Revista Cubana de Ciencias Informáticas
Vol. 13, No. 4, Mes Octubre-Diciembre, 2019
ISSN: 2227-1899 | RNPS: 2301
http://rcci.uci.cu
Pág. 118-150

Tipo de artículo: Artículo de revisión
Temática: Reconocimiento de patrones
Recibido: 20/08/2019 | Aceptado: 11/12/2019

Los principales algoritmos para regresión con salidas múltiples. Una revisión para Big Data

Multitarget Regression Problem. A review for Big Data

Julio Camejo Corona [1*], Héctor González Diez [2], Carlos Morell [3]

[1] Universidad de Cienfuegos Carlos Rafael Rodríguez. Carretera a Rodas. Km. 4. Cienfuegos, Cuba.
[2] Universidad de las Ciencias Informáticas (UCI). Km 21/2 Autopista La Habana - San Antonio de los Baños, La Habana, Cuba.
[3] Universidad Central Marta Abreu de las Villas. Carretera Camajuaní, km 5. Santa Clara, Villa Clara, Cuba.

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

Resumen
En muchas ocasiones se presentan problemas de regresión donde se desea estimar de manera simultánea más de un rasgo o variable real. En estos casos se pueden modelar tantos regresores como variables de salidas existan, lo cual desestima la dependencia condicional entre los pares variable de salida considerando cada problema independiente. Recientemente se ha demostrado que considerar esta dependencia mejora la capacidad predictiva de los modelos de aprendizaje. El elevado costo computacional de estos algoritmos, y la enorme cantidad de información almacenada en millones de bases de datos, ha traído consigo tiempos de procesamiento excesivamente grandes en la generación de estos modelos. Este hecho nos conlleva a abordar estos problemas desde un enfoque de Big Data. El objetivo de este artículo es ofrecer una panorámica sobre el estado actual de las principales propuestas de regresión con salidas múltiples y sus posibilidades de ser reformulados para enfrentar el trabajo en problemas con grandes volúmenes de datos. Además, se aborda la metodología seguida por la Regresión Linear Múltiple ya implementada en la plataforma Apache Spark que sentará las bases para definir nuevos modelos en este contexto. Finalmente, se exponen los principales métodos de optimización que emplean estos métodos y sus variantes desde Big Data.

Palabras clave: Regresión con múltiples salidas, Regresión, Apache Spark, Big Data, Grandes volúmenes de datos, Optimización

Abstract
In many cases regression problems with more than one objective feature can be present. In these cases, you can model as many regressors as output variables exist, which underestimates the conditional dependence between the variable output pairs considering each independent problem. Recently it has been shown that considering this dependency

Revista Cubana de Ciencias Informáticas
Vol. 13, No. 4, Mes Octubre-Diciembre, 2019
ISSN: 2227-1899 | RNPS: 2301
http://rcci.uci.cu
Pág. 118-150

produces better results since in many problems the output variables yield results that are related to each other. The high computational cost of these algorithms, and the enormous amount of information stored in millions of databases, has resulted in excessively large processing times in the generation of these models, which implies the need to manage these problems from Big Data concept. The objective of this article is to provide an overview of the current state of the main regression proposals with multiple outputs and their possibilities of being reformulated to Large-Scale problems. Besides, the followed methodology by the Multiple Linear Regression already implemented in the Apache Spark platform is addressed. Finally, the main optimization techniques that use these methods and their variants from Big Data are exposed.

Keywords: *Multi-target regression, Regression, Apache Spark, Big Data, Large-Scale, Optimization*

Introducción

El aprendizaje supervisado tradicional, como uno de los paradigmas de aprendizaje automático más adoptados en sistemas y aplicaciones inteligentes del mundo real, favorece la toma de decisiones rápidas y precisas para las tareas de regresión y predicción. El aprendizaje supervisado se caracteriza por utilizar ejemplos del dominio de aplicación, previamente etiquetados, y con ello aprender modelos que permitan estimar nuevos ejemplos no empleados en el conjunto de entrenamiento. El Aprendizaje supervisado estándar asume que los ejemplos son autocontenidos, esto es, que toda la información necesaria para hacer una predicción está contenida en la descripción del objeto. La descripción del objeto, en general, está dada en forma de pares ordenados (atributo, valor) donde los valores pueden ser numéricos, categóricos o una mezcla de ellos. La naturaleza de la etiqueta asignada a un ejemplo tiene grandes implicaciones en los algoritmos supervisados.

Dependiendo de la naturaleza de la variable objetivo, los problemas de aprendizaje pueden ser clasificados en predicción, para valores reales, o clasificación, para datos categóricos. Si la etiqueta puede tomar valores continuos entonces la tarea de predicción se conoce como Regresión. En ambos casos, el algoritmo de aprendizaje debe sintetizar un modelo que sea capaz de predecir la etiqueta de un ejemplo desconocido a partir de sus variables predictoras James et al. (2013). Entre las áreas de aplicación de los modelos de regresión podemos mencionar los siguientes ejemplos: viscosidad de crudos a partir de los niveles de temperatura, gravedad API y el porcentaje de asfaltenos Velásquez (2017). En educación se analiza el rendimiento académico en la disciplina de matemática a partir de calificaciones obtenidas en matemática en los últimos 6 años Jeylin Meybelin Pérez Obregón and Díaz (2018). En economía para predecir la probabilidad de riesgo de quiebra en función del tiempo a partir de razones financieras CASTRO et al. (2019). En la predicción de inundaciones fluviales en un núcleo costero a partir de los hidrogramas y el nivel de marea prescritos como variables predictoras Bermúdez et al. (2017). En la predicción de gravedad de accidentes de tránsito: a

partir de la información del lugar del accidente (dirección, datos de georreferencia y localidad), clase de accidente (atropello, choque, caída de ocupante, incendio, volcamiento, autolesión u otro), fecha y hora de ocurrencia, condición de la víctima (peatón, pasajero, motociclista, ciclista o conductor), edad y género de la víctima Sonia E. Monroy Varela (2018).

En ocasiones se presentan problemas que involucran predicciones con más de una variable en estudio, normalmente representadas mediante varias etiquetas relacionadas entre sí, en forma de un vector. Por ejemplo, En la visión por computadora con frecuencia se necesita asociar una imagen a varias categorías de manera simultánea, o incluso requerir una lista clasificada de anotaciones Bucak et al. (2009). En el procesamiento del lenguaje natural, hay que traducir oraciones de un lenguaje específico a otro, donde cada oración es una secuencia de palabras Koehn (2005). En la modelación de la relación entre las variables de temperatura, luz, pH y oxígeno disuelto, y el crecimiento del cultivo de microalgas Carrasquilla-Batista et al. (2016). En bioinformática, se analiza la secuencia de proteínas (estructura primaria) para predecir las estructuras de las proteínas, incluidos los elementos de estructura secundaria, la disposición de los elementos y la asociación de cadenas Liu et al. (2005). En metalurgia se estiman varios factores de flotación a partir de la ubicación espacial de los datos Gonzales (2018).

Estos problemas que involucran predicciones simultaneas de un vector de números reales suelen ser tratados en el contexto del aprendizaje de salidas múltiples (*Multi-Target Prediction*; MTP) Xu et al. (2019). Este nuevo paradigma de aprendizaje automático emergente, que apunta a predecir simultáneamente múltiples salidas a partir de un conjunto de variables de entrada a tenido un popular auge en los últimos años Borchani et al. (2015); Zhen et al. (2018a,b); Spyromitros-Xioufis et al. (2016). En comparación con el aprendizaje tradicional de salida única, tiene una naturaleza multivariable y las salidas múltiples pueden tener interacciones complejas que solo pueden manejarse mediante una inferencia bien estructurada. Los valores de salida tienen diversos tipos de datos en diversos problemas de aprendizaje automático Waegeman et al. (2019). Por ejemplo, los valores de salida binarios pueden referirse al problema de clasificación de múltiples etiquetas Zhang and Zhou (2014), valores de salida nominales a problema de clasificación multidimensional Bielza et al. (2011), valores de salida ordinales para etiquetar problema de clasificación Fürnkranz and Hüllermeier (2010) y salidas de valores continuos al problema de regresión con salidas múltiples (*Multi-Target Regression*; MTR) Borchani et al. (2015).

La enorme cantidad de datos recolectados por las empresas en la actualidad y el incremento en la diversidad de sensores incorporados en la industria han traído consigo problemas de predicción que involucran grandes volúmenes de datos, ya sea en ejemplos de entrenamiento o cantidad de variables. Este contexto impone serios problemas en términos de eficiencia a las técnicas clásicas de predicción y sobre todo a las que involucran múltiples variables de

salida, es decir, tiempos de procesamiento computacional, excesivamente grandes y en algunos casos intolerables. Aunque en la literatura ya se puede notar una tendencia al aumento de las investigaciones para solucionar este problema, aún se evidencia una dificultad para encontrar técnicas que aborden el problema de MTR desde la perspectiva de Big Data Bahri et al. (2019); Prabhu (2019); Shafique et al. (2019); Li et al. (2019); Asch et al. (2018). Por lo tanto, el objetivo que este artículo de revisión consiste en ofrecer un análisis de las principales técnicas de MTR y sus posibilidades de ser escalados, mediante un enfoque distribuido, para entornos con grandes volúmenes de datos. Para argumentar estas posibilidades seguidamente se abordan varios trabajos referentes a las técnicas de regresión lineal múltiple y los principales modelos de optimización que sustentan los algoritmos de aprendizaje supervisado en el contexto de Big Data.

Materiales y métodos

Recientemente, la tarea de MTR ha sido ampliamente estudiada en la comunidad de investigación de minería de datos. Esta puede verse como una extensión de la tarea de predicción simple, pero donde en lugar de un solo rasgo objetivo, cada ejemplo está asociado con múltiples rasgos objetivos. Este enfoque resuelve aquellos problemas donde para cada ejemplo del conjunto de aprendizaje se presentan varias variables dependientes del resto, cuyos valores deben ser estimados de manera simultánea. Problemas como este, se presentan a menudo en diferentes áreas como la predicción del mercado de valores, el pronóstico de generación de energía, el modelado ecológico, el procesamiento del lenguaje natural, para estimar la calidad de la vegetación Kocev et al. (2009), en la predicción del espectro de audio de ruido del viento (representado por varias variables de presión de sonido) de una componente dada de los vehículos Struyf (2009), entre otros.

En la predicción con salidas múltiples, el trabajo propuesto en Borchani et al. (2015) establece dos grandes enfoques. El primer enfoque, basado en transformación del problema, consiste en emplear cada variable objetivo de conjunto con las variables predictoras, siguiendo determinadas heurísticas para conformar nuevos conjuntos de entrenamientos. En este nuevo espacio se conforman múltiples problemas de salidas simples asumiendo que estas son independientes. Un segundo enfoque basado en adaptación de método, permite adaptar los enfoques clásicos de aprendizaje al nuevo contexto y con ello construir un único modelo para predecir todas las variables de salida simultáneamente. En ambos enfoques se toma en cuenta en alguna medida la posible dependencia inherente entre las variables de salidas. La manera en que ambas familias de algoritmos son capaces de tratar la dependencia entre las variables de salidas, ha sido poco debatido en los trabajos precedentes y hay aspectos que no han sido considerados de manera muy sencilla. Por ejemplo, en las estrategias basadas en transformación se emplean diferentes heurísticas

para combinar las variables de salidas y luego incluirlas en el conjunto de variables predictoras transformado el problema a uno de salida simple. En este escenario el nuevo conjunto de meta-variables dejan de ser independientes e idénticamente distribuidas (*iid* como se conoce por sus siglas en inglés) lo cual es un presupuesto teórico de varios modelos de aprendizaje automático. De igual manera los modelos basados en transformación presuponen que el problema se pueda descomponer por el conjunto de variables de salidas. En tal sentido los modelos basados en adaptación tienen una naturaleza más apropiada para tomar en cuenta la dependencia entre el conjunto de variables de salidas debido a que los algoritmos son adaptados para predecir de manera simultánea el conjunto de variables.

Los algoritmos de MTR obtienen mejores resultados en cuanto al error predictivo cuando se explota la inter dependencia entre las variables de salida. En este sentido, los primeros resultados fueron propuestos en Xioufis et al. (2012), cuyos resultados de predicción pueden ser mejorados en algunas de las bases de datos de dominio específico donde han sido evaluados. Los enfoques más recientemente en el campo de la regresión con salidas múltiples Lu et al. (2012); Rothman et al. (2010); Zhen et al. (2018b,a); Diez et al. (2018) tienen en cuenta la dependencia entre el conjunto de variables de salidas. Para ello utilizan diferentes estructuras matriciales, generalmente de rango deficiente o matrices dispersas, dependiendo de los regularizadores empleados. Una forma de promover matrices dispersas puede ser proyectando a un espacio de mayor dimensión como en el caso de Tsoumakas et al. (2014), donde se generan subespacios aleatorios para formar nuevas variables combinando linealmente el conjunto de las q variables de salidas con pesos aleatorios extraídos de una distribución normal. En los problemas de predicción con salidas múltiples, donde se ha estudiado rigurosamente el problema de la dependencia entre variables de salidas se han utilizado dos pasos de predicción, introduciendo variables latentes y regularizadores convenientes Chen et al. (2011). Estos enfoques compuestos generalmente conducen a formulaciones no convexas que requieren estrategias de resolución particulares. Otros trabajos para la predicción simultánea de todas las variables objetivo al mismo tiempo (el enfoque global) se ha considerado en la configuración de lotes Struyf and Dzeroski (2005). Además, en Appice and Dzeroski (2007) se propuso un algoritmo para la inducción paso a paso de modelos de árboles para múltiples objetivos. En el contexto de *streaming*, también se han hecho algunos trabajos en regresión con múltiples objetivos como es el caso de Ikonomovska et al. (2011).

Regresión lineal multivariada.

En esencia, el objetivo de aprender una regresión lineal con múltiples salidas no es más que aprender un modelo matemático capaz de predecir múltiples variables numéricas de forma simultánea Osojnik et al. (2017) o,

Revista Cubana de Ciencias Informáticas
Vol. 13, No. 4, Mes Octubre-Diciembre, 2019
ISSN: 2227-1899 | RNPS: 2301
http://rcci.uci.cu
Pág. 118-150

formalmente, la tarea de estimar un vector de valores reales $\boldsymbol{y} \in \mathbb{R}^{\boldsymbol{q}}$, donde $\boldsymbol{q}$ es el número de variables para una determinada instancia $\boldsymbol{x}$ de un espacio de entrada $\boldsymbol{\mathcal{X}}$

Dado un conjunto de entrenamiento con $\boldsymbol{N}$ ejemplos, $\boldsymbol{p}$ variables predictoras y $\boldsymbol{q}$ variables de salidas, cada instancia $\boldsymbol{i}$ se caracteriza como un vector de variables descriptivas $\boldsymbol{x^i} = \left(\boldsymbol{x_1^i}, \boldsymbol{x_2^i}, \ldots, \boldsymbol{x_p^i}\right)$ y un vector de variables de salida $\boldsymbol{y^i} = \left(\boldsymbol{y_1^i}, \boldsymbol{y_2^i}, \ldots, \boldsymbol{y_q^i}\right)$ para $\boldsymbol{i} = \boldsymbol{1}, \ldots, \boldsymbol{N}$. La tarea de aprender un modelo de regresión multi-objetivo a partir de una matriz de ejemplos $\boldsymbol{D}$ entonces consiste en encontrar una función $\boldsymbol{h}$ que transforme cada variable de entrada, dada por el vector $\boldsymbol{x^i}$, en un vector de salida $\boldsymbol{y^i}$ conformado por los $\boldsymbol{q}$ rasgos $\boldsymbol{h}$: $\boldsymbol{\mathcal{X}} \rightarrow \mathbb{R}^{\boldsymbol{q}}$.

En general, un problema de regresión lineal multivariada puede ser expresado como un problema de optimización con o sin restricciones donde, se combina una función de perdida $\boldsymbol{\ell(D, W)}$ y una función de regularización $\boldsymbol{R(W)}$ que controla el sobre ajuste del modelo, dependiendo de la naturaleza de los mismos, en la forma:

$$\boldsymbol{\mathcal{L}(D, W)} = \boldsymbol{\ell(D, W)} + \boldsymbol{R(W)} = \sum_{\boldsymbol{i=1}}^{\boldsymbol{N}} \left\| \boldsymbol{y^i} - \boldsymbol{W x^i} \right\|_{\boldsymbol{2}}^{\boldsymbol{2}} + \boldsymbol{R(W)} \qquad (1)$$

donde $\boldsymbol{W} \in \boldsymbol{R^{q \times p}}$ contiene los coeficientes de regresión.

Predicción con salidas múltiples

Entre las técnicas no convencionales de MTR que mayor desarrollo han tenido en los últimos años se encuentra la predicción estructurada (*Structured Prediction*) 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 multietiqueta (cuando son vectores con valores binarios) o predicción con salidas múltiples (cuando las salidas están formadas por vectores reales) Borchani et al. (2015).

Entre los algoritmos representativos basados en transformación del problema, se destacan los trabajos de Spyromitros-Xioufis et al. (2012) y Spyromitros-Xioufis et al. (2016), en los mismos se extienden enfoques clásicos de clasificación multietiqueta conocidos como *Stacking* y *Regressor Chain* para proponer los algoritmos *Multi-Target Stacking* (MTS) y *Ensemble of Regressor Chains* (ERC) respectivamente y sus variantes corregidas MTSC y ERCC, que constituyen el estado del arte en esta área. Para analizar los modelos matemáticos que dan soporte a estos algoritmos se plantea la siguiente notación:

Sea $\boldsymbol{x} = [\boldsymbol{x_1}, \ldots, \boldsymbol{x_p}] \in \mathbb{R}^{\boldsymbol{p}}$, $\boldsymbol{y} = [\boldsymbol{y_1}, \ldots, \boldsymbol{y_q}] \in \mathbb{R}^{\boldsymbol{q}}$, dos vectores aleatorios correspondientes a los espacios de entrada y salida respectivamente. Cada instancia de entrenamiento puede ser escrita como: $\boldsymbol{\mathcal{D}^i} = \left(\boldsymbol{x^i}, \boldsymbol{y^i}\right) \in \boldsymbol{\mathcal{X}} \times \mathbb{R}^{\boldsymbol{q}}$. Un

Revista Cubana de Ciencias Informáticas
Vol. 13, No. 4, Mes Octubre-Diciembre, 2019
ISSN: 2227-1899 | RNPS: 2301
http://rcci.uci.cu
Pág. 118-150

problema de predicción con salidas múltiples en su forma general permite aprender un modelo $\boldsymbol{h}: \boldsymbol{\mathcal{X}} \rightarrow \mathbb{R}^{\boldsymbol{q}}$ sobre el conjunto de entrenamiento $\boldsymbol{\mathcal{D}}$ que permita predecir el conjunto de variables de salida de forma simultánea $\hat{\boldsymbol{y}} = \boldsymbol{h}(\boldsymbol{x})$. En lo sucesivo se describen los métodos representativos que constituyen el estado del arte de estos enfoques.

El algoritmo MTS Spyromitros-Xioufis et al. (2016), fue desarrollado tomando como base el *generalized stacked* para clasificación multi-etiqueta. El proceso de entrenamiento de este algoritmo consta de dos etapas. En la primera etapa, se aprenden $\boldsymbol{q}$ modelos independientes $\boldsymbol{h_j}: \mathbb{R}^{\boldsymbol{p}} \rightarrow \mathbb{R}$ tomando cada conjunto de variables predictoras con cada variable objetivo en forma de relevancia binaria. La segunda etapa del algoritmo MTS aprende $\boldsymbol{q}$ meta-modelos $\boldsymbol{h_j^*}: \mathbb{R}^{\boldsymbol{p+q-1}} \rightarrow \mathbb{R}$ añadiendo a las predictoras, las $\boldsymbol{q-1}$ predicciones de la primera etapa $\boldsymbol{y_{-j}}$ excepto la variable $\boldsymbol{j}$, para obtener $\boldsymbol{p+q-1}$ meta-variables y con ello aprender el predictor en este nuevo espacio. Cada meta-modelo $\boldsymbol{h_j^*}$ se estima sobre la base del conjunto de datos de entrenamiento $\boldsymbol{\mathcal{D}_j^*} = \{([\boldsymbol{x^i}, \boldsymbol{y^i_{-j}}], \boldsymbol{y_j^i})\}$ que se amplía con las predicciones de la primera fase con el algoritmo de relevancia binaria. Nótese que se ha denotado por $[\boldsymbol{x^i}, \boldsymbol{y^i_{-j}}]$ al conjunto ampliado de variables predictoras usando el vector de estimaciones iniciales $\boldsymbol{y^i_{-j}}$ sin tomar en cuenta el valor de la variable en estudio.

Al mismo tiempo el algoritmo ERC propuesto en Spyromitros-Xioufis et al. (2016), se basa en la idea de modelos de encadenamiento para problemas de aprendizaje con salidas simples. El entrenamiento del ERC tiene como punto de partida un ordenamiento aleatorio de las variables de salida $\prod_{\boldsymbol{c}}(\boldsymbol{y_1}, \boldsymbol{y_2}, \ldots, \boldsymbol{y_q})$ a partir de establecer permutaciones aleatorias. Luego, el modelo de aprendizaje sigue dos pasos en forma análoga al MTS para el primer paso. Esto significa que se establece un primer nivel de predicción usando el algoritmo relevancia binaria para cada variable de salida de manera independiente. En la segunda etapa del aprendizaje se van incorporando variables de salida en el proceso de aprendizaje de modo que para estimar la salida $\hat{\boldsymbol{y}}_{\boldsymbol{j}}$ se tomarán en cuenta las anteriores predicciones $\hat{\boldsymbol{y}}_{\boldsymbol{j}}, \ldots \hat{\boldsymbol{y}}_{\boldsymbol{j-1}}$ como entradas para ese modelo. Luego, el algoritmo aprende un primer modelo con las variables predictoras y los restantes modelos $\boldsymbol{h_j^*}: \mathbb{R}^{\boldsymbol{p+j-1}} \rightarrow \mathbb{R}$ se aprenden sobre el conjunto de entrenamiento $\boldsymbol{\mathcal{D}_j^*} = \{([\boldsymbol{x^i}, \boldsymbol{y^i_{-[j:q]}}], \boldsymbol{y_j^i})\}$ transformado en forma de cadena. El principal inconveniente del algoritmo ERC es que depende del ordenamiento de las variables de salida por lo que en Spyromitros-Xioufis et al. (2016) se ejecutan todas las permutaciones posibles del conjunto de variables de salida.

Diversos han sido los trabajos que han sido adaptados en el contexto de máquinas de soporte vectorial para regresión destacándose en los más recientes los trabajos Melki et al. (2017); Chang and Lin (2011); Do and Bui (2019). El primero implica la creación de modelos independientes de SVR para cada variable de salida. El segundo, *Support Vector Regression with Random Chains* (SVRRC), construye un conjunto de cadenas aleatorias utilizando el primer método

Revista Cubana de Ciencias Informáticas
Vol. 13, No. 4, Mes Octubre-Diciembre, 2019
ISSN: 2227-1899 | RNPS: 2301
http://rcci.uci.cu
Pág. 118-150

como modelo base. El tercero, Support Vector *Regression with Correlation Chaining* (SVRCC), calcula las correlaciones de los objetivos y forma una cadena de correlación máxima, que se utiliza para construir un modelo de SVR de cadena única. Los resultados de estos trabajos muestran que el enfoque de SVR de correlación máxima mejora el rendimiento del uso de conjuntos de cadenas aleatorias.

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.

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

$$\tilde{\boldsymbol{y}}_i = \bar{\boldsymbol{y}}_i + \sum_{k=1}^{q} \boldsymbol{b}_{ik}(\tilde{\boldsymbol{y}}_k - \bar{\boldsymbol{y}}_k), \quad \boldsymbol{i} = \boldsymbol{1}, \ldots, \boldsymbol{q} \tag{2}$$

donde:

$$\hat{\boldsymbol{y}}_i = \bar{\boldsymbol{y}}_i + \sum_{j=1}^{p} \hat{\boldsymbol{a}}_{ij}(\boldsymbol{x}_j - \bar{\boldsymbol{x}}_j), \quad \boldsymbol{j} = \boldsymbol{1}, \ldots, \boldsymbol{q} \tag{3}$$

Para estimar los coeficientes de la matriz de regresión lineal $\hat{\boldsymbol{a}}_{ij}$ se utiliza el método de los mínimos cuadrados ordinarios y como regularizador la norma ℓ_2 o *Ridge Regression*

$$\{\hat{\boldsymbol{a}}_{ij}\}_{j=1}^{p} := \boldsymbol{argmin}_{\{a_{ij}\}_{j=1}^{p}} \left(\sum_{n=1}^{N} \left[\boldsymbol{y}_{ni} - \bar{\boldsymbol{y}}_i - \sum_{j=1}^{p} \boldsymbol{a}_j(\boldsymbol{x}_{nj} - \bar{\boldsymbol{x}}_j) \right]^2 + \boldsymbol{\lambda} \sum_{j=1}^{p} \boldsymbol{a}_{ij}^2 \right) \tag{4}$$

Se asume, además, que las variables predictoras y las variables objetivo están centradas por la media correspondiente del conjunto de entrenamiento $\{\boldsymbol{y}_i \leftarrow \boldsymbol{y}_i - \bar{\boldsymbol{y}}_i\}_1^q$, $\{\boldsymbol{x}_j \leftarrow \boldsymbol{x}_j - \bar{\boldsymbol{x}}_j\}_1^p$, así como, todas las variables objetivo estimadas son centradas por la media correspondiente de la muestra $\{\hat{\boldsymbol{y}}_i \leftarrow \hat{\boldsymbol{y}}_i - \bar{\boldsymbol{y}}_i\}_1^q$, $\{\tilde{\boldsymbol{y}}_i \leftarrow \tilde{\boldsymbol{y}}_i - \bar{\boldsymbol{y}}_i\}_1^q$.

Una propuesta para dar solución al problema de dependencia condicional entre las variables de salida se presenta en Zhen et al. (2018a). En este trabajo, se propone un algoritmo denominado Multi-layer Multi-target Regression (MMR), que permite modelar simultáneamente las correlaciones intrínsecas entre las variables de salida y las relaciones no lineales de entrada-salida en un marco general a través de un aprendizaje robusto de bajo rango. Específicamente, el MMR puede codificar explícitamente las correlaciones entre las variables de salida en una matriz de estructura mediante redes elásticas matriciales (Matrix Elastic Nets; MEN). El MMR combina una variante con *kernel* y la regresión lineal multivariada para determinar las relaciones entre el espacio de entrada-salida, posiblemente con dependencia no lineal. Este algoritmo se resuelve mediante un método de optimización desarrollado de manera alternada con convergencia garantizada y descrita en Zhen et al. (2018a). En resumen, esta propuesta presenta un nuevo paradigma de aprendizaje de múltiples capas para la regresión de múltiples variables que está dotado de alta generalidad, flexibilidad y capacidad expresiva. No obstante, el empleo en esta propuesta de un *kernel* de Hilbert de espacio de dimensión infinita (*reproducing kernel Hilbert space*; RKHS) Xu et al. (2015) conduce a un costo computacional relativamente elevado. Un aporte similar, lo constituye el algoritmo *Multitarget Sparse Latent Regression* (MSLR) presentado en Zhen et al. (2018b) el cual aborda este problema al implementar una matriz de con estructura de rango deficiente que permite codificar explícitamente las correlaciones entre las variables de salida utilizando la norma $\boldsymbol{\ell_{2,1}}$.

Otro trabajo del estado del arte que resuelve el problema de regresión lineal multivariada de manera eficiente, introduciendo un espacio de variables latentes o un aprendizaje en capaz, lo constituye el método propuesto en Diez et al. (2018). 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. Para exponer esta propuesta se presenta la siguiente notación:

Sean, $\boldsymbol{x} \in \boldsymbol{\mathcal{X}} \subseteq \mathbb{R}^{\boldsymbol{p}}$ y $\boldsymbol{y} \in \boldsymbol{\mathcal{Y}} \subseteq \mathbb{R}^{\boldsymbol{q}}$ dos vectores aleatorios definidos en los espacios de entrada y salida, respectivamente, de un determinado problema de aprendizaje automático con salidas múltiples reales de dimensión $\boldsymbol{q}$. Para el conjunto de datos, con $\boldsymbol{N}$ instancias de entrenamiento $\boldsymbol{\mathcal{D}} = \{(\boldsymbol{x^1}, \boldsymbol{y^1}), \ldots, (\boldsymbol{x^N}, \boldsymbol{y^N})\}$, se define 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 $\boldsymbol{y} = \boldsymbol{xSW}$. Este modelo define un conjunto de variables latentes $\boldsymbol{z} \in \mathbb{R}^{\boldsymbol{r}}$ entre el conjunto de variables de entrada y salida. Las variables latentes son estimadas a partir de la matriz $\boldsymbol{W_{p \times r}}$ que establece la relación entre el conjunto de entrada y salida. En el caso en que

Revista Cubana de Ciencias Informáticas
Vol. 13, No. 4, Mes Octubre-Diciembre, 2019
ISSN: 2227-1899 | RNPS: 2301
http://rcci.uci.cu
Pág. 118-150

$\boldsymbol{r = q}$ se dice que $\boldsymbol{W}$ 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 $\boldsymbol{S_{r \times q}}$ que combina linealmente el conjunto de predicciones individuales en el modelo de regresión lineal.

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

$$\{\boldsymbol{W^*, S^*}\} = \boldsymbol{argmin_{W,S}} \frac{\boldsymbol{1}}{\boldsymbol{2N}} \|\boldsymbol{SWX - Y}\|_{\boldsymbol{F}}^{\boldsymbol{2}} + \boldsymbol{\lambda_1 g_1(W) + \lambda_2 g_2(S)} \tag{5}$$

donde $\boldsymbol{g_1}$ y $\boldsymbol{g_2}$ representan el término asociado al regularizador para la matriz $\boldsymbol{W}$ y $\boldsymbol{S}$ respectivamente. Vale resaltar que en este contexto $\boldsymbol{g_1}$ y $\boldsymbol{g_2}$ pueden ser genéricos, permiten utilizar varios tipos de normas y son escalables a diferentes esquemas de regularización.

Medidas de Evaluación

En el contexto de la predicción con salidas múltiples se han utilizado como medidas de evaluación más comunes las siguientes:

Average Relative Root Means Squared Error. aRRMSE

$$aRRMSE(h; D_{test}) = \frac{1}{q}\sum_{j=1}^{q}\sqrt{\frac{\sum_{(x,y)\in D_{test}} (h(x_j) - y_j)^2}{\sum_{(x,y)\in D_{test}} (\bar{y}_j - y_j)^2}}$$

Average Root Means Squared Error. aRMSE

$$aRMSE(h; D_{test}) = \frac{1}{q}\sum_{j=1}^{q}\sqrt{\frac{\sum_{(x,y)\in D_{test}} (h(x_j) - y_j)^2}{|D_{test}|}}$$

Means Squared Error. MSE

$$MSE(h; D_{test}) = \sum_{j=1}^{q}\frac{\sum_{(x,y)\in D_{test}} (h(x_j) - y_j)^2}{|D_{test}|}$$

Revista Cubana de Ciencias Informáticas
Vol. 13, No. 4, Mes Octubre-Diciembre, 2019
ISSN: 2227-1899 | RNPS: 2301
http://rcci.uci.cu
Pág. 118-150

Todas estas medidas evalúan cada variable de salida de manera independiente, lo cual da una idea del poder predictivo por cada variable y a su vez una medida global del conjunto de variables. En el caso de las medidas MSE y aRMSE tienen como limitación fundamental que, para bases de datos con variables en intervalos de medición con escalas diferentes, es necesario normalizar dichas escalas de medición haciendo uso de sus valores medios y la varianza, lo cual no ocurre con la medida aRRMSE.

Base de Datos

Un aspecto a tener en cuenta ante toda tarea de MTR es el conjunto de datos a procesar. Si las dimensiones de este son pequeñas basta seleccionar un algoritmo cuyo modelo se ajuste al tipo de los datos y al tipo de función que siguen los datos de salida. En otro caso, cuando los conjuntos de datos son a gran escala, dígase gran número de instancias, gran número de variables de entrada, o gran número de variables de salida, entonces se presentan serias limitaciones en cuanto a eficiencia de los algoritmos. Hay muchos trabajos de investigación que se centran en resolver los problemas de escalabilidad causados por un gran número de instancias de datos, como los métodos de selección de instancias Brighton and Mellish (2002), o la alta dimensionalidad del espacio de características, como los métodos de selección de características Zhai et al. (2014). La causa de las altas dimensiones de salida ha recibido mucha menos atención.

En la literatura se pueden encontrar desde bases de datos pequeñas Yeh (2007) a bases de datos con cientos de miles de instancias Torres-Sospedra and Trilles (2014). La siguiente tabla permite observar la diversidad dimensional de algunas de las bases de datos que pueden ser utilizadas para MTR. En la primera columna se indica el nombre de la base de datos, la segunda se refiere al número de observaciones de la base de datos. La tercera y cuarta columna indican la cantidad de atributos en el espacio de entrada y salida respectivamente. Por último, se señala la fuente de la que fue tomada la base de datos.

Tabla 1. Características de los conjuntos de datos.

Base de datos	# Instancias	# Entradas	# Salidas	Recurso
EDM[1]	154	16	2	Karalič and Bratko (1997)
SF1[2]	323	10	3	Bache and Lichman (2013)
SF2[3]	1066	10	3	Bache and Lichman (2013)
OES97[4]	343	263	16	Spyromitros-Xioufis et al. (2016)
OES10[5]	403	298	16	Spyromitros-Xioufis et al. (2016)

Atp1d[6]	201/136	411	6	Spyromitros-Xioufis et al. (2016)
Atp7d[7]	188/108	411	6	Spyromitros-Xioufis et al. (2016)
RF1[8]	4108/5017	64	8	Spyromitros-Xioufis et al. (2016)
RF2[9]	4108/5017	576	8	Spyromitros-Xioufis et al. (2016)
SCM1d[10]	8145/1658	280	16	Spyromitros-Xioufis et al. (2016)
SCM20d[11]	7463/1503	61	16	Spyromitros-Xioufis et al. (2016)
WQ[12]	1060	16	14	Džeroski et al. (2000)
ENB[13]	768	8	2	Tsanas and Xifara (2012)
SLUMP[14]	103	7	3	Yeh (2007)
ANDRO[15]	49	30	6	Hatzikos et al. (2008)
JURA[16]	359	15	3	Coburn (2000)
OSALES[17]	639	413	12	(2012)
SCPF[18]	1137	23	3	(2013)
SGEMM GPU[19]	241600	15	3	Ballester-Ripoll et al. (2017)
CCU[20]	1111	129	18	Redmond (2017)
UJIIL[21]	21048	525	4	Joaquín Torres-Sospedra and Trilles (2014)
NPMSMP[22]	93239	11	4	Moniz and sTorgo (2018)
AEP[23]	19735	27	2	Luis M. Candanedo (2017)
CBMNPP[24]	11934	16	2	Coraddu et al. (2014)
EGSSD[25]	10000	11	3	Arzamasov (2018)
KEGGMRN[26]	65554	26	3	Shannon and Ideker (2011)
ONPD[27]	65554	61	3	Fernandes et al. (2015)
Superconductivity[28]	21263	79	2	idieh (2018)

- EDM[1] (Electrical Discharge Machining): Es una base de datos que representa los tiempos de descarga eléctrica de ciertos tipos de maquinarias. Este problema consta de dos variables de salida y 16 variables de entrada continuas. Cada variable de salida puede tomar 3 valores distintos (-1,0,1) en dependencia del estado en que se encuentre cada equipo.
- SF1[2] y SF2[3] (Solar Flare): Son dos bases de datos que contiene 3 variables de salida correspondientes a la intensidad, en 3 niveles, de los rayos solares (común, moderada y severa). Los datos recopilados representan los valores observados durante 24 horas de variables relacionadas con indicadores físico-químicos del sol. Hay dos versiones de esta base de datos, SF1 contiene los datos del año 1969 y SF2 del año 1978.

- OES97[4] y OES10[5] (Occupational Employment Survey): Son bases de datos obtenidas en los años 1997 (OES97) y 2010 (OES10) del Occupational Employement Survey compiladas por *US Bureau of Labor Statistics*. Cada instancia (responde a una ciudad) provee el estimado de tiempo laboral de los empleados en cada tipo de oficio para el área metropolitana de cada ciudad. Existen 334 y 403 ciudades (instancias) en la base de datos de 1997 OES97 y 2010 OES10 respectivamente. Las variables de entrada, en estas bases de datos, son una secuencia aleatoria de empleos (ejemplo: doctor, dentista, mecánico) observados en al menos el 50% de las ciudades. Las variables de salida fueron seleccionadas, aleatoriamente, del conjunto de categorías para las cuales se supera el 50% del umbral.
- Atp1d[6] y Atp7d[7] (The Airline Ticket Price): Los conjuntos de datos se refieren a la predicción de los precios de los billetes de avión. Las filas son una secuencia de observaciones en tiempo, ordenado a lo largo de varios das. Cada muestra en esta base de datos representa un conjunto de observaciones de una fecha de salida. Las variables de entrada para cada muestra son valores que pueden ser útiles para la predicción de los precios de los billetes de avión para una fecha de salida específica. Las variables objetivo en estos conjuntos de datos representan los precios al día siguiente (ATP1d) o el precio mínimo observado en los próximos 7 días (ATP7d) para seis preferencias de vuelo estudiadas. La base de dato se compone de 411 características o variables de entrada descritas claramente en Groves and Gini (2011). La naturaleza de estos conjuntos de datos es heterogénea con una mezcla de varios tipos de variables como pueden ser booleanas, de precios o cantidades enteras.
- RF1[8] y RF2[9] (The river flow): Los conjuntos de datos, sobre fluido de los ríos, se refieren a la predicción del comportamiento en las próximas 48 horas de la red fluvial, para zonas geográficas específicas. El conjunto de datos contiene los datos de las observaciones de flujo por hora de 8 sitios en la red del río Mississippi tomados del Servicio Meteorológico Nacional de Estados Unidos. Cada fila incluye la observación más reciente para cada uno de los 8 sitios, así como observaciones en tiempo de 6, 12, 18, 24, 36, 48 y 60 horas en el pasado. En RF1, cada punto de observación, aporta 8 variables para conformar la base de datos. Hay un total de 64 variables de entrada y 8 variables de salida. El conjunto de datos RF2, extiende al conjunto RF1 mediante la adición de información del pronóstico de precipitación para cada uno de los 8 sitios (lluvia esperada reportado como valores discretos: 0,0, 0,01, 0,25, 1,0 pulgadas). Los dos conjuntos de datos contienen más de 1 año de observaciones horarias (>9000 horas) recogidas a partir de septiembre de 2011 hasta el año 2012.
- SCM1d[10] y SCM20d[11] (The Supply Chain Management): El conjunto de datos de la gestión de la cadena de suministro se obtiene de la Competencia *Trading Agent Competition in Supply Chain Management* (TAC SCM)

en el año 2010. Los métodos para pre-procesamiento y normalización de los datos se describen en Groves and Gini (2013). Algunos valores de referencia, para medir la eficacia de la predicción en este tipo de problema, están disponibles en Pardoe and Stone (2010). Las variables de entrada de esta base de dato se corresponden con un da específico del torneo de 220 das recopilados. Además, se incluyen cuatro observaciones de mediciones temporales para cada producto observado y el componente (1,2,4 y 8 días de retardado) para facilitar cierta anticipación de las tendencias de los productos. Los conjuntos de datos contienen 16 variables de salida las cuales se corresponde con el valor medio del día siguiente para la base de datos (SCM1d) o el valor medio para 20 días en la base de datos (SCM20d).

- WQ[12] (Water Quality): Es una base de datos que contiene 14 variables de salida que representa las especies de plantas y animales representativos de los ríos de Eslovenia y 16 variables de entrada que se refieren a los parámetros físicos-químicos que determinan la calidad del agua de estos ríos.
- ENB[13] (The Energy Building): El conjunto de datos relacionado con la energía en edificios (enb) se utilizó para estudiar el efecto de ocho indicadores (compacidad relativa, superficie, superficie de pared, superficie del techo, altura total, orientación, superficie de cristales, distribución de la superficie de cristales) sobre las variables carga térmica (HL) y carga de enfriamiento (CL), de varios edificios residenciales.
- SLUMP[14] (Concrete Slump): Este conjunto de datos permite predecir tres propiedades del hormigón (asentamiento, flujo y resistencia a la compresión) en función del contenido de siete ingredientes (cemento, cenizas volantes, escoria de alto horno, agua, superplastificante, áridos gruesos y áridos finos).
- ANDRO[15] (Andromeda): El conjunto de datos de Andrómeda se utilizó para predecir los valores de seis variables de calidad del agua (temperatura, pH, conductividad, salinidad, oxígeno, turbidez) en el Golfo de Thermaikos de Hessaloniki, Grecia. Las mediciones de las variables objetivo se toman desde sensores submarinos con un intervalo de muestreo de 9s y luego se determina el valor medio de todas las mediciones para un día.
- JURA[16] (Jura): El conjunto de datos de Jura contiene las mediciones de las concentraciones de siete metales pesados (cadmio, cobalto, cromo, cobre, níquel, plomo y zinc), registradas en 359 lugares en la capa superior del suelo de una región del Jura suizo. Adicionalmente, se incorpora información sobre el uso de la tierra (Bosque, Pasto, Pradera y Labranza) y el tipo de roca (Argoviano, Kimmeridgiano, Secuaniano, Portlandiano y Cuaternario). Específicamente, la concentración de tres metales (cadmio, cobre y plomo) es más cara de medir que la de otros metales. Por lo tanto, la concentración de estos metales se trata como variables objetivo, mientras

que el resto de los metales se tratan como variables predictoras de conjunto con variables producidas por información adicional.

- OSALES[17] (Online Product Sales): Esta base de datos que fue empleada en la competencia Kaggles 2012 para predecir el consumo mensual de ventas en línea de productos. Cada línea, en el conjunto de datos, representa un producto de consumo que se describe por sus diversas características, así como indicadores relacionados con una campaña publicitaria (413 características de entrada en total). Hay 12 variables de salida correspondientes a las ventas mensuales tras el lanzamiento del producto.
- SCPF[18] (See Click Predict Fix): Este conjunto de datos permite cuantificar y predecir la cantidad de opiniones, votos y comentarios que podrá recibir un tema específico, a partir de 23 características o variables de entrada. Algunas de estas características son: el número de días que un problema permaneció en línea, la fuente a partir del cual se creó el problema (por ejemplo, androide, iphone, api remoto, etc.), las coordenadas geográficas del problema, entre otros. Los datos han sido tomados en muestras de cuatro ciudades (Oakland, Richmond, New Haven, Chicago) en los EE.UU. y abarcan un período de 12 meses (01/2012-12/2012).
- SGEMM GPU[19] (Sobol Tensor Trains for Global Sensitivity Analysis): Este conjunto de datos mide el tiempo de ejecución de un producto de matriz-matriz A * B = C, donde todas las matrices tienen un tamaño de 2048 x 2048, utilizando un kernel SGEMM GPU parametrizable con 241600 combinaciones de parámetros posibles. Para cada combinación probada, se realizaron 4 ejecuciones y sus resultados se informan cómo las cuatro últimas columnas. Todos los tiempos se miden en milisegundos.
- CCU[20] (Communities and Crime Unnormalized): Los datos combinan datos socioeconómicos del Censo ’90, datos policiales de la década de 1990, estadísticas administrativas y datos de delitos de la UCR del FBI de 1995. Se incluyen muchas variables para que se puedan probar los algoritmos que seleccionan o aprenden de las variables independientes. El FBI señala que el uso de estos datos para evaluar comunidades es demasiado simplista, ya que no se incluyen muchos factores relevantes. Por ejemplo, las comunidades con un gran número de visitantes tendrán un mayor crimen per cápita (medido por los residentes) que las comunidades con menos visitantes, en igualdad de condiciones.
- UJIIL[21] (UJIIndoorLoc): El UJIIndoorLoc es una base de datos de localización de interiores de varios pisos para varios edificios para probar el sistema de posicionamiento en interiores que se basa en la huella digital de WLAN / WiFi. Esta cubre tres edificios de la Universitat Jaume I con 4 o más pisos y casi 110.000m^2. Puede ser utilizado para la clasificación, por ejemplo, identificación real de edificios y pisos, o regresión, por ejemplo, Estimación de longitud y latitud reales. Fue creado en 2013 por medio de más de 20 usuarios diferentes y 25

dispositivos Android. La base de datos consta de 19937 registros de entrenamiento / referencia (archivo trainingData.csv) y 1111 registros de validación / prueba (archivo validationData.csv).

- NPMSMP[22] (News Popularity in Multiple Social Media Platform): Este es un gran conjunto de datos de noticias y sus respectivos comentarios sociales en múltiples plataformas: Facebook, Google+ y LinkedIn. Los datos recopilados se refieren a un período de 8 meses, entre noviembre de 2015 y julio de 2016, que representan alrededor de 100.000 noticias sobre cuatro temas diferentes: economía, Microsoft, Obama y Palestina. Este conjunto de datos está diseñado para comparaciones evaluativas en tareas de análisis predictivo, aunque permite tareas en otras áreas de investigación como detección y seguimiento de temas, análisis de sentimientos en texto corto, detección de la primera historia o recomendación de noticias.
- AEP[23] (Appliances energy prediction): Contiene datos experimentales utilizados para crear modelos de regresión del uso de energía de los aparatos en un edificio de bajo consumo energético. El conjunto de datos es de 10 min durante aproximadamente 4,5 meses. Las condiciones de temperatura y humedad de la casa se controlaron con una red de sensores inalámbricos ZigBee. Cada nodo inalámbrico transmitió las condiciones de temperatura y humedad alrededor de 3.3 min. Luego, los datos inalámbricos se promediaron durante períodos de 10 minutos. Los datos de energía se registraron cada 10 minutos con medidores de energía de m-bus. El clima de la estación meteorológica del aeropuerto más cercano (Chievres Airport, Bélgica) se descargó de un conjunto de datos públicos de Reliable Prognosis (rp5.ru) y se combinó con los conjuntos de datos experimentales utilizando la columna de fecha y hora. Se han incluido dos variables aleatorias en el conjunto de datos para probar los modelos de regresión y para filtrar los atributos no predictivos (parámetros).
- CBMNPP[24] (Condition Based Maintenance of Naval Propulsion Plants): Los experimentos se han llevado a cabo mediante un simulador numérico de un buque naval (Fragata) caracterizado por una planta de propulsión de turbina de gas (Gas Turbine; GT). Los diferentes bloques que forman el simulador completo se han desarrollado y afinado a lo largo del año en varias plantas de propulsión reales similares. En vista de estas observaciones, los datos disponibles concuerdan con un posible buque real. En esta versión del simulador también es posible tener en cuenta la disminución del rendimiento a lo largo del tiempo de los componentes de GT, como el compresor GT y las turbinas. Los datos almacenan una serie de medidas (16 características) que representan indirectamente el estado del sistema sujeto a deterioro del rendimiento.
- EGSSD[25] (Electrical Grid Stability Simulated Data): Este conjunto de datos contiene catos simulados de la estabilidad de la red eléctrica. Estos corresponden al análisis de estabilidad local de un sistema de cuatro nodos

Revista Cubana de Ciencias Informáticas
Vol. 13, No. 4, Mes Octubre-Diciembre, 2019
ISSN: 2227-1899 | RNPS: 2301
http://rcci.uci.cu
Pág. 118-150

donde el productor de electricidad se encuentra en el centro, implementando el concepto de control descentralizado de redes inteligentes.

- KEGGMRN[26] (KEGG Metabolic Reaction Network (Undirected)): El conjunto de datos se refiere a una variedad de características gráficas presentadas en larvas metabólicas según la enciclopedia de genes y genomas de Kyoto (Kyoto Encyclopedia of Genes and Genomes; KEGG) Kanehisa and Goto (2000) modeladas como una red de reacción no dirigida.
- ONPD[27] (Online News Popularity Data): Este conjunto de datos resume un conjunto heterogéneo de características sobre los artículos publicados por Mashable en un período de dos años. El objetivo es predecir el número de acciones en las redes sociales (popularidad).
- Superconductivty[28]: Este conjunto de datos está formado por dos archivos: train.csv contiene 81 ca racterísticas extraídas de 21263 superconductores junto con la temperatura crítica en la columna 82, unique_m.csv contiene la fórmula química dividida para todos los 21263 superconductores de train.csv. Las dos últimas columnas tienen la temperatura crítica y la fórmula química.

MTR. Una revisión para Big Data

Actualmente se siguen desarrollado una gran cantidad de trabajos orientados a resolver los problemas de MTR que reportan muy buenos resultados en sus áreas de aplicación, pero ya sea desde el enfoque adaptativo o desde el de transformación, no se encontraron trabajos que hicieran referencia a la solución de problemas con grandes volúmenes de datos. A pesar de ello, existen problemas reales que ameritan ser estudiados en el contexto de Big data. Ejemplos de estos problemas son las 10 últimas bases de datos caracterizadas en la tabla 1. Estas bases de datos cuentan con gran número de instancias y algunas de ellas de alta dimensionalidad en el espacio de atributos de entrada. Se impone entonces, realizar un análisis de las principales técnicas descritas en este ámbito y su escalabilidad a Big data. En nuestro estudio enfatizaremos en los métodos de optimización disponibles que pueden ser adaptados al contexto de la predicción con salidas múltiples.

Big Data es una disciplina emergente y en pleno auge, la cual se ocupa de todas las actividades relacionadas con los sistemas que manipulan grandes conjuntos de datos. Vale destacar ese sentido las dificultades más habituales vinculadas a la gestión de estas cantidades de datos se centran en la recolección y el almacenamiento Kusnetzky (2010), búsqueda, compartición y análisis Vance (2010), siendo en esta última en la que incurren la mayoría de los modelos de MTR existentes.

Revista Cubana de Ciencias Informáticas
Vol. 13, No. 4, Mes Octubre-Diciembre, 2019
ISSN: 2227-1899 | RNPS: 2301
http://rcci.uci.cu
Pág. 118-150

En este sentido algunos de los principales trabajos aquí expuestos enfrentan varias limitaciones. En MTS por ejemplo, se puede esperar que en presencia de miles de atributos predictores la influencia de tomar una, dos y hasta un número n mucho menor que q de variables de salida como predictoras resulte irrelevante en el esquema de aprendizaje. Eso se acentúa principalmente en los problemas donde p es mucho mayor que q, de manera análoga se vería muy afectado el algoritmo ERC. Cabe mencionar que si N es muy grande y p y q son pequeñas, estas variantes inspiradas en la regresión lineal simple pueden sufrir un problema de redundancia en los datos de entrada, lo que llevaría a un problema de eficiencia computacional en la convergencia de los métodos de optimización y una pobre eficacia.

Desde la disciplina de Big Data, particularmente incorporados en la plataforma Apache Spark Jinliang Wei (2016); Zaharia et al. (2010), ya se pueden encontrar algunos trabajos que, si bien no abordan el problema de MTR, sí resuelven el problema clásico de regresión con una variable de salida. Otros trabajos en ese marco se centran en la solución de problemas de optimización, como por ejemplo el método Limited-memory BFGS (L BFGS) Ge et al. (2018); Livieris et al. (2018) y el del Gradiente Descendiente Estocástico (Stochastic Gradient Descent; SGD) Zhang et al. (2018) del que cabe señalar, parte el método del gradiente proximal acelerado empleado en GMLR.

La siguiente tabla detalla una síntesis de los principales trabajos de MTR y su relación con los modelos de optimización y los métodos de solución empleados. Sobre esta base es posible extender los algoritmos de MTR al escenario de grandes volúmenes de datos.

Tabla 2. Algoritmos de MTR

Algoritmo	Tipo	M. Solver	Regularizadores	Recurso
MORTs	Adaptación	Regression Tree	-	Struyf and Dzeroski (2005)
FIMT-DD	Adaptación	Regression Tree	-	Ikonomovska et al. (2011)
RMTL	Adaptación	GD con aceleración de Nesterov	$\ell_*, \ell_{2,1}$	Chen et al. (2011)
MTRS & RC	Transformación	Ridge regression, SVR, Regression Tree, Stochastic Gradient Boosting	Ridge	Spyromitros-Xioufis et al. (2012)
RLTC	Transformación	Regression Tree	-	Tsoumakas et al. (2014)
SST & ERC	Transformación	Ridge regression, SVR, Regression Tree, Stochastic Gradient Boosting	ℓ_1, ℓ_2	Spyromitros-Xioufis et al. (2016)
SVRRC & SVRCC	Transformación	SVR	Ridge	Zhen et al. (2018a)
MMR	Adaptación	GD, Sylvester	ℓ_2 Matrix Elastic-Net	Zhen et al. (2018a)
MSLR	Adaptación	Solución analtica, Sylvester	$\ell_{2,1}$	Zhen et al. (2018b)
GMLR	Adaptación	Gradiente Proximal Acelerado	$\ell_*, \ell_1, \ell_{2,1}$, Elastic Net	Diez et al. (2018)

Revista Cubana de Ciencias Informáticas
Vol. 13, No. 4, Mes Octubre-Diciembre, 2019
ISSN: 2227-1899 | RNPS: 2301
http://rcci.uci.cu
Pág. 118-150

Métodos de optimización en Big Data

El método más simple para resolver problemas de optimización de la forma $\boldsymbol{mín}_{\boldsymbol{w}\in\boldsymbol{R}^{\boldsymbol{d}}}\, \boldsymbol{f}(\boldsymbol{w})$ es el Gradiente Descendiente (Gradient Descent, GD) Wang and Hu (2019); Ma et al. (2018). Este, como los demás métodos de optimización de primer orden Vakili and Zhao (2019) (incluidas las variantes estocásticas de los mismos) son muy adecuados para el cálculo a gran escala y distribuido. Los métodos de descenso de gradiente tienen como objetivo encontrar un mínimo local de una función dando pasos de forma iterativa en la dirección del descenso más pronunciado, que es el negativo de la derivada (llamada gradiente) de la función en el punto actual, es decir, en el valor del parámetro actual. Si la función objetivo f no es diferenciable en todos los argumentos, pero sigue siendo convexa, entonces un sub-gradiente es la generalización natural del gradiente, y asume el papel de la dirección del paso. En cualquier caso, calcular un gradiente o un sub-gradiente de f es costoso: requiere un pase completo a través del conjunto de datos completo para calcular las contribuciones de todos los términos de pérdida. Los problemas de optimización cuya función objetivo f se escribe como una suma son particularmente adecuados para resolverse utilizando el descenso de gradiente estocástico (SGD). Un problema de optimización convexa en aprendizaje supervisado generalmente se define como,

$$\boldsymbol{f}(\boldsymbol{w}) := \boldsymbol{\lambda}\boldsymbol{\mathcal{R}}(\boldsymbol{w}) + \frac{\boldsymbol{1}}{\boldsymbol{n}}\sum_{\boldsymbol{i}=\boldsymbol{1}}^{\boldsymbol{n}} \boldsymbol{\ell}(\boldsymbol{w}, \boldsymbol{x_i}, \boldsymbol{y_i}) \tag{6}$$

Un subgradiente estocástico selecciona aleatoriamente de un vector un conjunto reducido de datos, sin afectar el valor del sub-gradiente. Para lograr la convergencia se realizan las selecciones aleatorias en cada iteración del problema de optimización. Seleccionando un punto de datos $\boldsymbol{i} \in [\boldsymbol{1} \ldots \boldsymbol{n}]$ de manera uniforme al azar, obtenemos un subgradiente estocástico de (7), con respecto a w de la siguiente manera:

$$\boldsymbol{f'_{w,i}} := \boldsymbol{\ell'_{w,i}} + \boldsymbol{\lambda}\boldsymbol{\mathcal{R}'_w} \tag{7}$$

donde $\boldsymbol{\ell'_{w,i}} \in \mathbb{R}^{\boldsymbol{d}}$ es un subgradiente de la parte de la función de pérdida determinada por el i-ésimo punto de datos, que es $\boldsymbol{\ell'_{w,i}} \in \frac{\boldsymbol{\partial}}{\boldsymbol{\partial w}}\boldsymbol{\ell}(\boldsymbol{w}, \boldsymbol{x_i}, \boldsymbol{y_i})$. Además, $\mathcal{R}'_w$ es un sub-gradiente del regularizador $\boldsymbol{\mathcal{R}}(\boldsymbol{w})$, es decir, $\boldsymbol{\mathcal{R}'_w} \in \frac{\boldsymbol{\partial}}{\boldsymbol{\partial w}}\boldsymbol{\mathcal{R}}(\boldsymbol{w})$. El término $\boldsymbol{\mathcal{R}'_w}$ no depende de qué punto de datos aleatorio se elija. Claramente, en la selección aleatoria de $\boldsymbol{i} \in [\boldsymbol{1} \ldots \boldsymbol{n}]$ se tiene que $\boldsymbol{f'_{w,i}}$ es un sub-gradiente de la función objetivo $\boldsymbol{f}$, o lo que es equivalente $\mathbb{E}[\boldsymbol{f'_{w,i}}] \in \frac{\boldsymbol{\partial}}{\boldsymbol{\partial w}}\boldsymbol{f}(\boldsymbol{w})$.

La búsqueda del SGD consiste en encontrar un nuevo punto en la dirección del sub-gradiente estocástico negativo sobre la función objetivo en la forma:

$$\boldsymbol{w^{(t+1)}} := \boldsymbol{w^{(t)}} - \boldsymbol{\gamma}\boldsymbol{f'_{w,i}} \tag{8}$$

El SGD distribuido implementado en Apache Spark utiliza una muestra simple (distribuida) de los datos de ejemplos. Dado que esto requeriría acceso al conjunto completo de datos, el parámetro miniBatchFraction especifica que fracción

Revista Cubana de Ciencias Informáticas
Vol. 13, No. 4, Mes Octubre-Diciembre, 2019
ISSN: 2227-1899 | RNPS: 2301
http://rcci.uci.cu
Pág. 118-150

de los datos completos se debe usar en su lugar. El promedio de los gradientes sobre este subconjunto, es decir, $\frac{1}{|S|}\sum_{i\in S} \ell'_{w,i}$, es un gradiente estocástico. Aquí *S* es el subconjunto muestreado de tamaño $|S| = miniBatchFraction * n$. En cada iteración, el muestreo sobre el conjunto de datos distribuido, así como el cálculo de la suma de los resultados parciales de cada máquina, se realiza mediante las rutinas de Spark estándares. Si miniBatchFraction se establece en 1 (predeterminado), el paso resultante en cada iteración es exactamente el subgradiente descendiente. En este caso, no hay aleatoriedad ni variación en las direcciones de los pasos. En el otro extremo, si se elige miniBatchFraction muy pequeño, de modo que solo se muestrea un solo punto, es decir, $|S| = miniBatchFraction * n = 1$, entonces el algoritmo es equivalente al SGD estándar. En ese caso, la dirección del paso depende de la muestra aleatoriamente uniforme del punto.

Se han desarrollado otros algoritmos inspirados en el SGD para el procesamiento eficiente de grandes volúmenes de datos; un ejemplo lo constituye el algoritmo *Accelerated Mini-batch Randomized Block Coordinate Descent Method* (MRBCD) propuesto en Zhao et al. (2014). MRBCD emplea una función de regularización separable por bloques, lo que permite resolver los problemas de minimización mediante un descenso coordinado de bloques al azar (*Randomized Block Coordinate Descent*; RBCD). Los métodos RBCD existentes generalmente disminuyen el valor objetivo al explotar el gradiente parcial de un bloque de coordenadas seleccionado al azar en cada iteración. Sin embargo, tal configuración de "batch" puede ser computacionalmente costosa en la práctica. Para superar este inconveniente anterior, el método MRBCD es doblemente estocástico, en el sentido de que no solo selecciona aleatoriamente un bloque de coordenadas, pero también muestrea aleatoriamente un mini-lote de funciones de componentes de todos los f_i. Dado que la varianza introducida por el muestreo estocástico sobre las funciones de los componentes no va a cero a medida que aumenta el número de iteraciones, este utiliza una secuencia de tamaños de pasos decrecientes.

Como ya se había mencionado, otro algoritmo de optimización disponible en Apache Spark es L-BFGS. Este es un algoritmo de optimización en la familia de métodos cuasi-Newton para resolver los problemas de optimización de la forma $min_{w\in R^d} f(w)$. El método L-BFGS aproxima la función objetivo localmente como una acción cuadrática sin evaluar las segundas derivadas parciales de la función objetivo para construir la matriz Hessiana. La matriz Hessiana es aproximada por a las evaluaciones de gradientes anteriores, por lo que no hay un problema de escalabilidad vertical (el número de funciones de entrenamiento) cuando se calcula explícitamente la matriz Hessiana en el método de Newton. Como resultado, L-BFGS a menudo logra una convergencia más rápida en comparación con otros problemas de optimización de primer orden.

En Chen et al. (2014) se propone un nuevo algoritmo L-BFGS, llamado VL-BFGS, que evita las operaciones de productos de puntos costosos en la recursión de dos bucles y mejora en gran medida la eficiencia del cálculo con un

alto grado de paralelismo. Este algoritmo se escala muy bien y permite una variedad de algoritmos de aprendizaje automático para manejar un gran número de variables en grandes conjuntos de datos. La recursión central de dos bucles en VL-BFGS es independiente sobre el número de variables. Esto permite que sea fácilmente paralelizado en Map-Reduce Mu et al. (2018) y escalar hasta miles de millones de variables. En resumen, el algoritmo VL-BFGS tiene una complejidad general similar, pero nace con un grado masivo de paralelismo.

En Najafabadi et al. (2017) siguen el principio básico de que para implementar un algoritmo L-BFGS a gran escala donde la longitud del vector de parámetros $\boldsymbol{x}$ es muy grande, una solución natural será almacenar y manipular el vector $\boldsymbol{x}$ en varias máquinas. Si se utilizan $\boldsymbol{N}$ máquinas, el vector de parámetros se divide en $\boldsymbol{N}$ particiones no superpuestas. Cada partición se almacena y manipula localmente en cada máquina. La distribución del almacenamiento y los cálculos en varias máquinas beneficia tanto los requisitos de memoria como los tiempos de ejecución computacionales. Siguiendo esta idea cada máquina manipula la porción del vector de parámetro asignado localmente. Los cachs L-BFGS (pares $\boldsymbol{s_i}$, $\boldsymbol{y_i}$) también se almacenan en las máquinas localmente. Por ejemplo, si la máquina *j*-ésima almacena la *j*-ésima partición del vector de parámetros, también termina almacenando la *j*-ésima partición de los vectores $\boldsymbol{s_i}$ y $\boldsymbol{y_i}$, realizando todos los cálculos localmente. Cada máquina realiza la mayoría de las operaciones de forma independiente. Por ejemplo, la suma de dos vectores que están ambos distribuidos en varias máquinas, incluye sumar sus particiones correspondientes en cada máquina localmente.

Una técnica que como se ha visto, está estrechamente vinculada a los algoritmos de MTR basados en transformación y suele emplear algoritmos de optimización conexa es la regresión lineal múltiple.

Regresión Linear Múltiple desde Big Data

Como muchos métodos de aprendizaje automático, la Regresión Lineal Múltiple (*Multiple Linear Regression*; MLR) Salleh et al. (2017); Sherimon and Cherian (2017) se puede formular como un problema de optimización convexo. Formalmente, esto se puede escribir como el problema de optimización $\boldsymbol{mín_{w\in R^d}\, f(w)}$, donde la función objetivo es de la forma:

$$\boldsymbol{f(w) := \lambda \mathcal{R}(w) + \frac{1}{n}\sum_{i=1}^{n} \ell(w, x_i, y_i)} \quad (9)$$

Aquí los vectores $\boldsymbol{x_i \in R^d}$ son los ejemplos de datos de entrenamiento, para $\boldsymbol{1 \leq i \leq n}$, y $\boldsymbol{y_i \in R}$ que son los valores a predecir. En la regresión lineal, caso múltiple, la función de pérdida en la formulación dada por la pérdida al cuadrado:

$$\boldsymbol{\ell(w, x, y) := \frac{1}{2}\left(w^T x - y\right)^2} \quad (10)$$

Diversos métodos de regresión relacionados se derivan utilizando diferentes tipos de regularización: mínimos cuadrados ordinarios o mínimos cuadrados lineales no utiliza la regularización; la regresión *ridge* utiliza la regularización de $\boldsymbol{\ell_2}$;

Revista Cubana de Ciencias Informáticas
Vol. 13, No. 4, Mes Octubre-Diciembre, 2019
ISSN: 2227-1899 | RNPS: 2301
http://rcci.uci.cu
Pág. 118-150

y *Lasso* usa la regularización de $\boldsymbol{\ell_1}$. Para todos estos modelos, la pérdida media o error de entrenamiento, $\frac{1}{n}\sum_{i=1}^{n}\left(w^T x_i - y_i\right)^2$, mejor conocida como MSE. En ese sentido la biblioteca MLlib Assefi et al. (2017) de Apache Spark ya incorpora esta técnica utilizando como método de optimización SDG o L-BFGS.

Otros trabajos en Big Data para resolver problemas de MLR se basan en la descomposición el problema de optimización en varios subproblemas. En Jun et al. (2015) por ejemplo, dividen el conjunto de datos de aprendizaje en M sub-conjuntos, entrenar los **M** modelos y combinar los $\boldsymbol{w_i, i} \in [\boldsymbol{1 \ldots n/M}]$ vectores resultantes como la media de cada una de sus componentes, es decir, y por último evaluando la calidad del modelo final obtenido mediante el MSE. El trabajo Adjout and Boufares (2014), se basa en la misma idea de descomponer el problema en $\boldsymbol{N}$ sub-problemas pero resuelve cada uno de estos mediante la doscomposición $\boldsymbol{QR}$ Benson et al. (2013) de la matriz de valores de entrada $\boldsymbol{X}$ asociada; luego se obtienen las matrices $\boldsymbol{Q}$ y $\boldsymbol{R}$ asociadas al problema original mediante la concatenación de las resultantes de cada subproblema.

En Tejasviram et al. (2015) se propone un modelo híbrido que combina la Máquina de Aprendizaje Extremo Asociativo Automático (*Auto Associative Extreme Learning Machine*; AAELM) con MLR (AAELM+MLR) para realizar la regresión en Big Data. Este funciona utilizando el modelo de computación paralela Hadoop MapReduce que se implementa en Python utilizando la API Dumbo. Funciona en dos fases. En la primera fase, se entrena AAELM de tres capas. La salida de los nodos ocultos de AAELM se trata como NLPC. En la segunda fase, el modelo MLR se ajusta utilizando estos NLPC como variables de entrada. La efectividad del modelo AAELM + MLR se demuestra en dos grandes conjuntos de datos, a saber, el conjunto de datos de retardo de vuelo de la aerolínea y el conjunto de datos de sensores de gas, tomados de la web. Se observa que AAELM + MLR superó el rendimiento del modelo MLR al producir menos error de media al cuadrado promedio (MSE) y valores MAPE en el marco de validación cruzada de 10 veces. Una prueba estadística confirma su superioridad a un nivel de significación del 1

Otros trabajos utilizan el framework MapReduce Sona and Mulerikkal (2017). Por ejemplo, en Meng and Mahoney (2013) describen un algoritmo con mejores propiedades de comunicación que es eficientes para resolver problemas de regresión $\boldsymbol{\ell_p}$ fuertemente sobre-determinados a precisión moderada en MapReduce. Este algoritmo se basa en 4 aspectos de particular interés: utiliza un algoritmo de redondeo rápido recientemente desarrollado (que toma $\boldsymbol{O(mn^3 \log m)}$ tiempo) para construir un redondeo $\boldsymbol{2n}$ de un conjunto convexo centralmente simétrico en $\boldsymbol{\mathcal{R}^n}$ Clarkson et al. (2016) para construir un algoritmo de condicionamiento determinístico de una sola pasada para la regresión $\boldsymbol{\ell_p}$, mediante el uso de una forma restringida de la regresión $\boldsymbol{\ell_p}$ (que también se utilizó recientemente Clarkson et al. (2016), se muestra que el método de muestreo aleatorio para preservar el sub espacio Dasgupta et al. (2009) se puede implementar

(fácilmente) en el marco de MapReduce en una sola pasada, al utilizar múltiples soluciones de sub-muestreo del muestreo aleatorio de un solo paso, se puede construir una pequeña región de búsqueda inicial para los métodos de plano de corte de puntos interiores (*interior point cutting-plane methods*; IPCPM) Naoum-Sawaya (2011), por último, al realizar en paralelo múltiples consultas en cada iteración, obtienen un IPCPM aleatorio para resolver el problema de regresión ℓ_p convexo. En este trabajo además de describir el algoritmo básico, también se presentan los resultados empíricos de una implementación numérica de este algoritmo aplicado a los problemas de regresión ℓ_1 en conjuntos de datos de hasta un terabyte de tamaño.

Conclusiones

En esta revisión se logró resumir las principales técnicas de MTR y su factibilidad en problemas con grandes volúmenes de datos, concluyendo que estas presentan serias deficiencias en este tipo de tareas. Cabe señalar que no se encontró ninguna literatura sobre trabajos del área para problemas de Big Data. Se debe resaltar, que el enfoque de transformación presenta la posibilidad de partiendo de una técnica de regresión lineal, desarrollar algoritmos para MTR, lo que, ligado a la existencia de exponentes de esta técnica para Big Data, como la incorporada en el framework Apache Spark, sugiere la posibilidad de desarrollar nuevos trabajos de MTR basados en estos exponentes que permitan afrontar el manejo de grandes volúmenes de datos. En cambio, algunos algoritmos como GLMR están basados en la optimización de funciones convexas, área en la que se han desarrollados varios trabajos para problemas con muchos datos algunos de los cuáles también cuentan con implementaciones para Big Data como las implementadas en el mencionado framework. Por tanto, se sugiere replantear aquellas técnicas cuya esencia ya ha sido afrontada desde Big Data, de modo que surjan nuevos trabajos orientados a la solución eficiente de problemas de MTR en este campo.

Referencias

Kaggle (2012). Online product sales. https://www.kaggle.com/c/onli ne-sales., 2012. URL https://www.kaggle.com/c/online-sales.

Kaggle (2013). See click predict fix. https://www.kaggle.com/c/see- click-predict-fix., 2013. URL https://www.kaggle.com/c/see-click-predict-fix.

M. R. Adjout and F. Boufares. A massively parallel processing for the multiple linear regression. In *2014 Tenth International Conference on Signal-Image Technology and Internet-Based Systems*, pages 666–671, Nov 2014. doi: 10.1109/SITIS.2014.26.

Annalisa Appice and Saso Dzeroski. Stepwise induction of multi-target model trees, 2007. URL https://doi.org/10.1007/978-3-540-74958-5_46.

Vadim Arzamasov. Uci machine learning repository. electrical grid stability simulated data, 2018. URL http://archive.ics.uci.edu/ml.

M. Asch, Terry Moore, Rosa M. Badia, Micah Beck, Peter H. Beckman, T. Bidot, François Bodin, Franck Cappello, Alok N. Choudhary, Bronis R. de Supinski, Ewa Deelman, Jack J. Dongarra, Anshu Dubey, Geof frey C. Fox, H. Fu, Sergi Girona, William Gropp, Michael A. Heroux, Yutaka Ishikawa, Katarzyna Keahey, David E. Keyes, Bill Kramer, J.-F. Lavignon, Y. Lu, Satoshi Matsuoka, Bernd Mohr, Daniel A. Reed, S. Requena, Joel H. Saltz, Thomas C. Schulthess, Rick L. Stevens, D. Martin Swany, Alexander S. Szalay, William M. Tang, G. Varoquaux, Jean-Pierre Vilotte, Robert W. Wisniewski, Z. Xu, and I Zacharov. Big data and extreme-scale computing. *IJHPCA*, 32(4):435–479, 2018. doi: 10.1177/1094342018778123. URL https://doi.org/10.1177/1094342018778123.

Mehdi Assefi, Ehsun Behravesh, Guangchi Liu, and Ahmad Pahlavan Tafti. Big data machine learning using apache spark mllib. In *2017 IEEE International Conference on Big Data, BigData 2017, Boston, MA, USA, December 11-14, 2017*, pages 3492–3498, 2017. doi: 10.1109/BigData.2017.8258338. URL https://doi.org/10.1109/BigData.2017.8258338.

Kevin Bache and Moshe Lichman. Uci machine learning repository (http://archive.ics.uci.edu/ml), university of california, school of information and computer science. *Irvine, CA*, 2013.

Safa Bahri, Nesrine Zoghlami, Mourad Abed, and João Manuel R. S. Tavares. BIG DATA for healthcare: A survey. *IEEE Access*, 7:7397–7408, 2019. doi: 10.1109/ACCESS.2018.2889180. URL https://doi.org/10.1109/ACCESS.2018.2889180.

Gükhan H. Bakir, Thomas Hofmann, Bernhar d Schölkopf, Alexander J. Smola, Ben Taskar, and S. V. N. Vishwanathan. *Predicting Structured Data (Neural Information Processing)*. The MIT Press, 2007. ISBN 0262026171.

Rafael Ballester-Ripoll, Enrique G. Paredes, and Renato Pajarola. Sobol tensor trains for global sensitivity analysis. *CoRR*, abs/1712.00233, 2017. URL http://arxiv.org/abs/1712.00233.

Austin R. Benson, David F. Gleich, and James Demmel. Direct QR factorizations for tall-and-skinny matrices in mapreduce architectures. In *Proceedings of the 2013 IEEE International Conference on Big Data, 6 9 October 2013, Santa Clara, CA, USA*, pages 264–272, 2013. doi: 10.1109/BigData.2013.6691583. URL https://doi.org/10.1109/BigData.2013.6691583.

María Bermúdez, César A. HUERTAS, Carlos E. OBANDO, and Carlos F. VALENCIA, editors. *Predicción de inundaciones fluviales en un núcleo costero mediante un modelo de regresión de Máquinas de Vectores Soporte de Mínimos Cuadrados (LS-SVM)*, JIA 2017, 2017. V Jornadas de Ingeniería del Agua. URL http://geama.org/jia2017/wp-content/uploads/ponencias/tema_M/m18.pdf.

Concha Bielza, Guangdi Li, and Pedro Larrañaga. Multi-dimensional classification with bayesian networks. *Int. J. Approx. Reasoning*, 52(6):705–727, 2011. doi: 10.1016/j.ijar.2011.01.007. URL https://doi.org/10.1016/j.ijar.2011.01.007.

Hanen Borchani, Gherardo Varando, Concha Bielza, and Pedro Larrañaga. A survey on multi-output regression. *Wiley Int. Rev. Data Min. and Knowl. Disc.*, 5(5):216–233, September 2015. ISSN 1942-4787. doi: 10.1002/widm.1157. URL http://dx.doi.org/10.1002/widm.1157.

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.

Henry Brighton and Chris Mellish. Advances in instance selection for instance-based learning algorithms. *Data Min. Knowl. Discov.*, 6(2):153–172, 2002. doi: 10.1023/A:1014043630878. URL https://doi.org/10.1023/A:1014043630878.

Serhat Selcuk Bucak, Pavan Kumar Mallapragada, Rong Jin, and Anil K. Jain. Efficient multi-label ranking for multi-class learning: Application to object recognition. In *IEEE 12th International Conference on Computer Vision, ICCV 2009, Kyoto, Japan, September 27 - October 4, 2009*, pages 2098–2105, 2009. doi: 10.1109/ICCV.2009.5459460. URL https://doi.org/10.1109/ICCV.2009.5459460.

Arys Carrasquilla-Batista, Johnny Valverde-Cerdas, and Maritza Guerrero-Barrantes. Regresión lineal simple y múltiple: aplicación en la predicción de variables naturales relacionadas con el crecimiento microalgal. *Revista Tecnologia en Marcha*, 2016. URL https://doi.org/10.18845/tm.v29i8.2983.

Yamile CASTRO, César A. HUERTAS, Carlos E. OBANDO, and Carlos F. VALENCIA. Análisis de super vivencia para predicción de bancarrota: Caso de las industrias minoristas en colombia. *Espacios*, 40(1), 1 2019. ISSN 0798 1015.

Chih-Chung Chang and Chih-Jen Lin. Libsvm: a library for support vector machines. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 2(3):27, 2011.

Jianhui Chen, Jiayu Zhou, and Jieping Ye. Integrating low-rank and group-sparse structures for robust multi task learning, 2011. URL http://doi.acm.org/10.1145/2020408.2020423.

Weizhu Chen, Zhenghao Wang, and Jingren Zhou. Large-scale L-BFGS using mapreduce. In *Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing Systems 2014, December 8-13 2014, Montreal, Quebec, Canada*, pages 1332–1340, 2014. URL http://papers.nips.cc/paper/5333-large-scale-l-bfgs-using-mapreduce.

Kenneth L. Clarkson, Petros Drineas, Malik Magdon-Ismail, Michael W. Mahoney, Xiangrui Meng, and David P. Woodruff. The fast cauchy transform and faster robust linear regression. *SIAM J. Comput.*, 45(3):763–810, 2016. doi: 10.1137/140963698. URL https://doi.org/10.1137/140963698.

Timothy C Coburn. Geostatistics for natural resources evaluation, 2000.

Andrea Coraddu, Luca Oneto, Alessandro Ghio, Stefano Savio, Davide Anguita, and Massimo Figari. Machi ne learning approaches for improving condition-based maintenance of naval propulsion plants. *Journal of* Engineering *for the Maritime Environment*, – (–): –, 2014.

Anirban Dasgupta, Petros Drineas, Boulos Harb, Ravi Kumar, and Michael W. Mahoney. Sampling algorithms and coresets for $\ellp regression. *SIAM J. Comput.*, 38(5):2060–2078, 2009. doi:10.1137/070696507. URL https://doi.org/10.1137/070696507.

Hector Raúl González Diez, Carlos A. Morell Pérez, and Francesc J. Ferri. *Extensión del aprendizaje basado en instancia para problemas de predicción con salidas múltiples.* Thesis, 2018.

Thanh-Nghi Do and Le-Diem Bui. Parallel learning algorithms of local support vector regression for dealing with large datasets. *T. Large-Scale Data- and Knowledge-Centered Systems*, 41:59–77, 2019. doi: 10.1007/ 978-3-662-58808-6_3. URL https://doi.org/10.1007/978-3-662-58808-6_3.

Sašo Džeroski, Damjan Demšar, and Jasna Grbović. Predicting chemical parameters of river water quality from bioindicator data. *Applied Intelligence*, 13(1):7–17, 2000.

Kelwin Fernandes, Pedro Vinagre, and Paulo Cortez. A proactive intelligent decision support system for predicting the popularity of online news. In *Progress in Artificial Intelligence - 17th Portuguese Conference on Artificial Intelligence, EPIA 2015, Coimbra, Portugal, September 8-11, 2015. Proceedings*, pages 535–546, 2015. doi: 10.1007/978-3-319-23485-4_53. URL https://doi.org/10.1007/978-3-319-23485-4_53.

Johannes Fürnkranz and Eyke Hüllermeier, editors. *Preference Learning*. Springer, 2010. ISBN 978-3-642 14124-9. doi: 10.1007/978-3-642-14125-6. URL https://doi.org/10.1007/978-3-642-14125-6.

Fuchao Ge, Yuntao Ju, Zhinan Qi, and Yi Lin. Parameter estimation of a gaussian mixture model for wind power forecast error by riemann L-BFGS optimization. *IEEE Access*, 6:38892–38899, 2018. doi: 10.1109/ACCESS.2018.2852501. URL https://doi.org/10.1109/ACCESS.2018.2852501.

George Carlos Mogollón Gonzales. Predicción de resultados metalúrgicos en flotación de mineral es mediante análisis multivariante y aprendizaje automático. Master's thesis, Universidad Politécnica de Valencia, 2018.

William Groves and Maria Gini. A regression model for predicting optimal purchase timing for airline tickets. Technical report, Technical Report 11-025, University of Minnesota, Minneapolis, MN, 2011.

William Groves and Maria Gini. Improving prediction in tac scm by integrating multivariate and tempo ral aspects via pls regression. In *Agent-Mediated Electronic Commerce. Designing Trading Strategies and Mechanisms for Electronic Markets*, pages 28–43. Springer, 2013.

Evaggelos V Hatzikos, Grigorios Tsoumakas, George Tzanis, Nick Bassiliades, and Ioannis Vlahavas. An empi rical study on sea water quality prediction. *Knowledge-Based Systems*, 21(6):471–478, 2008.

Arthur E Hoerl and Robert W Kennard. Ridge regression: Biased estimation for nonorthogonal problems. *Technometrics*, 12(1):55–67, 1970.

Kam Ham idieh. Uci machine learning repository. superconductivty, 2018. URL http://archive.ics.uci.edu/ml.

Elena Ikonomovska, JoÃ£o Gama, and Saso Dzeroski. Learning model trees from evolving data streams. *Data Min. Knowl. Discov.*, 23(1):128–168, 2011. doi: 10.1007/s10618-010-0201-y. URL https://doi.org/10.1007/s10618-010-0201-y.

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.

Jeylin Jeylin Meybelin Pérez Obregón and Tonys Romero Díaz Díaz. Análisis del rendimiento académico mediante regresión logística y múltiple. *Revista Electrónica de Conocimientos, Saberes y Prácticas*, 1:33–42, 10 2018. doi: 10.30698/recsp.v1i2.10.

Garth A. Gibson Jinliang Wei, Jin Kyu Kim. Benchmarking apache spark with machine learning applications. 2016.

Adolfo Martínez-Usó Tomar J. Arnau Joan P. Avariento Mauri Benedito-Bordonau Joaquín Huerta Yasmina Andreu óscar Belmonte Vicent Castello Irene Garcia-Martı Diego Gargallo Carlos González Nadal Francisco Josep López Ruben Martínez Roberto Mediero Javier Ortells Nacho Piqueras Ianisse Quizán David Rambla Luis E. Rodríguez Eva Salvador Balaguer Ana Sanchís Carlos Serra Joaquín Torres-Sospedra, Raul Montoliu and Sergi Trilles. Uci machine learning repository. ujiindoorloc, 2014. URL http://archive.ics.uci.edu/ml.

Sunghae Jun, Seung-Joo Lee, and Jea-Bok Ryu. A divided regression analysis for big data. 2015.

Minoru Kanehisa and Susumu Goto. KEGG: kyoto encyclopedia of genes and genomes. *Nucleic Acids Research*, 28(1):27–30, 2000. doi: 10.1093/nar/28.1.27. URL https://doi.org/10.1093/nar/28.1.27.

Aram Karalič and Ivan Bratko. First order regression. *Machine Learning*, 26(2-3):147–176, 1997.

Dragi Kocev, SaÅ¡o DÅ$^{3}_{4}$eroski, Matt D. White, Graeme Newell, and Peter Griffioen. *Using single- and multi target regression trees and ensembles to model a compound index of vegetation condition*, volume 220. 2009. doi: 10.1016/j.ecolmodel.2009.01.037.

Philipp Koehn. Europarl: A parallel corpus for statistical machine translation, 2005.

Dan Kusnetzky. What is "big data?", 2010. URL https://www.zdnet.com/article/what-is-big-data/.

Gang Li, Jianlong Tan, and Sohail S. Chaudhry. Industry 4.0 and big data innovations. *Enterprise IS*, 13(2):145–147, 2019. doi: 10.1080/17517575.2018.1554190. URL https://doi.org/10.1080/17517575.2018.1554190.

Yan Liu, Eric P. Xing, and Jaime G. Carbonell. Predicting protein folds with structural repeats using a chain graph model. In *Machine Learning, Proceedings of the Twenty-Second International Conference (ICML 2005), Bonn, Germany, August 7-11, 2005*, pages 513–520, 2005. doi: 10.1145/1102351.1102416. URL https://doi.org/10.1145/1102351.1102416.

Ioannis E. Livieris, Vassilis Tampakas, and Panayiotis E. Pintelas. A descent hybrid conjugates gradient method based on the memoryless BFGS update. *Numerical Algorithms*, 79(4):1169–1185, 2018. doi: 10.1007/s11075-018-0479-1. URL https://doi.org/10.1007/s11075-018-0479-1.

Zhaosong Lu, Renato D. C. Monteiro, and Ming Yuan. Convex optimization methods for dimension reduction and coefficient estimation in multivariate linear regression. *Math. Program.*, 131(1-2):163–194, 2012. doi: 10.1007/s10107-010-0350-1. URL https://doi.org/10.1007/s10107-010-0350-1.

Dominique Deramaix Luis M. Candanedo, Veronique Feldheim. Uci machine learning repository. Appliances energy prediction, 2017. URL http://archive.ics.uci.edu/ml.

Liwen Ma, Jiaji Wu, and Chunyuan Li. Localization of a high-speed train using a speed model based on the gradient descent algorithm. *Future Generation Comp. Syst.*, 85:201–209, 2018. doi: 10.1016/j.future.2018. 03.041. URL https://doi.org/10.1016/j.future.2018.03.041.

Gabriella Melki, Alberto Cano, Vojislav Kecman, and Sebastián Ventura. Multi-target support vector regression via correlation regressor chains. *Inf. Sci.*, 415:53–69, 2017. doi: 10.1016/j.ins.2017.06.017. URL https://doi.org/10.1016/j.ins.2017.06.017.

Xiangrui Meng and Michael W. Mahoney. Robust regression on mapreduce. In *Proceedings of the 30th Interna tional Conference on Machine Learning, ICML 2013, Atlanta, GA, USA, 16-21 June 2013*, pages 888–896, 2013. URL http://jmlr.org/proceedings/papers/v28/meng13b.html.

Nuno Moniz and LuaÂ sTorgo. Uci machine learning repository. news popularity in multiple social media platform, 2018. URL http://archive.ics.uci.edu/ml.

Yashuang Mu, Lidong Wang, and Xiaodong Liu. A fast rank mutual informationbased decision tree and its implementation via map-reduce. *Concurrency and Computation: Practice and Experience*, 30(10), 2018. doi: 10.1002/cpe.4387. URL https://doi.org/10.1002/cpe.4387.

Maryam M. Najafabadi, Taghi M. Khoshgoftaar, Flavio Villanustre, and John Holt. Large-scale distributed l-bfgs. *Journal of Big Data*, 4(1):22, 2017. ISSN 2196-1115. doi: 10.1186/s40537-017-0084-5. URL https://doi.org/10.1186/s40537-017-0084-5.

Joe Naoum-Sawaya. *Interior Point Cutting Plane Methods in Integer Programming*. PhD thesis, University of Waterloo, Ontario, Canada, 2011. URL http://hdl.handle.net/10012/6105.

Aljaz Osojnik, Pance Panov, and Saso Dzeroski. Multi-label classification via multi-target regression on data

streams. *Machine Learning*, 106(6):745–770, 2017. doi: 10.1007/s10994-016-5613-5. URL https://doi.org/10.1007/s10994-016-5613-5.

David Pardoe and Peter Stone. The 2007 tac scm prediction challenge. In *Agent-Mediated Electronic Commerce and Trading Agent Design and Analysis*, pages 175–189. Springer, 2010.

C. S. R. Prabhu. *Fog Computing, Deep Learning and Big Data Analytics-Research Directions*. Sprin ger, 2019. ISBN 978-981-13-3208-1. doi: 10.1007/978-981-13-3209-8. URL https://doi.org/10.1007/978-981-13-3209-8.

Michael Redmond. Uci machine learning repository. communities and crime unnormalized, 2017. URL http://archive.ics.uci.edu/ml.

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. ISSN 1061-8600. doi: 10.1198/jcgs. 2010.09188. URL https://doi.org/10.1198/jcgs.2010.09188.

Faridah Hani Mohamed Salleh, Suhaila Zainudin, and Shereena Mohd Arif. Multiple linear regression for reconstruction of gene regulatory networks in solving cascade error problems. *Adv. Bioinformatics*, 2017: 4827171:1–4827171:14, 2017. doi: 10.1155/2017/4827171. URL https://doi.org/10.1155/2017/4827171.

Muhammad Noman Shafique, Muhammad Mahboob Khurshid, Haji Rahman, Ashish Khanna, and Deepak Gupta. The role of big data predictive analytics and radio frequency identification in the pharmaceutical industry. *IEEE Access*, 7:9013–9021, 2019. doi: 10.1109/ACCESS.2018.2890551. URL https://doi.org/10.1109/ACCESS.2018.2890551.

Markiel A. Ozier O. Baliga-N.S. Wang J.T. Ramage D. Amin N. Schwikowski B. Shannon, P. and T. Ideker. Uci machine learning repository. kegg metabolic reaction network (undirected), 2011. URL http://archive.ics.uci.edu/ml.

Vinu Sherimon and Sherimon Puliprathu Cherian. Building a multiple linear regression model to predict students'marks in a blended learning environment. In *Interactive Mobile Communication Technologies and Learning - Proceedings of the 11th IMCL Conference, 30 November - 1 December 2017, Mediterranean Palace Hotel, Thessaloniki, Greece*, pages 903–911, 2017. doi: 10.1007/978-3-319-75175-7_88. URL https://doi. org/10.1007/978-3-319-75175-7_88.

C. P. Sona and Jaison Paul Mulerikkal. Performance comparison of distributed pattern matching algorithms on hadoop mapreduce framework. In *Mobile Networks and Management - 9th International Conference, MONAMI 2017, Melbourne, Australia, December 13-15, 2017, Proceedings*, pages 45–55, 2017. doi: 10.1007/ 978-3-319-90775-8_4. URL https://doi.org/10.1007/978-3-319-90775-8_4.

Hernando Díaz Sonia E. Monroy Varela, editor. *Modelo de predicción de gravedad de accidentes de tránsito: un análisis de los siniestros en Bogotá, Colombia*, 2018. VI Congreso Ibero-Americano de Seguridad Vial. URL https://vicisev.institutoivia.org/wp-content/uploads/2018/11/edwin-urbano-CISEV-Articulo_gravedad_accidentes-VRFNL.pdf.

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.

Dzeroski Struyf, J. Curve prediction with kernel regression. in: Proceedings of the ecml/pkdd 2009 workshop on learning from multi-label data. pages 61–68, 2009.

Jan Struyf and Saso Dzeroski. Constraint based induction of multi-objective regression trees, 2005. URL https://doi.org/10.1007/11733492_13.

V. Tejasviram, H. Solanki, V. Ravi, and Sk. Kamaruddin. Auto associative extreme learning machine based non-linear principal component regression for big data applications. In *Tenth International Conference on Digital Information Management, ICDIM 2015, Jeju Island, South Korea, October 21-23, 2015*, pages 223-228, 2015. doi: 10.1109/ICDIM.2015.7381854. URL https://doi.org/10.1109/ICDIM.2015.7381854.

Athanasios Tsanas and Angeliki Xifara. Accurate quantitative estimation of energy performance of residential buildings using statistical machine learning tools. *Energy and Buildings*, 49:560–567, 2012.

Grigorios Tsoumakas, Eleftherios Spyromitros Xioufis, Aikaterini Vrekou, and Ioannis P. Vlahavas. Multi target regression via random linear target combinations, 2014. URL https://doi.org/10.1007/978-3-662-44845-8_15.

Sattar Vakili and Qing Zhao. A random walk approach to first-order stochastic convex optimization. *CoRR*, abs/1901.05947, 2019. URL http://arxiv.org/abs/1901.05947.

Ashley Vance. Start-up goes after big data with hadoop helper, 2010.

Tomás Darío Marín Velásquez. Modelo matemático para la predicción de la viscosidad de crudos pesados muertos producidos en el estado monagas, venezuela. *Enfoque UTE*, 8(3):16–27, 2017. ISSN 1390â6542. URL http://ingenieria.ute.edu.ec/enfoqueute.

Willem Waegeman, Krzysztof Dembczynski, and Eyke Hüllermeier. Multi-target prediction: a unifying view on problems and methods. *Data Min. Knowl. Discov.*, 33(2):293–324, 2019. doi: 10.1007/s10618-018-0595-5. URL https://doi.org/10.1007/s10618-018-0595-5.

Baobin Wang and Ting Hu. Distributed pairwise algorithms with gradient descent methods. *Neurocomputing*, 333:364–373, 2019. doi: 10.1016/j.neucom.2019.01.007. URL https://doi.org/10.1016/j.neucom.2019.01.007.

Eleftherios Spyromitros Xioufis, William Groves, Grigorios Tsoumakas, and Ioannis P. Vlahavas. Multi-label classification methods for multi-target regression. *CoRR*, abs/1211.6581, 2012. URL http://arxiv.org/abs/1211.6581.

Donna Xu, Yaxin Shi, Ivor W. Tsang, Yew-Soon Ong, Chen Gong, and Xiaobo Shen. A survey on multi-output learning. *CoRR*, abs/1901.00248, 2019. URL http://arxiv.org/abs/1901.00248.

Lixiang Xu, Bin Luo, Yuanyan Tang, and Xiaohua Ma. An efficient multiple kernel learning in reproducing kernel hilbert spaces (RKHS). *IJWMIP*, 13(2), 2015. doi: 10.1142/S0219691315500083. URL https://doi.org/10.1142/S0219691315500083.

I-Cheng Yeh. Modeling slump flow of concrete using second-order regressions and artificial neural networks. *Cement and Concrete Composites*, 29(6):474–480, 2007.

Matei Zaharia, Mosharaf Chowdhury, Michael J. Franklin, Scott Shenker, and Ion Stoica. Spark: Cluster computing with working sets, 2010. URL https://www.usenix.org/conference/hotcloud-10/spark-cluster-computing-working-sets.

Yiteng Zhai, Yew-Soon Ong, and Ivor W. Tsang. The emerging? big dimensionality? *IEEE Comp. Int. Mag.*, 9(3):14–26, 2014. doi: 10.1109/MCI.2014.2326099. URL https://doi.org/10.1109/MCI.2014.2326099.

Jinjing Zhang, Fei Hu, Xiaofei Xu, and Li Li. Stochastic gradient descent with variance reduction technique. *Web Intelligence*, 16(3):187–194, 2018. doi: 10.3233/WEB-180386. URL https://doi.org/10.3233/WEB-180386.

Min-Ling Zhang and Zhi-Hua Zhou. A review on multi-label learning algorithms. *IEEE Trans. Knowl. Data Eng.*, 26(8):1819–1837, 2014. doi: 10.1109/TKDE.2013.39. URL https://doi.org/10.1109/TKDE.2013.39.

Tuo Zhao, Mo Yu, Yiming Wang, Raman Arora, and Han Liu. Accelerated mini-batch randomized block coordinate descent method. In *Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing Systems 2014, December 8-13 2014, Montreal, Quebec, Canada*, pages 3329–3337, 2014. URL http://papers.nips.cc/paper/5614-accelerated-mini-batch-randomized-block-coordinate-descent-method.

Xiantong Zhen, Mengyang Yu, Xiaofei He, and Shuo Li. Multi-target regression via robust low-rank learning. *IEEE Trans. Pattern Anal. Mach. Intell.*, 40(2):497–504, 2018a.doi: 10.1109/TPAMI.2017.2688363. URL https://doi.org/10.1109/TPAMI.2017.2688363http://doi.ieeecomputersociety.org/10.1109/TPAMI.2017.2688363.

Xiantong Zhen, Mengyang Yu, Feng Zheng, Ilanit Ben Nachum, Mousumi Bhaduri, David T. Laidley, and Shuo Li. Multitarget sparse latent regression. *IEEE Trans. Neural Netw. Learning Syst.*, 29(5):1575–1586, 2018b. doi: 10.1109/TNNLS.2017.2651068. URL https://doi.org/10.1109/TNNLS.2017.2651068.