

Tipo de artículo: Artículos originales

Temática: Reconocimiento de patrones

Recibido: 03/03/2020 | Aceptado: 12/05/2020 | Publicado: 02/06/2020

Un estudio empírico del modelo de red neuronal MLP para problemas de predicción con salidas múltiples.

An empirical evaluation of the MLP neural network model for multitarget regression.

Julio Camejo Corona^{1*}, Hector R. Gonzalez Diez², Carlos Morell³

¹Universidad de Cienfuegos Carlos Rafael Rodríguez. Facultad de Ingeniería, Departamento de informática. Km 4, Cuatro Caminos, Cienfuegos, Cuba.

²Universidad de las Ciencias Informáticas (UCI). Facultad 2, Centro CIGED. Km 2^{1/2} Autopista La Habana - San Antonio de los Baños La Habana, Cuba.

³Universidad Central Marta Abreu de las Villas. Centro de Estudio de Informática. Carretera Camajuaní, km 5. Santa Clara, Villa Clara, Cuba.

*Autor para correspondencia: jcamejo@ucf.edu.cu

Resumen

Existe una amplia gama de problemas en los cual se desean estimar de manera simultanea más de una variable asociada, de ahí que la predicción con salidas múltiples ha tenido un punto de atención por la comunidad científica. El algoritmo de predicción con salidas múltiples GMLR, es una generalización a otros propuestos en la literatura y además, el empleo del método del Gradiente Proximal Acelerado en un esquema alternado, junto a una función objetivo bi-convexa permite obtener resultados prominentes a la vez que los supera en eficiencia. Sin embargo, un esquema equivalente a este modelo generalizado, son las redes neuronales artificiales tales como el perceptron multicapa MLP si se considera una arquitectura en la cual se tiene una neurona por cada variable de salida en la última capa de la red. En la presente investigación hemos considerado comparar los resultados obtenidos en términos de eficacia del modelo GMLR respecto a la red neuronal MLP mediante un estudio empírico de ambos algoritmos en en el contexto de la predicción con salidas múltiples.

Palabras claves: Regresión con múltiples salidas, Regresión, GMLR, MLP, Optimización

Abstract

There is a wide range of problems in which it is desired to estimate more than one associated variable simultaneously, hence the multitarget regression has had a point of attention by the scientific community. The algorithm of prediction with multiple outputs variables, GMLR is a generalization to others proposed in the literature and also, the use of the Accelerated Proximal Gradient method in an alternate scheme, together with A bi-convex objective function enables prominent results to be obtained while it's better of them in efficiency. However, an equivalent scheme to this generalized model is the artificial neural networks such as the MLP multilayer perceptron. The equivalence are to in MLP where are consider in the output layer one neuron per each output variable. In the current research, we have considered comparing the experimental results in terms of efficacy of the GMLR model respect to the MLP neural network by means of an empirical study of both algorithms in the

context of Multitarget Regression.

Keywords: *Multi-target regression, Regression, GMLR, MLP, Optimization*

1 Introducción

Diversos problemas en Reconocimiento de Patrones y Aprendizaje Automático han permitido la evolución de estas disciplinas a nuevos enfoques no convencionales de predicción o clasificación. Entre los métodos no convencionales que mayor desarrollo han tenido en los últimos años se encuentra la Predicción Estructurada [Bakir et al. \(2007\)](#), cuya diferencia con los métodos convencionales radica en que la variable de salida es modelada como una estructura de datos compleja. Estas estructuras, para la variable de salida, pueden ser modeladas como grafos, jerarquías, secuencias, cadenas de texto o vectores, dependiendo del tipo de problema que se desea estudiar. En particular, aquellos problemas que se componen por vectores en la variable de salida son conocidos como problemas de Clasificación Multi-etiqueta (cuando son vectores con valores binarios) o Predicción con Salidas Múltiples (cuando las salidas están formadas por vectores reales) [Borchani et al. \(2015\)](#).

Múltiples aplicaciones han sido estudiadas en el ámbito de la Predicción con Salidas Múltiples: en el campo de la quimiometría para evaluar diversos parámetros de calidad de agua [Džeroski et al. \(2000\)](#), en el modelado de los componentes de sistemas ecológicos [Kocev et al. \(2009\)](#) o para la venta de pasajes de cierta aerolínea [Spyromitros-Xioufis et al. \(2012\)](#).

Un artículo de revisión publicado recientemente sobre los problemas de Predicción con Salidas Múltiples establece dos grandes categorías [Borchani et al. \(2015\)](#) para este tipo de tarea: el primer enfoque contiene los métodos que transforman los problemas de salidas múltiples en diferentes problemas de predicción de una salida; mientras que el otro enfoque se basa en adaptar métodos clásicos conocidos de Aprendizaje Automático para estimar de manera simultánea las múltiples salidas. En ambas categorías los métodos pueden tener en cuenta o no la interdependencia entre las variables de salida, aunque de manera natural el segundo enfoque establece esta relación de forma directa.

En el caso de los algoritmos basados en transformación, se aprenden q modelos de regresión (uno por cada salida lo cual es conocido también como un problema de Relevancia Binaria BR). La principal desventaja de este enfoque es que ignora la interdependencia entre las variables de salida. En [Spyromitros-Xioufis et al. \(2012\)](#), se extienden enfoques clásicos de Clasificación Multi-etiqueta conocidos como *Stacking* y *Regressor Chain* para proponer los algoritmos *MTS* y *ERC* y sus variantes corregidas *MTSC* y *ERCC*, que constituyen el estado del arte de los algoritmos de Predicción con Salidas Múltiples basados en transformación.

Los primeros trabajos que abordaron el problema de múltiples salidas provienen del campo del aprendizaje estadístico. Los enfoques más simples en este campo están basados en regresión lineal haciendo uso de regularizadores de tipo ℓ_1 o ℓ_2 . Estos enfoques predicen cada salida de manera independiente a través de una función lineal de regresión. En el ámbito de la estadística multivariada dos algoritmos representativos, que explotan la interdependencia entre las variables de salida, son el *Curds and Wheys (CW)* Breiman and Friedman (1997) y el *Reduced Rank Regression (RR)* Izenman (1975). En el caso del *CW* se emplean las regresiones lineales de cada variable de salida independiente, para luego ser ajustada aprendiendo una combinación lineal de las variables de salida a través de la correlación canónica Hotelling (1936). Por otra parte, el *RR* hace uso de la reducción de rango en la descomposición de la matriz de co-varianza formada entre las variables de entrada y salida.

En los algoritmos anteriormente descritos, han sido empleados como regresores base, métodos provenientes del Aprendizaje Automático convencional tales como: *SVR*, *Random Forest*, *Bagging* y *Ridge Regression* entre otros Smola and Vapnik (1997); Breiman (2001, 1996); Hoerl and Kennard (1970). Una ausencia notable entre los métodos que han sido empleados, lo constituye el algoritmo basado en la regla de los k -Vecinos más Cercanos (k -NN) Cover and Hart (1967), a pesar de su simplicidad y posibilidades de escalabilidad. Este método ha demostrado ser competitivo en la modelación de problemas convencionales de clasificación y regresión para diversos dominios de aplicación Chen et al. (2016); James et al. (2013); Hyndman and Athanasopoulos (2018); Pedregosa et al. (2011). En la literatura consultada sobre Predicción con Salidas Múltiples sólo se encontró reportado el trabajo Pugelj and Džeroski (2011), que propone el algoritmo KNN-SP para tres tipos de tareas de Predicción Estructurada. La propuesta de este algoritmo no toma en cuenta la interdependencia entre las variables de salida en ninguna fase de su funcionamiento. Una experimentación extendida del KNN-SP propuesta en González et al. (2016), demuestra que existen oportunidades de mejorar su capacidad predictiva a través de diversas estrategias de aprendizaje.

Los algoritmos basados en adaptación consideran un modelo único capaz de predecir de manera simultánea el conjunto de variables de salida sin modificar las condiciones del problema. Los primeros trabajos en este campo provienen del aprendizaje estadístico o estadística multivariada como es el caso de la regresión por contracción (*Ridge Regression*) Hoerl and Kennard (1970). De igual manera, en Izenman (1975) se aprende una estructura de bajo rango en la propuesta de algoritmo *Reduced Rank Regression* que usa un modelo de regresión lineal.

Breiman y Friedman en su trabajo Breiman and Friedman (1997) proponen la contracción simultánea tanto en el espacio de entrada como en el de salida. En general, para q variables de respuesta $\mathbf{y} = (y_1, \dots, y_q)$ con una predicción basada en regresión lineal simple para cada variable de salida, se obtienen $\hat{\mathbf{y}} = (\hat{y}_1, \dots, \hat{y}_q)$. Si las variables de salida se encuentran correlacionadas es posible obtener un predictor más preciso, denotado como \tilde{y}_i para cada variable de salida, usando una combinación lineal.

El algoritmo C&W [Breiman and Friedman \(1997\)](#) estableció un hito importante en los modelos de predicción sobre múltiples conjuntos de datos ya que a través de la correlación canónica modela la relación intrínseca en el conjunto de variables de salida. Otro enfoque que explota la correlación canónica es el algoritmo denominado FICYREG propuesto en [Van Der Merwe and Zidek \(1980\)](#).

Otros modelos de MTR son los conocidos como FES [Yuan and Lin \(2006\)](#); [Yang et al. \(2013\)](#), que emplean un esquema de regularización basado en la norma ℓ_{21} [Simon et al. \(2013b\)](#), el cual permite de manera simultánea realizar selección de atributos. En [Yuan and Lin \(2006\)](#), se comparan diferentes soluciones al problema de LASSO o lo que es equivalente, el uso de la norma ℓ_1 como regularizador. Por otra parte, el algoritmo MRCE es propuesto en [Rothman et al. \(2010\)](#). El mismo obtiene estructuras de bajo rango para estimar la matriz de covarianza de los datos. A partir de los algoritmos anteriormente mencionados, se comienzan a generalizar los esquemas de regularización con estructuras dispersas y de bajo rango entre los que resaltan los trabajos, introducidos inicialmente por Hastie en [Friedman et al. \(2010\)](#) y extendidos en los últimos años [Vincent and Hansen \(2014\)](#); [Bak \(2017\)](#); [Münch et al. \(2018\)](#), con el uso de funciones de regularización convenientes para obtener dichas estructuras. Para el caso particular de los problemas con múltiples salidas, la regularización basada en la norma ℓ_{21} combinada con el método de coordenada descendente fue empleada en [Simon et al. \(2013a\)](#).

2 Materiales y métodos o Metodología computacional

De manera reciente en [Diez et al. \(2018\)](#) se resuelve el problema de regresión lineal multivariada de manera más eficiente, introduciendo un espacio de variables latentes o un aprendizaje en capas. Esta propuesta, llamada *Generalized Multitarget Linear Regression with Output Dependence Estimation* (GMLR) constituye un esquema general, flexible y adaptable a diferentes escenarios y esquemas de regularización. Este se basa en la proyección de las variables a un espacio de variables latentes y el uso de un método flexible y escalable de optimización mediante la modelación de un problema Biconvexo soluble de manera alternada. Su principal ventaja es la escalabilidad para considerar disímiles situaciones aprovechando las bondades del método de optimización gradiente proximal acelerado.

Por otro lado, las Redes Neuronales Artificiales (Artificial Neural Network; ANN) han demostrado ser muy efectivas para tareas de aprendizaje supervisado y no supervisado debido a sus propiedades no lineales de aprendizaje adaptativo no paramétrico. Como una herramienta analítica útil, son ampliamente aplicadas en el análisis de los datos almacenados en bases de datos o almacenes de datos hoy en día. Una arquitectura de ANN, muy reconocida por sus buenos resultados y considerado un regresor universal cuando se predice con el mismo conjunto de datos con los que fue entrenado, es el Perceptron Multicapa (Multi Layer Perceptron; MLP)

2.1 GMLR

Según la definición de GMLR, sean $\mathbf{x} \in \mathbb{R}^p$ y $\mathbf{y} \in \mathbb{R}^q$ dos vectores aleatorios definidos en los espacios de entrada y salida, respectivamente, de un determinado problema de Aprendizaje Automático con múltiples variables de salida reales de dimensión q . Para el conjunto de datos, con N instancias de entrenamiento $\mathcal{D} = \{(\mathbf{x}_1, \mathbf{y}_1), \dots, (\mathbf{x}_N, \mathbf{y}_N)\}$, es posible definir un modelo de regresión con múltiples salidas que tome en cuenta las relaciones entre las variables de entrada y salida y en el conjunto de variables de salida, siguiendo un modelo de regresión lineal cuya función de predicción es $\mathbf{y} = \mathbf{x}WS$. Este modelo define un conjunto de variables latentes $\mathbf{z} = \mathbf{x}W \in \mathbb{R}^r$ entre el conjunto de variables de entrada y salida. Las variables latentes se obtienen a partir de la matriz $W \in \mathbb{R}^{p \times r}$ que establece las relaciones entre el conjunto de variables de entrada y las variables latentes. En el caso en $r = q$ se dice que W es una estructura de rango completo. Luego, para relacionar el conjunto de variables de salida a través de las variables latentes se incluye en el modelo la estructura matricial $S \in \mathbb{R}^{r \times q}$ que combina linealmente el conjunto de predicciones individuales en el modelo de regresión lineal.

La figura 1 ilustra las diferentes estructuras del modelo.

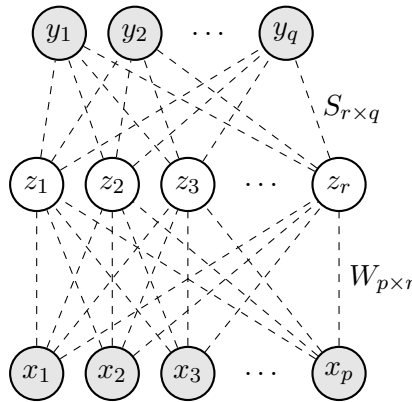


Figura 1. Representación gráfica de las estructuras que intervienen en el modelo generalizado.

En términos generales, GMLR para estimar las estructuras que relacionan el conjunto de variables de entrada y salida y entre variables de salida define un problema de optimización para minimizar un funcional de pérdida $\ell(\mathcal{D}, W, S)$ y una función de regularización $\mathcal{R}(W, S)$ que controla la generalización del modelo y al mismo tiempo establece las estructuras de cada matriz.

$$(W^*, S^*) = \underset{W, S}{\operatorname{argmin}} \ell(\mathcal{D}, W, S) + \lambda \mathcal{R}(W, S) \quad (1)$$

Quedando expresadas, para una matriz de datos $X_{N \times p}$ y $Y_{N \times q}$, la matrices de dependencia entre entrada y salida $W_{p \times r}$, y entre variables de salidas $S_{r \times q}$ cuyas dimensiones han sido denotadas como sub-indices. Luego el problema general de GMLR se expresa como:

$$(W^*, S^*) = \underset{W, S}{\operatorname{argmin}} \|Y - XWS\|_F^2 + \lambda_1 g(W) + \lambda_2 h(S) \quad (2)$$

Para resolver este problema en GMLR se fija una de las matrices y actualiza la otra, de manera alternada (Problema biconvexo).

$$W_{t+1} = \underset{W}{\operatorname{argmin}} \frac{1}{2N} \|Y - XW_t S_t\|_F^2 + \lambda_1 g(W_t) \quad \text{Fijando } S_t \quad (3)$$

$$S_{t+1} = \underset{S}{\operatorname{argmin}} \frac{1}{2N} \|Y - XW_{t+1} S_t\|_F^2 + \lambda_2 h(S_t) \quad \text{Fijando } W_{t+1} \quad (4)$$

mediante los operadores proximales definidos en [Diez et al. \(2018\)](#):

$$W_{t+1} := \underset{\lambda_1 s_t g_1}{\operatorname{prox}}(W_t - s_t \nabla_W \ell(W_t, S_t)) \quad (5)$$

$$S_{t+1} := \underset{\lambda_2 s_t g_2}{\operatorname{prox}}(S_t - s_t \nabla_S \ell(W_{t+1}, S_t)) \quad (6)$$

donde

$$\nabla_W \ell = X(X^T W_t S_t - Y) S_t^T \quad (7)$$

$$\nabla_S \ell = W_{t+1}^T X(X^T W_{t+1} S_t - Y) \quad (8)$$

2.2 Perceptron Multi-Capa.

El MLP es una generalización del Perceptron Simple y surgió como consecuencia de las limitaciones de dicha arquitectura en lo referente al problema de la separabilidad no lineal. Diferentes autores han demostrado independientemente que este es un aproximador universal, en el sentido de que cualquier función continua en un espacio \mathbb{R}^n puede aproximarse con un MLP, con al menos una capa oculta de neuronas.

La arquitectura de MLP se caracteriza porque tiene sus neuronas agrupadas en capas de diferentes niveles. Cada una de las capas está formada por un conjunto de neuronas y se distinguen tres tipos de capas diferentes: la capa de entrada, las capas ocultas y la capa de salida.

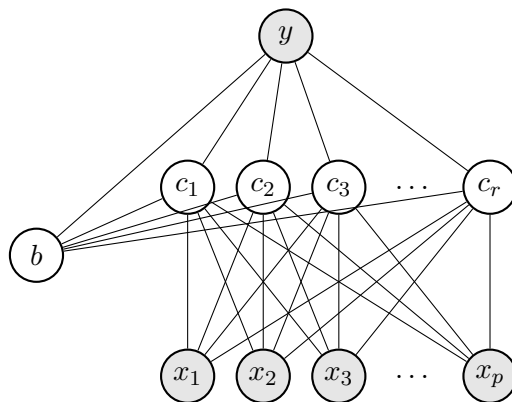


Figura 2. Esquema de un MLP con una capa de neuronas ocultas

Las neuronas de la capa de entrada no actúan como neuronas propiamente dichas, sino que se encargan únicamente de recibir las señales o patrones del exterior y propagar las a todas las neuronas de la siguiente capa. La última capa actúa como salida de la red, proporcionando al exterior la respuesta de la red para cada uno de los patrones de entrada. Las neuronas de las capas ocultas realizan un procesamiento no lineal de los patrones recibidos.

Como se observa en la figura 2, las conexiones del MLP siempre están dirigidas hacia adelante, es decir, las neuronas de una capa se conectan con las neuronas de la siguiente capa, de ahí que reciban también el nombre de redes alimentadas hacia adelante o redes feedforward. Las conexiones entre las neuronas de la red llevan también asociado un umbral, que el caso del MLP suele tratarse como una conexión más a la neurona, cuya entrada es constante e igual a 1.

Generalmente, todas las neuronas de una capa están conectadas a todas las neuronas de la siguiente capa. Se dice entonces que existe conectividad total o que la red está totalmente conectada.

Aunque en la mayor parte de los casos la arquitectura del Perceptrón multicapa está asociada al esquema de la figura 2, es posible también englobar dentro de este tipo de redes a arquitecturas con las siguientes características:

- Redes con conexiones de todas o ciertas neuronas de una determinada capa a neuronas de capas poste-

rios, aunque no inmediatamente posteriores.

- Redes en las que ciertas neuronas de ciertas capas no están conectadas a neuronas de la siguiente capa, es decir, el peso de la conexión es constante e igual a cero.

En la mayoría de las aplicaciones se parte de una arquitectura totalmente conectada, es decir, todas las neuronas de una capa están conectadas a todas las neuronas de la siguiente capa. Sin embargo, no es posible demostrar que si se utilizan arquitecturas en las que se eliminan o se añaden conexiones de una capa a capas no inmediatamente posteriores, se puedan obtener mejores resultados. El MLP es considerado un regresor universal cuando se predice con el mismo conjunto de datos con los que fue entrenado.

3 Comparativa del modelo GMLR con una red neuronal tipo MLP

En esta sección hemos considerado comparar los resultados obtenidos del modelo GMLR respecto a la red neuronal MLP. Hemos utilizado la implementación del módulo para redes neuronales disponible en MATLAB. En la implementación configuramos el MLP con una capa oculta equivalente la capa de variables latentes del GMLR. Se utilizó un conjunto de 16 bases de datos públicas disponibles para MTR consideradas en la experimentación [Spyromitros-Xioufis et al. \(2016\)](#). Dependiendo de los atributos de entrada y salida de cada base de datos, hemos configurado la cantidad de neuronas de las capas de entrada y salida, mientras que la capa oculta hemos usado la misma cantidad de neuronas que en la capa de salida. La figura 3 muestra un ejemplo de la arquitectura de red empleada en la experimentación para la base de datos edm. Esta base de datos posee 16 atributos predictores y 2 variables de salida. Se empleó como algoritmo de aprendizaje *Levenberg-Marquardt backpropagation* [Sapna et al. \(2012\)](#) y se dividieron internamente los datos en entrenamiento y validación con vista a lograr una mayor capacidad de generalización de la red. El proceso de entrenamiento en todos los casos no excedió los 20 ciclos de entrenamientos pues se logra la convergencia en todos los casos. Se selecciona como medida interna de validación cruzada el MSE el cual permitió determinar la mejor configuración en los ajustes de los pesos de la red.

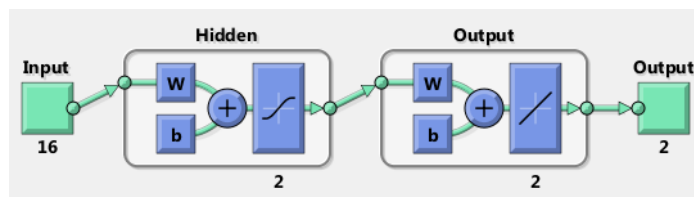


Figura 3. Arquitectura del MLP empleado en la experimentación. Ejemplo del dataset EDM

Los resultados de la medida aRRMSE fueron reportados en la tabla 1 aplicado sobre el conjunto de datos de entrenamiento y prueba para el MLP y los casos particulares GMLR1 y GMLR2. Como se puede apreciar existen bases de datos, tales como wq, rf1, rf2, sf1 y sf2 y enb, donde el MLP tiene un comportamiento adecuado asociado con la medida de error que se emplea en la experimentación. Estos resultados se corresponden con el hecho de que el MLP es un buen regresor cuando predice con los propios datos con los que fue entrenado Mehrotra et al. (1997); Kay et al. (1999). Sin embargo, si observamos los resultados del MLP sobre el conjunto de datos de prueba, se aprecia que los resultados son muy distantes de los alcanzados por el GMLR a excepción de bases de datos muy puntuales como el rf1 y rf2.

Tabla 1. Estudio comparativo del MLP respecto al GMLR

Algoritmos Dataset	MLP		GLMR ₁		GLMR ₂	
	train_error	test_error	train_error	test_error	train_error	test_error
edm	0.761	0.856	0.713	0.792	0.728	0.793
wq	0.922	0.959	0.939	0.949	0.934	0.949
sf1	0.982	0.997	0.950	0.965	0.952	0.966
sf2	0.949	1.124	0.953	0.967	0.942	0.977
oes10	1.135	1.566	0.296	0.339	0.210	0.308
oes97	0.657	1.449	0.397	0.582	0.243	0.519
andro	1.078	1.771	0.580	0.696	0.646	0.662
atp1d	0.555	0.861	0.422	0.558	0.342	0.504
atp7d	1.021	1.151	0.535	0.720	0.523	0.708
enb	0.335	0.328	0.323	0.299	0.318	0.296
jura	0.777	0.775	0.580	0.599	0.586	0.593
slump	0.694	1.112	0.633	0.701	0.589	0.709
rf1	0.252	0.330	0.303	0.407	0.302	0.406
rf2	0.286	0.343	0.445	0.515	0.244	0.359

La causa esencial de este resultado lo constituye el uso de regularizadores en el GMLR, lo cual permitió obtener un modelo que garantiza generalización en contraposición con el algoritmo MLP. Nótese, además, que existe una marcada diferencia entre los resultados del MLP sobre los conjuntos de datos en entrenamiento y prueba no siendo así para los modelos GMLR evaluados.

Con vista a mostrar el comportamiento del MLP con diferentes modelos de optimización se realizó un estudio de diferentes solucionadores además del ya empleado Levenberg-Marquardt backpropagation. Para ello se emplearon las siguientes configuraciones:

- Se emplearon los siguientes esquemas de aprendizaje.
 - cgf. Conjugate gradient backpropagation with Fletcher-Reeves updates.
 - cgp. Conjugate gradient backpropagation con Polak-Ribière updates.
 - gda. Gradient descent with adaptive learning rate backpropagation.

- gdm. Gradient descent with momentum backpropagation.
 - gdx Gradient descent with momentum and adaptive learning rate backpropagation.
 - nlm Levenberg-Marquardt backpropagation
 - noss One-step secant backpropagation
 - nrp Resilient backpropagation
 - scg Scaled conjugate gradient backpropagation
- Se diseñó una capa de entrada con p variables predictoras, una capa oculta con q neuronas, capa de salida con q neuronas equivalentes al número de salidas. Poner más capas permite modelar la no linealidad pero el modelo difiere del GMLR y por tanto la comparación no es justa.
 - Se ejecutaron 10 corridas del procedimiento validación cruzada para generar los conjuntos de entrenamiento, validación y prueba.
 - La función de transferencia fue la lineal.
 - Se reportan como medidas de error de la red el MSE y aRRMSE para poder comparar con el GMLR.

Los resultados alcanzados para las 6 bases de datos estudiadas se muestran en la figura 4. EN la misma se muestra muy poca dispersión en el conjunto de test lo cual denota que no existen diferencias entre los algoritmos estudiados.

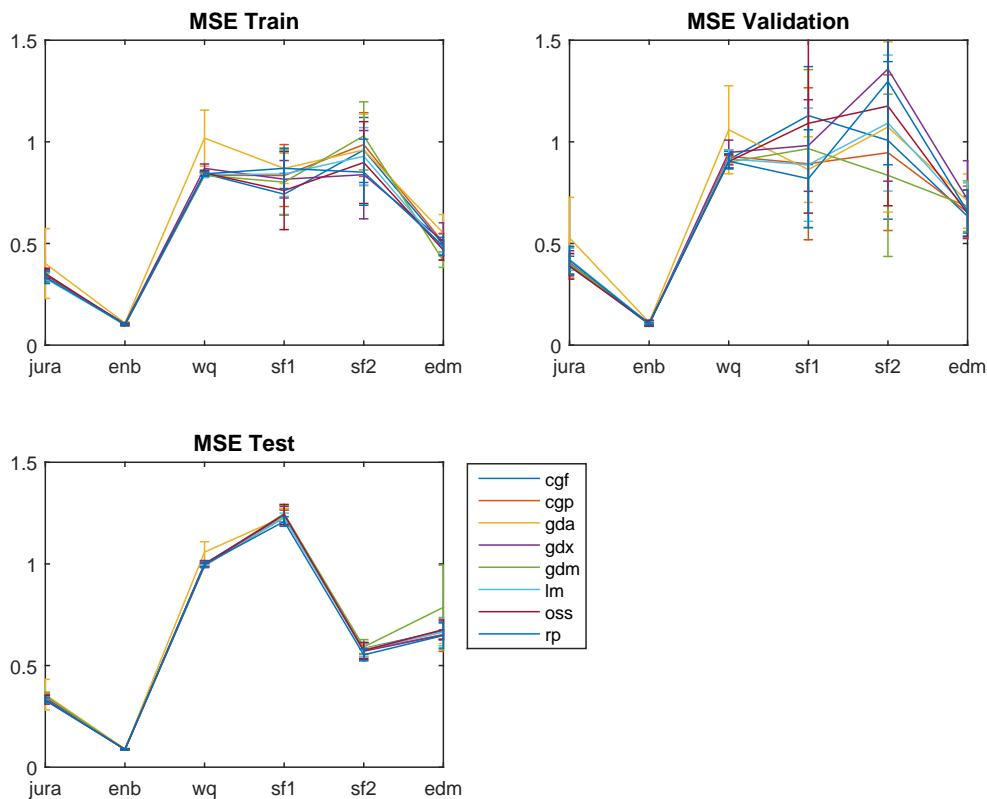


Figura 4. Resultados de los experimentos con diferentes solucionadores.

4 Conclusiones

Por la similitud del modelo GMLR con los esquemas de redes neuronales tipo MLP, se compararon los resultados en cuanto a eficacia entre estos dos modelos, lográndose mayor capacidad de generalización producto del esquema de regularización de LASSO agrupado propuesto. El algoritmo GMLR, obtiene resultados superiores, en cuanto a eficacia, respecto al modelo de red neuronal MLP. Este estudio sugiere una ampliación a nuevas arquitecturas de redes neuronales profundas con mecanismos de regularización equivalentes a los empleados en el GMLR.

Bibliografía

- Stephen Bak. *Generalized linear regression model with LASSO, group LASSO, and sparse group LASSO regularization methods for finding bacteria associated with colorectal cancer using microbiome data*. PhD thesis, 2017.
- Gükhan H. Bakir, Thomas Hofmann, Bernhard Schölkopf, Alexander J. Smola, Ben Taskar, and S. V. N. Vishwanathan. *Predicting Structured Data (Neural Information Processing)*. The MIT Press, 2007. ISBN 0262026171.
- Hanen Borchani, Gherardo Varando, Concha Bielza, and Pedro Larrañaga. A survey on multi-output regression. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 5(5):216–233, 2015.
- Leo Breiman. Bagging predictors. *Machine learning*, 24(2):123–140, 1996.
- Leo Breiman. Random forests. *Machine learning*, 45(1):5–32, 2001.
- Leo Breiman and Jerome H Friedman. Predicting multivariate responses in multiple linear regression. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 59(1):3–54, 1997.
- Mei Chen, Longjie Li, Bo Wang, Jianjun Cheng, Lina Pan, and Xiaoyun Chen. Effectively clustering by finding density backbone based-on knn. *Pattern Recognition*, 60:486–498, 2016.
- Thomas Cover and Peter Hart. Nearest neighbor pattern classification. *Information Theory, IEEE Transactions on*, 13(1):21–27, 1967.
- Hector Raúl González Díez, Carlos A. Morell Pérez, and Francesc J. Ferri. *Extensión del aprendizaje basado en instancia para problemas de predicción con salidas múltiples*. Thesis, 2018.
- Sašo Džeroski, Damjan Demšar, and Jasna Grbović. Predicting chemical parameters of river water quality from bioindicator data. *Applied Intelligence*, 13(1):7–17, 2000.
- Jerome Friedman, Trevor Hastie, and Rob Tibshirani. Regularization paths for generalized linear models via coordinate descent. *Journal of statistical software*, 33(1):1, 2010.
- Héctor González, Gabriela Santos, Frank Campos, and Carlos Morell Pérez. Evaluación del algoritmo knn-sp para problemas de predicción con salidas compuestas. *Revista Cubana de Ciencias Informáticas*, 10(3): 119–129, 2016.
- Arthur E Hoerl and Robert W Kennard. Ridge regression: Biased estimation for nonorthogonal problems. *Technometrics*, 12(1):55–67, 1970.

- Harold Hotelling. Relations between two sets of variates. *Biometrika*, pages 321–377, 1936.
- Rob J Hyndman and George Athanasopoulos. *Forecasting: principles and practice*. OTexts, 2018.
- Alan Julian Izenman. Reduced-rank regression for the multivariate linear model. *Journal of multivariate analysis*, 5(2):248–264, 1975.
- Gareth James, Daniela Witten, Trevor Hastie, and Robert Tibshirani. *An introduction to statistical learning*, volume 112. Springer, 2013.
- Jim W Kay, D Mike Titterington, et al. *Statistics and neural networks: advances at the interface*. Oxford University Press on Demand, 1999.
- Dragi Kocev, Sašo Džeroski, Matt D White, Graeme R Newell, and Peter Griffioen. Using single-and multi-target regression trees and ensembles to model a compound index of vegetation condition. *Ecological Modelling*, 220(8):1159–1168, 2009.
- Kishan Mehrotra, Chilukuri K Mohan, and Sanjay Ranka. *Elements of artificial neural networks*. MIT press, 1997.
- Magnus M Münch, Carel FW Peeters, Aad W van der Vaart, and Mark A van de Wiel. Adaptive group-regularized logistic elastic net regression. *arXiv preprint arXiv:1805.00389*, 2018.
- Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, et al. Scikit-learn: Machine learning in python. *Journal of machine learning research*, 12(Oct):2825–2830, 2011.
- Mitja Pugalj and Sašo Džeroski. Predicting structured outputs k-nearest neighbours method. In *Discovery Science*, pages 262–276. Springer, 2011.
- Adam J Rothman, Elizaveta Levina, and Ji Zhu. Sparse multivariate regression with covariance estimation. *Journal of Computational and Graphical Statistics*, 19(4):947–962, 2010.
- S Sapna, A Tamilarasi, M Pravin Kumar, et al. Backpropagation learning algorithm based on levenberg marquardt algorithm. *Comp Sci Inform Technol (CS and IT)*, 2:393–398, 2012.
- Noah Simon, Jerome Friedman, and Trevor Hastie. A blockwise descent algorithm for group-penalized multi-response and multinomial regression. *arXiv preprint arXiv:1311.6529*, 2013a.
- Noah Simon, Jerome Friedman, Trevor Hastie, and Robert Tibshirani. A sparse-group lasso. *Journal of Computational and Graphical Statistics*, 22(2):231–245, 2013b.

- Alex Smola and Vladimir Vapnik. Support vector regression machines. *Advances in neural information processing systems*, 9:155–161, 1997.
- Eleftherios Spyromitros-Xioufis, Grigorios Tsoumakas, William Groves, and Ioannis Vlahavas. Multi-label classification methods for multi-target regression. *arXiv preprint arXiv:1211.6581*, 2012.
- Eleftherios Spyromitros-Xioufis, Grigorios Tsoumakas, William Groves, and Ioannis Vlahavas. Multi-target regression via input space expansion: treating targets as inputs. *Machine Learning*, 104(1):55–98, 2016.
- A Van Der Merwe and JV Zidek. Multivariate regression analysis and canonical variates. *Canadian Journal of Statistics*, 8(1):27–39, 1980.
- Martin Vincent and Niels Richard Hansen. Sparse group lasso and high dimensional multinomial classification. *Computational Statistics & Data Analysis*, 71:771–786, 2014.
- Wei Yang, Kuanquan Wang, and Wangmeng Zuo. Prediction of protein secondary structure using large margin nearest neighbour classification. *International journal of bioinformatics research and applications*, 9(2):207–219, 2013.
- Ming Yuan and Yi Lin. Model selection and estimation in regression with grouped variables. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 68(1):49–67, 2006.