn/files/DWD3.pdf

[7] Marron, S. (2018, Mayo 16). *J. S. Marron (Steve Marron)*. Recuperado de http://marron.web.unc.edu/marron-dryden-ooda-book/

[8] Meyer D. and Dimitriadou E. and Hornik K. and Weingessel A. and Leisch F. (2017), *e1071: Misc Functions of the Department of Statistics, Probability Theory Group (Formerly: E1071)*, R package version 1.6-8, https://CRAN.R-project.org/package=e1071,

[9] R Core Team, *R: A Language and Environment for Statistical Computing*, R Foundation for Statistical Computing; Vienna, Austria, 2014 y = http://www.R-project.org/

[10] Tütüncü, R. H., Toh, K. C., and Todd, M. J. (2001b), *SDPT3 — a MATLAB software package for semidefinite-quadratic-linear programming*, ya no disponible en http://www.math.cmu.edu/reha/home.html

[11] Winston Chang and Barbara Borges Ribeiro (2018),*shinydashboard: Create Dashboards with 'Shiny'*, R package version 0.7.0, https://CRAN.R-project.org/package=shinydashboard

[12] Wang, B. and Zou, H. (2017) *Another Look at Distance Weighted Discrimination*, Journal of Royal Statistical Society, Series B, 80(1), 177–198. https://rss.onlinelibrary.wiley.com/doi/10.1111/rssb.12244

[13] Wang B. y Zou H. (2017), *kerndwd: Distance Weighted Discrimination (DWD) and Kernel Methods*, versión 2.0.2, https://CRAN.R-project.org/package=kerndwd

[14] Yushkevich, P., Pizer, S. M., Joshi, S. and Marron, J. S. (2001), *Intuitive, Localized Analysis of Shape Variability*, en Information Processing in Medical Imaging (IPMI), eds. Insana, M. F. and Leahy, R. M., 402—408.