Regularización en un modelo con Múltiples Respuestas

Proyecto final, Optimización

Autores: Enrique Santibáñez & Victor Manuel Martínez

02 de Junio de 2021

Centro de Investigación en Matemáticas,
Maestría en Cómputo Estadístico.

Contenido

Platamiento del problema

Definición del problema

Planteamiento del problema

Sea $X \in \mathbb{R}^{n \times p}$ una matriz de $n \times p$ con n ejemplos de entrenamiento y p características.

Y $\in \mathbb{R}^{n \times q}$ tal que cada fila representa q respuestas

Análogamente a encontrar una función $f(\vec{x}_{1 \times p})$ que prediga las q salidas de un vector de entrada $\vec{x}_{1 \times p}$, el objetivo será encontrar B en el modelo definido como Y=XB, donde B es de tamaño $p \times q$

Importancia y Objetivo del proyecto

Importancia y Objetivo del proyecto

- En altas dimensiones $p > n$ $X^t X$ podria no ser invertible
- La interpretación con muchos predictores podria resultar dificil
- Evitar el sobreajuste de un modelo

Enfoque general.

La regresión multivariada es una generalización del modelo de regresión clásico pero considerando $q > 1$ variables respuestas. Es decir, sea X la matriz de las variables independientes $n \times p$, Y la matriz de las variables independientes $n \times q$ y sea E la matriz de error aleatorio $n \times q$. Entonces el modelo de regresión multivariada es

$$Y = XB + E, \tag{2.1}$$

donde B es la matriz de coeficientes de regresión $p \times q$.

La función de verosimilitud logarítmica negativa de (B, Ω), donde $\Omega = \Sigma^{-1}$ se puede expresar como

$$g(B, \Omega) = \left[\frac{1}{n}(Y - XB)^T (Y - XB)\Omega\right] - \log(\det(\Omega)) \qquad (2.2)$$

Es fácil ver (derivando con respecto a B e igualando a 0, y simplificando), que el estimador de máxima verosimilitud de B es

$$\hat{B}^{OLS} = (X^T X)^{-1} X^T Y. \qquad (2.3)$$

Lo anterior es equivalente a realizar las estimaciones de B utilizando mínimos cuadrados ordinarios de forma separada para cada una de las q variables de respuestas y no este implica que no dependan de Ω.

Analisis del problema

El problema definido (2.2) también se puede modificar para agregar un parámetro de regularización. Denotando a $C(B)$ como el parámetro de regularización en función de B, es decir,

$$\hat{B} = \underset{B}{argmin} \, \|Y - XB\|_2^2 = \underset{B}{argmin} \, [(Y - XB)^T (Y - XB)] \quad (2.4)$$

$$\text{sujeto a:} \quad C(B) \leq t$$

Y se puede mostrar que el problema anterior es equivalente a resolver el problema (ver (Peng y col., 2010)).

$$\hat{B} = \underset{B}{argmin} \; \{ \quad \|Y - XB\|_2^2 - \lambda(C(B)) \; \} \quad (2.5)$$

Del planteamiento inicial podemos observar dos enfoques distintos cuando se considera una regresión multivariada.

1. Considerar que los datos no están correlacionados.

En este trabajo presentamos estos dos enfoques, pero agregamos un parámetro de regularización.

Del planteamiento inicial podemos observar dos enfoques distintos cuando se considera una regresión multivariada.

1. Considerar que los datos no están correlacionados.
2. Considerar la matriz de covarianzas de los errores para estimar B.

En este trabajo presentamos estos dos enfoques, pero agregamos un parámetro de regularización.

Metodología

REMMAP

El problema de minimización con restricciones propuesto por (Peng y col., 2010), considera una optimización L1 y L2, considera tambien que las q respuestas observadas no estan correlacionadas, la función a optimizar es representada como:

$$L(\hat{B}, X, Y) = \underset{B}{argmin}\{\frac{1}{2}\sum_{k=1}^{q}(y_k - xB_k)^2\} + \lambda_1 \sum_{k=1}^{q}|B_k| + \lambda_2\sqrt{\sum_{k=1}^{q}(B_k)^2}$$

Considerando $\lambda_2 = 0$, pues solo trabajaremos con la restricción de norma L1, tenemos:

$$L(\hat{B}, X, Y) = \underset{B}{argmin}\{\frac{1}{2}\sum_{k=1}^{q}(y_k - xB_k)^2\} + \lambda_1 \sum_{k=1}^{q}|B_k|$$

REMMAP

La actualización de los parametros se realiza considerando un descenso coordinado para cada elemento en b_{jk}, estamos frente a un algoritmo de orden $O(\text{TPQ})$.

De acuerdo con (Peng et al, 2010) la actualización de cada elemento $b_{j.k}$ se realiza tal que:

$$\hat{B}_{j_0,k} = (|X_{j_0}^T \tilde{Y}_k| - \lambda_1)_+ \frac{sign(X_{j_0}^T \tilde{Y}_k)}{\|X_{j0}\|_2^2}$$

$$Donde \qquad \tilde{Y}_k = Y_k - \sum_{j \neq j_0} X_j B_{jk}$$

En palabras $\tilde{Y}_k$ es el residual que se obtiene de ajustar los pesos sin considerar la j-esima variable, la cual se esta optimizando, esta es la escencia del gradiente por descenso coordinado.

El pseudocodigo de la función

Input: $Y_{n\times q}$,$X_{n\times p}$

Result: $B_{p\times q}$

Inicializamos parametros, $[B = 0_{p\times q}, ...]$;

while *True* **do**

Para j=1,...p ; k=1,...,q

$$B_{j_0,k} = (|X_{j_0}^T \tilde{Y}_k| - \lambda_1)_+ \frac{sign(X_{j_0}^T \tilde{Y}_k)}{\|X_{j0}\|_2^2}$$

if *B no cambia* **then**

break;

return(B);

end

end

Algorithm 1: REMMAP (Peng y col., 2010)

Caso cuando $\lambda_2 \neq 0$

(Peng et al, 2010) muestran que la actualización se realiza tal que la $B_{j0,k}$ ya calculada que llamaremos $B_{j0,k}^{lasso}$, es evaluada nuevamente en una función donde

$$\hat{B}_{j0,k} = (1 - \frac{\lambda_2}{\|B_j^{lasso}\|_2 \quad \|X_k\|_2^2})_+ (b_{j0,k}^{lasso})$$

$\hat{B}_{j0,k} = 0$ si $\|B_j^{lasso}\|_2 = 0$

MRCE

Rothman y col., 2010 plantea un procedimiento para construir un estimador de una matriz de coeficientes de regresión multivariada que tenga en cuenta la correlación de las variables de respuesta. Básicamente propone un estimador para B que considera los errores correlacionados utilizando la verosimilitud normal. Considera dos penalizaciones a la verosimilitud logarítmica negativa (2.2) para construir un estimador disperso B que dependa de $\Omega = \{\omega_{j'j}\}$,

$$(\hat{B}, \hat{\Omega}) = \arg\min_{B,\Omega} \left\{ g(B, \Omega) + \lambda_1 \sum_{j' \neq j} |\omega_{j'j}| + \lambda_2 \sum_{j=1}^{p} \sum_{k=1}^{q} |b_{jk}| \right\} \quad (3.1)$$

donde $\lambda_1 \geq 0$ y $\lambda_2 \geq 0$ son los parámetros de regularización.

El problema de optimización en (3.1) no es convexo, sin embargo, resolver B o Ω con el otro parámetro fijo hace al problema convexo. Entonces, si dejamos fijo B en un punto B_0 el problema de optimización para Ω se convierte a

$$\hat{\Omega}(B_0) = \arg\min \left\{ tr\left(\hat{\Sigma}_R \Omega\right) - \log(\det \Omega) + \lambda_1 \sum_{j' \neq j} |\omega_{j'j}| \right\}, \quad (3.2)$$

donde $\hat{\Sigma}_R = \frac{1}{n}(Y - XB_0)^T(Y - XB_0)$. Este problema es conocido como el problema de estimación de covarianza considerando una penalización L_1. Friedman y col., 2008 plantea el algoritmo de LASSO gráfico para resolver el problema de optimización 3.2.

LASSO gráfico

Sea W el estimador para Σ (matriz de covarianza poblacional). Se puede mostrar que se puede resolver el problema optimizando cada fila y la columna correspondiente a W en una forma de descenso de coordenadas de bloque. Partimos W y S,

$$W = \begin{pmatrix} W_{11} & w_{12} \\ w_{12}^T & w_{22} \end{pmatrix}, \quad S = W = \begin{pmatrix} S_{11} & s_{12} \\ s_{12}^T & s_{22} \end{pmatrix}$$

donde S es la matriz de correlación empírica. Entonces se puede mostrar que

$$w_{12} = \arg\min_y \left\{ y^T W_{11}^{-1} y : \|y - s12\|_\infty \geq p \right\}. \tag{3.3}$$

Pero de igual manera se puede mostrar usando dualidad convexa que el problema (3.3) es equivalente a resolver el problema dual

$$\min_\beta \left\{ \frac{1}{2} |W_{11}^{-1/2} \beta - b|^2 + \lambda |\beta|_1 \right\} \tag{3.4}$$

donde $b = W_{11}^{-1/2} s_{12}$.
Si β resuelve (3.4) entonces $w_{12} = W_{11}\beta$ resuelve (3.3).
Para resolver (3.4) usamos W_{11} y s. Luego actualizamos w y corremos todas las variables hasta la convergencia. Consideramos que la solución de $w_{ii} = s_{ii} + \lambda$ para todo i. Este algoritmo se le conoce como algoritmo LASSO gráfico (ver **Algoritmo** 2).

Input: S, λ y ϵ.

Result: W

Inicializamos

$W = S + \lambda \mathbf{I}$

while $|W^{(m)}_{-diag} - W^{(m-1)}_{-diag}| > \epsilon |S_{-diag}|$ **do**

  for *j=1,2,*$\cdots, p, 1, 2, \cdots, p, \cdots$ **do**

    Resolver el problema de LASSO en (3.4). Esto regresa un vector solución $\hat{\beta}$ *de tamaño* $p-1$*, por lo que imputamos el renglón y la columna de* W *usando* $w_{12} = W_{11}\hat{\beta}$.

  end

end

return$\left(W^{-1}\right)$

Algorithm 2: LASSO gráfico (Friedman y col., 2008)

Para resolver el problema de LASSO en (3.4) del (**Algoritmo** 2) consideramos un descenso coordinado. Sea $V = W_{11}$ y $u = s_{12}$, entonces actualizamos β_j de la forma

$$\hat{\beta}_j = S(u_j - \sum_{k \neq j} V_{jk}\beta_k, \lambda)/V_{jj} \tag{3.5}$$

para $j = 1, 2, \cdots, p, 1, 2, \cdots, p, \cdots$. Donde S es el operador soft-threshold:

$$S(x, y) = sign(x)(|x| - t)_+.$$

Para más detalle de este algoritmo consulte Friedman y col., 2008, ahí se presentan las demostraciones más a detalle y más referencias sobre problemas similares.

Estimación para B.

Por otro lado, resolver 3.1 fijando Ω en un punto elegido Ω_0 transforma el problema a optimizar

$$\hat{B}(\Omega_0) = \arg\min \left\{ tr\left(\frac{1}{n}(Y - XB)^T (Y - XB)\Omega_0 + \lambda_2 \sum_{j=1}^{p} \sum_{k=1}^{q} |b_{jk}| \right) \right\} \tag{3.6}$$

Una solución para el problema anterior es utilizar un descenso de coordenadas cíclicas. Rothman y col., 2010 resume en el procedimiento de optimización como se describe en el **Algoritmo 3**. Se utiliza la estimación de mínimos cuadrados penalizados por rigde $\hat{B}^{ridge} = (X^T X + \lambda_2 I)^{-1} X^T Y$ para escalar nuestra prueba de convergencia de párametroas, ya que siempre está bien definida (incluso cuando $p > n$).

Algoritmo 3.

Input: $Y_{n\times q}, X_{n\times p}, \Omega_{p\times p}, \lambda_2$ y ϵ

Result: $\hat{B}_{p\times q}$

$S = X^T X, \quad H = X^T Y \Omega, \quad \hat{B}^{rigde} = (X^T X + \lambda_2 I)^{-1} X^T Y.$

while $\sum |\hat{B}^{(m)} - \hat{B}^{(m-1)}| > \epsilon \sum |\hat{B}^{rigde}|$ **do**

  for $r=1,...p$ **do**

    for $c=1,...,q$ **do**

      $\mu_{rc} = \sum_{j=1}^{p} \sum_{k=1}^{q} \hat{b}_{jk}^{(m)} s_{rj} w_{kc}$

      $\hat{b}_{rc}^{(m)} =$

      $sign\left(\hat{b}_{rc}^{(m)} + \frac{h_{rc}-\mu_{rc}}{s_{rr}\omega_{cc}}\right)\left(\left|\hat{b}_{rc}^{(m)} + \frac{h_{rc}-\mu_{rc}}{s_{rr}\omega_{cc}}\right| - \frac{n\lambda_2}{s_{rr}\omega_{cc}}\right)_+$

    end

  end

end

return$\left(\hat{B}^{(m)}\right)$

Algorithm 3: Descenso de coordenadas cíclicas (Rothman y col., 2010)

MRCE

- Considerando lo anterior, podemos resumir la resolución del problema de optimización 3.1 usando el descenso de coordenadas en bloque, es decir, iteramso minimizando con respecto a B y miniminzando con respecto a Ω.

MRCE

- Considerando lo anterior, podemos resumir la resolución del problema de optimización 3.1 usando el descenso de coordenadas en bloque, es decir, iteramso minimizando con respecto a B y miniminzando con respecto a Ω.
- El **Algoritmo 4** usa el descenso de coordenadas por bloques para calcular una solución loca para 3.1.

Algoritmo MRCE

Input: $Y_{n\times q}$,$X_{n\times p}$, $\lambda_1,\ \lambda_2, \epsilon = 1e-4$.

Result: $\hat{B}_{p\times q}$

Inicializamos

$\hat{B}^{(0)} = 0$

$\hat{\Omega}^{(0)} = \hat{\Omega}(\hat{B}_{(0)})$

while $\sum|\hat{B}^{(m)} - \hat{B}^{m-1}| > \epsilon \sum|\hat{B}^{rigde}|$ **do**

1. Calcular $\hat{B}^{m+1} = \hat{B}(\hat{\Omega}^{(m)}$ resolviendo 3.6 utilizando el **Algoritmo 3**.
2. Calcular $\hat{\Omega}^{(m+1)} = \hat{\Omega}(\hat{B}^{(m+1)})$ resolviendo 3.2 usando el algoritmo de LASSO gráfico.

end

return$\left(\hat{B}^{(m)}\right)$

Algorithm 4: MRCE (Rothman y col., 2010).

Resultados

Resultados

- Para verificar el rendimientos de los modelos descritos en las secciones anteriores, consideramos probar las estimaciones de los algoritmos REMMAP y MRCE, utilizando MSE como métrica para comparar los errores por la facilidad de comprender la misma. Estos algoritmos se implementaron en el lenguaje de programación Python, en el sistema x86_64, Ubuntu.

Resultados

- Para verificar el rendimientos de los modelos descritos en las secciones anteriores, consideramos probar las estimaciones de los algoritmos REMMAP y MRCE, utilizando MSE como métrica para comparar los errores por la facilidad de comprender la misma. Estos algoritmos se implementaron en el lenguaje de programación Python, en el sistema x86_64, Ubuntu.

Resultados

- Para verificar el rendimientos de los modelos descritos en las secciones anteriores, consideramos probar las estimaciones de los algoritmos REMMAP y MRCE, utilizando MSE como métrica para comparar los errores por la facilidad de comprender la misma. Estos algoritmos se implementaron en el lenguaje de programación Python, en el sistema x86_64, Ubuntu.

Observación

La paquetería de Python Scikit-learn (Pedregosa y col., 2011), tiene una función en donde esta implementado el **Algoritmo** 4. Entonces esta función nos ayudo a determinar si nuestra implementación estaba bien.

Conjunto de datos

- El conjunto de datos sinteticos fue generado con la función $make_regression()$ de la librería de Scikit-learn. Consideramos diferentes parámetros de la función anterior: $n\_samples(n) = [100, 20]$, $n\_features(p) = [20, 100]$, y $n\_targets(q) = [2, 5]$.

Conjunto de datos

- El conjunto de datos sinteticos fue generado con la función $make_regression()$ de la librería de Scikit-learn. Consideramos diferentes parámetros de la función anterior: $n\_samples(n) = [100, 20]$, $n\_features(p) = [20, 100]$, y $n\_targets(q) = [2, 5]$.
- Consideramos partir el conjunto de datos original, en dos conjuntos uno de prueba y otro de entrenamiento.

Conjunto de datos

- El conjunto de datos sinteticos fue generado con la función $make_regression()$ de la librería de Scikit-learn. Consideramos diferentes parámetros de la función anterior: $n\_samples(n) = [100, 20]$, $n\_features(p) = [20, 100]$, y $n\_targets(q) = [2, 5]$.
- Consideramos partir el conjunto de datos original, en dos conjuntos uno de prueba y otro de entrenamiento.
- Además de que nuestro conjunto de datos, consideramos una estandarización debido a los supuestos que se tienen en los modelos.

Primeros conjunto de datos

Observando la **Figura 3**, podemos notar que cuando se consideran tamaños de $n < p$ notamos que los mejores predictores son ocupando la metodología de REMMAP.

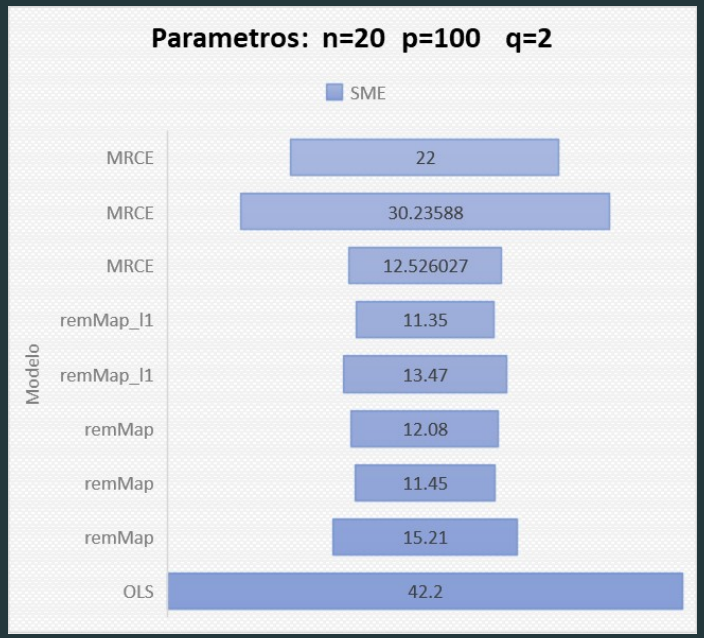

Figura 1

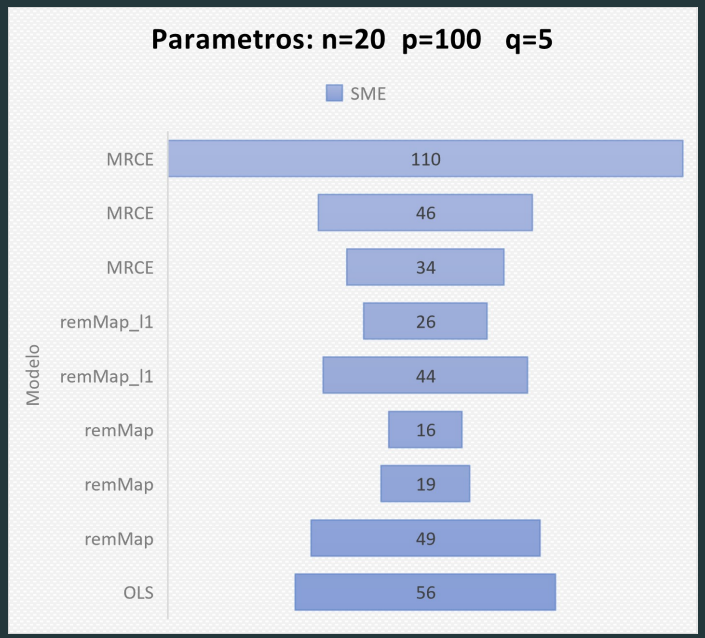

Figura 2

Figura 3: MSE considerando distintos modelos, con $n < p$.

Si comparamos los mejores modelos se observa claramente que en general los la metodología de MRCE tiene rendimientos similares que los estimadores de máxima verosimilitud. Pero, en general la metodología REMMAP es mejor.

Modelo	MSE	Parámetros	Numero de variables
MRCE	12.52	$\lambda_1 = 0{,}1 \lambda_2 = 0{,}5$	21
*REMMAP_I*1	11.35	$\lambda_1 = 0{,}1 \lambda_2 = 0$	15
OLS	42.2	–	–

Cuadro 1: Parámetros: n=20 , p=100 , q=2

Modelo	MSE	Parametros	Numero de variables
MRCE	34	$\lambda_1 = 0{,}1 \lambda_2 = 0{,}5$	10
*REMMAP_I*1	26.47	$\lambda_1 = 0{,}1 \lambda_2 = 0$	40
OLS	42.2	–	–

Cuadro 2: Parametros: n=20 , p=100 , q=5

Últimos conjuntos de datos.

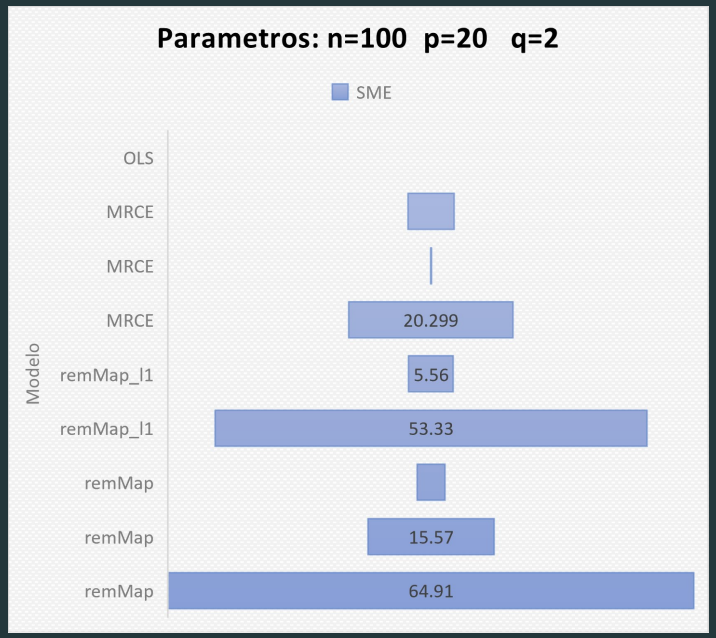

Figura 4

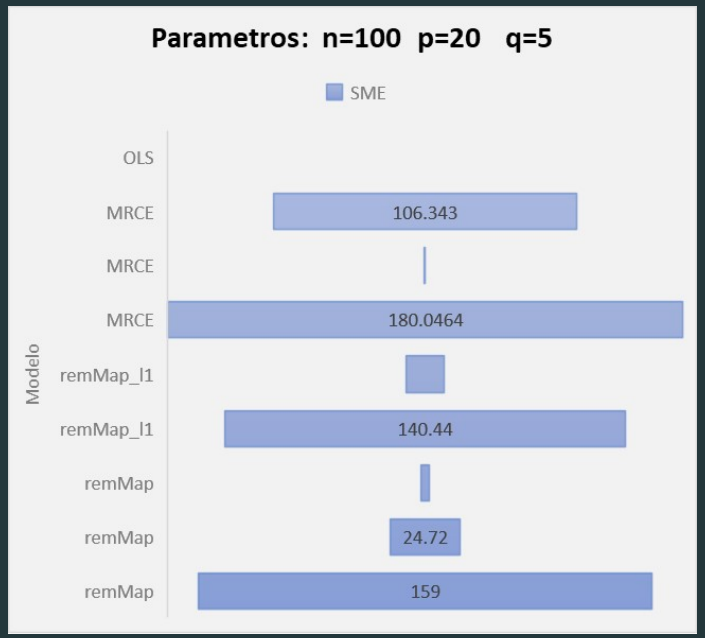

Figura 5

Figura 6: MSE considerando distintos modelos, con $n > p$.

Modelo	MSE	Parámetros	Numero de variables
MRCE	0.1227	$\lambda_1 = 0{,}1 \lambda_2 = 0{,}1$	14
REMMAP	3.43	$\lambda_1 = 1e-20 \lambda_2 = 0{,}1$	8
OLS	0.1e-16	–	–

Cuadro 3: Parámetros: n=100 , p=20 , q=2

Modelo	MSE	Parámetros	Numero de variables
MRCE	0.3224	$\lambda_1 = 0{,}1 \lambda_2 = 0{,}1$	33
REMMAP	2.89	$\lambda_1 = 1e-20 \lambda_2 = 0{,}1$	20
OLS	0.1e-16	–	–

Cuadro 4: Parámetros: n=100 , p=20 , q=5

Conclusiones

Conclusiones

- Primeramente podemos observar que se cumplió el objetivo principal de este trabajo, determinar una función f tal que sirva como función predictora usando un conjunto de datos X y Y planteándolo como un problema de optimización. Para resolver este problema, utilizamos distintos algoritmos de optimización:

Conclusiones

- Primeramente podemos observar que se cumplió el objetivo principal de este trabajo, determinar una función f tal que sirva como función predictora usando un conjunto de datos X y Y planteándolo como un problema de optimización. Para resolver este problema, utilizamos distintos algoritmos de optimización:
  - **LASSO gráfico** (2) (descenso de coordenadas por bloque y descenso coordinado)

Conclusiones

- Primeramente podemos observar que se cumplió el objetivo principal de este trabajo, determinar una función f tal que sirva como función predictora usando un conjunto de datos X y Y planteándolo como un problema de optimización. Para resolver este problema, utilizamos distintos algoritmos de optimización:
  - **LASSO gráfico** (2) (descenso de coordenadas por bloque y descenso coordinado)
  - Descenso de Coordenadas Cíclicas (3)

Conclusiones

- Primeramente podemos observar que se cumplió el objetivo principal de este trabajo, determinar una función f tal que sirva como función predictora usando un conjunto de datos X y Y planteándolo como un problema de optimización. Para resolver este problema, utilizamos distintos algoritmos de optimización:
  - **LASSO gráfico** (2) (descenso de coordenadas por bloque y descenso coordinado)
  - Descenso de Coordenadas Cíclicas (3)
  - Descenso de Coordenadas por Bloques (4)

- Por otro lado, planteamos dos metodología para resolver el problema de optimización de regresión multivariada con regularización.
- En generaI REMMAP parece desempeñarse mejor cuando $n < p$, mientras MRCE tiene un buen desempeño cuando $n > p$.
- MRCE presento la particularidad de requerir una parametrización mas cuidadosa por lo que en ese sentido es mas exigente, además de que involucra un costo computacional mas alto comparándolo con REMMAP.

Gracias ☻

Referencias

Friedman, J., Hastie, T. & Tibshirani, R. (2008). Sparse inverse covariance estimation with the graphical lasso. *Biostatistics*, *9*(3), 432-441. https://doi.org/10.1093/biostatistics/kxm045

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M. & Duchesnay, E. (2011). Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research*, *12*, 2825-2830.

Peng, J., Zhu, J., Bergamaschi, A., Han, W., Noh, D.-Y., Pollack, J. R. & Wang, P. (2010). Regularized multivariate regression for identifying master predictors with application to integrative genomics study of breast cancer. *The Annals*

of Applied Statistics, *4*(1), 53-77. https://doi.org/10.1214/09-AOAS271

Rothman, A. J., Levina, E. & Zhu, J. (2010). Sparse Multivariate Regression With Covariance Estimation [PMID: 24963268]. *Journal of Computational and Graphical Statistics*, *19*(4), 947-962. https://doi.org/10.1198/jcgs.2010.09188