IX Simposio de Teoría y Aplicaciones de la Minería de Datos (IX TAMIDA)

TAMIDA 1:
MODELOS PREDICTIVOS

Identifying ecosystem patterns from time series of anchovy (Engraulis ringens) and sardine (Sardinops Sagax) landings in northern Chile

1st Francisco Plaza
Unidad de Auditoría y Control
Instituto de Fomento Pesquero
Valparaiso, Chile
francisco.plaza.vega@ifop.cl

2nd Rodrigo Salas
Escuela de Ingeniería Biomédica
Universidad de Valparaíso
Valparaiso, Chile
rodrigo.salas@uv.cl

3rd Eleuterio Yáñez
Escuela de Ciencias del Mar
Pontificia Universidad Católica de Valparaíso
Valparaíso, Chile
eleuterio.yanez@pucv.cl

***Abstract*—This work presents a Knowledge Discovery from Data (KDD) approach in time series pattern identification for anchovy and sardine monthly fishery-biological data in northern Chile. Time series, multivariate analysis and data mining techniques, along with technical literature review for results validation are implemented. This approach, achieved an integration between variables, identifying relevant patterns, associated with fisheries abundance fluctuations and strong association with environmental changes such as El Niño and long-term cold-warm regimes between them, establishing predominant time-periods. The latter, establishes groundwork for studying underlying functional relationships that could reduce gaps in the national fisheries research and management policies.**

***Index Terms*—Anchovy, Sardine, Ecosystem, Patterns, Data Mining, Time Series, Knowledge Discovery from Data**

I. Introduction

The Chile-Peru Current System (CHPCS) is a very productive marine system due to nutrient transport by large-scale horizontal advection and persistent coastal upwelling. In Chile the average annual landing in the last 30 years is 4.8 million tons and the pelagic resources of the northern zone (18°21'S-24°S) represent 40% ([2]). In this area the fishery is successively based on anchovy (*Engraulis ringens*) and sardine (*Sardinops sagax*), with notable changes associated with fishing effort and fluctuations in the environment ([3]).

Ocean-climate changes alter marine ecosystems at various scales, in fact, [3] propose an integrative conceptual model of the main local to large-scale phenomena involved in northern Chile. On the intra-seasonal scale, coastal trapped waves are mainly responsible for most of the variability of sea surface temperature and currents on the continental shelf and slope of the CHPCS. On the interannual scale, changes in atmospheric and oceanographic conditions are mainly associated with the El Niño Southern Oscillation (ENSO) phenomenon. Interdecadal long-term regime shifts (warm or cold), would influence the reorganization of marine communities and trophodynamic relationships, inducing changes in dominant species. Thus, the link between the variation of anchovy abundance and environmental changes at different spatio-temporal scales opens the possibility for predicting fluctuations in landings in the short, medium, and long term, one of the main objectives of fisheries management ([4]; [5]). Therefore, the main objective for this work is to identify patterns of interaction between those variables in northern Chile.

II. Materials and methods

A. Data

The study analyzes monthly data considering local and global environmental variables and fishing variables from 1963-2011 in northern Chile (18°21'S-24°S and from the coast until 73°O). This area is where industrial purse seine fleet operates in northern Chile. Table I summarize all considered variables.

TABLE I
Summary of analyzed variables

Type	*Variable*	*Desctiption*
Local	SST	Sea Surface Temperature from Antofagasta coastal oceanographic station
	TI	Turbulence Index from Antofagasta coastal oceanographic station
	MSL	Mean Sea Level from Antofagasta coastal oceanographic station
Global	MEI	Multivariate ENSO Index
	PDO	Pacific Decadal Oscillation index
	N12	Climatic Index in the Niño12 area
	N34	Climatic Index in the Niño34 area
	SOI	Southern Oscillation Index
	CTI	Cold Tongue Index
Fisheries	VANC	% days per month of anchovy fishing prohibition (fisheries ban)
	VSAR	% days per month of sardine fishing prohibition (fisheries ban)
	LANC	Anchovy landings in northern Chile
	LSAR	Sardine landings in northern Chile

B. General procedure

For this study, a similar approach to [6] is considered, involving as a first step an inspection of the multivariate time series, using signal decomposition (trend, seasonality and noise), then Principal Components Analysis is performed to reduce dimensionality and to discover preliminary patterns (which are contrasted with field literature). Finally, as a third step these patterns are grouped, using k-means clustering, in order to obtain different ecosystem periods, which allows to identify changing trends (increases or decreases) of anchovy and sardine landings, associated to environmental conditions. Integrated graphical representations are then implemented to show discovered patterns.

III. Results and discussion

Figure 1 shows the additive decomposition performed on the 4 PCs, showing the trend, seasonality and irregular component.

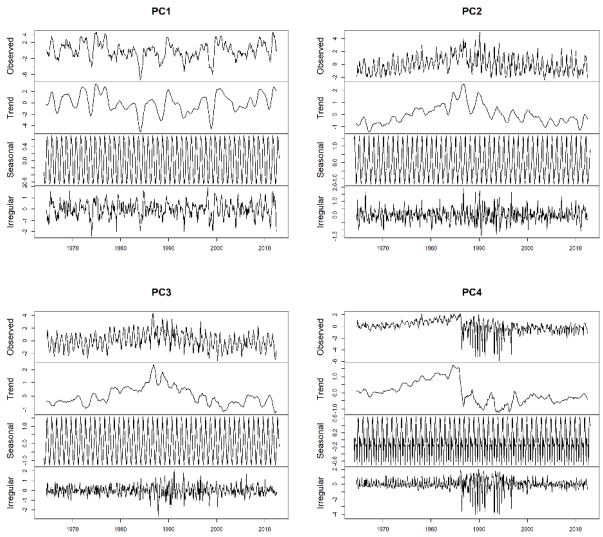


Fig. 1. Decomposition of the selected PCA components.

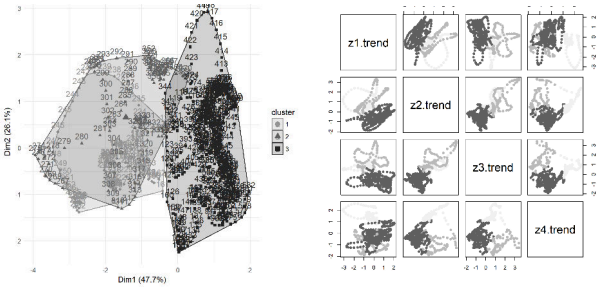


Fig. 2. Clustering performance and results. Representation of clusters centroids (left), classification of the principal components (right).

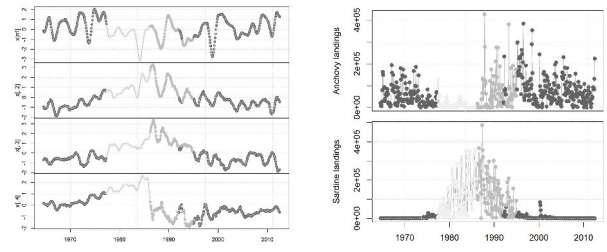


Fig. 3. Cluster classification applied to data. Cluster classification of PCA selected components (left), Cluster classification of anchovy and sardine landings time series (LANC and LSAR)(right).

In order to address the problem, a series of research questions were established to guide the analysis. To summarize the results obtained in this study, the answer for each research question is presented as follows:

1) *what is the relationship between the time series between sardine and anchovy?*. Results show an alternance between anchovy and sardine time series. The latter can be observed in both the time series of each fishery and in the results obtained from the clustering process (Fig. 3, having an anchovy period, then a transition period, followed by a sardine period, and finally an anchovy period, respectively.
2) *what is the relationship between environmental variables and the anchovy-sardine?*. As discussed before, the first 4 components from the PCA have higher correlation with environmental variables. Also, the clustering process identified 3 clusters which can be related to environmental long-term cold-warm-cold periods(Fig. 2). Furthermore, those long-term periods are related with anchovy and sardine landings presence or absence. The underlying processes that describe functional relationships between particular environmental variables and anchovy and sardine behavior are still being discussed among the scientific community. However, this study proposes a more integrated scope, considering the ecosystem as a whole.
3) *Are there any procedures that allow to identify patterns?*. A process of mixing three well-known techniques (time series decomposition, principal components analysis and clustering) was successfully implemented in order to identify ecosystemic patterns.
4) *Are there identifiable patterns in the time series of anchovy, sardine, and environmental variables?*. Four clear periods could be identified, from 1963-1977, 1977-1986, 1986-1995 and 1995-2011, having an anchovy period, then a transition period, followed by a sardine period, and finally an anchovy period, respectively (Fig. 3).

Acknowledgment

This paper is presented as an overview of the work done by [1].

References

References

[1] F. Plaza, R. Salas, and E. Yáñez, "Identifying ecosystem patterns from time series of anchovy (Engraulis ringens) and sardine (Sardinops sagax) landings in northern Chile," Journal of Statistical Computation and Simulation, vol.88(10), pp. 1863–1881.
[2] SERNAPESCA. Anuarios Estadísticos de Pesca. Servicio Nacional de Pesca y Acuicultura, Chile; 1950-2012.
[3] Yáñez E, Hormazábal S, Silva C, et al. Coupling between the environment and the pelagic resources exploited off northern Chile: ecosystem indicators and a conceptual model. Latin American journal of aquatic research. 2008 00; 36: 159–181.
[4] Zhou S, Smith ADM, Punt AE, et al. Ecosystem-based fisheries management requires a change to the selective fishing philosophy. Proceedings of the National Academy of Sciences. 2010;107(21):9485–9489.
[5] Chavez FP, Ryan J, Lluch-Cota SE, et al. From anchovies to sardines and back: multidecadal change in the pacific ocean. Science. 2003; 299(5604): 217–221.
[6] Fayyad U, Piatetsky-Shapiro G, Smyth P. From Data Mining to Knowledge Discovery in Databases. AI Magazine. 1996; 3(17): 37–54.

Análisis Big Data para la Respuesta a la Demanda en el Mercado Eléctrico

José Antonio Fábregas,
José María Luna-Romera
and José C. Riquelme Santos
Lenguajes y Sistemas Informáticos
Universidad de Sevilla
Sevilla, España

Ángel Arcos-Vargas
Organización Industrial y Gestión de Empresas
Universidad de Sevilla
Sevilla, España

Javier Tejedor Aguilera
Endesa S.A.
Madrid, España

***Abstract*—El modelo de negocio tradicional de las compañías energéticas está cambiando los últimos años. La introducción de los contadores inteligentes ha conllevado un aumento exponencial del volumen de datos disponibles, y su análisis puede ayudar a encontrar patrones de consumo entre los clientes eléctricos para reducir costes y proteger el medioambiente. Las centrales generan energía eléctrica para poder cubrir los picos de consumo en momentos puntuales. Un conjunto de técnicas denominadas "demand response" intenta dar solución a este problema usando propuestas de inteligencia artificial. En este documento se propone una metodología para el procesado de los grandes volúmenes de datos como los que generan los contadores inteligentes. Tanto para el preprocesado como para la optimización y realización de este análisis se utilizan técnicas big data. En concreto, una versión distribuida del algoritmo k-means y de varios índices de validación interna de clustering para big data en Spark. Los datos de origen corresponden los consumos de clientes eléctricos de Cataluña durante el año 2016. El análisis de estos consumidores realizado en este trabajo ayuda a su caracterización. Este mayor conocimiento sobre los hábitos de consumos y tipos de clientes, puede facilitar a las compañías eléctricas la labor.**

***Index Terms*—Big data, respuesta a la demanda, clustering, contadores inteligentes, consumo eléctrico**

I. Introducción

Durante la mayor parte del siglo XX, la relación entre los usuarios de electricidad y las empresas de distribución se mantuvo sin cambios. No se eligieron proveedores y, por lo tanto, no había necesidad de tratar a los consumidores como clientes. Sin embargo, la desregulación, la agenda verde y los continuos saltos tecnológicos han cambiado esta relación. Nuevas limitaciones como la seguridad del suministro, la competitividad y la sostenibilidad son los tres ejes prioritarios para cambiar el modelo energético actual, que pueden lograrse mediante objetivos como la reducción de las emisiones y la mejora de la generación de energía renovable y la eficiencia energética.

Una herramienta esencial en este nuevo modelo son los llamados "contadores inteligentes", que no deben entenderse sólo como dispositivos que miden el consumo sino como verdaderos sensores para una red eléctrica. Estos sensores facilitan una red altamente flexible y adaptable que integra de forma inteligente las acciones de los usuarios que se conectan a ella para conseguir un suministro eficiente, seguro y sostenible.

Uno de los principales problemas del sector eléctrico es la necesidad de disponer de capacidad de generación y de una red sobredimensionada para cubrir los picos de alto consumo de los clientes en determinados momentos. Sin embargo, existen soluciones basadas en la adaptación de la demanda a la energía disponible en lugar de aumentar la oferta para satisfacer la demanda. Esto se denomina "Respuesta a la Demanda" y su objetivo es cambiar los hábitos de consumo de electricidad de los clientes en respuesta a los cambios en los precios del suministro. El principal inconveniente es el gran volumen de información disponible en estas redes, ya que sólo puede manejarse con técnicas big data.

Nuestra propuesta se basa en el procesado de estos grandes conjuntos de datos de manera distribuida y paralela. En particular, la aplicación de técnicas de minería de datos para comprender mejor los patrones de consumo de los clientes. Por un lado, haremos uso de HDFS [1] para el almacenamiento de datos distribuido. Mientras que el procesado lo realizaremos con Spark [2], una plataforma de computación distribuida y paralela. En concreto, utilizaremos la implementación del algoritmo k-means de la librería MLlib [3] de Spark, así como cuatro índices de validación de clustering para big data [4]. Este estudio podría ayudar a la planificación de las conexiones de las fuentes de energía renovables a la red, con un doble objetivo: la reducción de precios y la sostenibilidad medioambiental.

La estructura de este artículo es la siguiente. En la sección II se describen los trabajos relacionados. En la sección III se detallan las características del dataset. En la sección IV se muestran los resultados de los experimentos realizados para preprocesar los datos y aplicar técnicas de clustering. Para finalizar, en la sección V se presentan las conclusiones de la investigación realizada.

II. Trabajo relacionados

La irregularidad de la demanda de electricidad es uno de los principales problemas del sector. Esto se debe a que las compañías eléctricas deben tener tanto una capacidad de generación sobredimensionada, como redundancia de la red para hacer frente a grandes cantidades de demanda que sólo se requieren unas pocas horas al año. Normalmente, se establece un umbral del 20% para la generación de electricidad latente, que debe cubrir aproximadamente el 5% del tiempo de servicio de la red (pico de demanda) [5]. Algunos de los recursos para resolver este problema necesitan la implicación de los usuarios. Estas soluciones se estudian bajo el nombre de 'respuesta a la demanda' (Demand Response DR) [6]. En contraste con las ideas convencionales de aumentar la oferta para satisfacer la demanda, las soluciones apuntan a satisfacerla con la energía disponible.

El objetivo es cambiar las pautas de consumo de energía de los clientes en respuesta a los cambios en los precios ofrecidos. Esto permitirá a las compañías eléctricas gestionar mejor la demanda con un mejor ajuste de las predicciones y una reducción del coste de la energía para los clientes. Existen múltiples iniciativas de posibles esquemas de fijación de precios, que en algunos casos incluso mantienen los beneficios para las empresas proveedoras [7]. Una de las principales ventajas de la respuesta a la demanda es ofrecer una opción sostenible con una generación de energía más volátil. Sobre todo en España, donde existe una alta presencia de fuentes de generación renovables. Para implementar los mecanismos de respuesta a la demanda, las redes eléctricas deben evolucionar a una infraestructura que permita el flujo de información entre los diferentes participantes del sistema eléctrico. En este campo es donde los grandes datos se convierten en una tecnología esencial para analizar este flujo de información y convertirlo en conocimiento útil.

Estos datos de consumo de los clientes, obtenidos mediante contadores inteligentes, no son sino múltiples series temporales. El análisis de series temporales puede entenderse como una secuencia de valores observados a lo largo del tiempo y ordenados cronológicamente [8]. Como el tiempo es una variable continua, las muestras se registran en puntos sucesivos igualmente espaciados. Por lo tanto, las series temporales son una secuencia de datos de tiempo discreto.

En el contexto de la minería de datos de series temporales, el principal desafío es cómo representar los datos. El enfoque más común es transformar las series temporales en otro ámbito para la reducción de la dimensionalidad y desarrollar un mecanismo de indexación. La medida de la similitud entre las sub-secuencias de series temporales y la segmentación son las dos tareas principales en la minería de series temporales que corresponden con las tareas clásicas de la minería de datos. El uso cada vez mayor de datos de series cronológicas ha dado lugar a una gran cantidad de intentos de investigación y desarrollo en el campo de la minería de datos [9].

En este trabajo nos centraremos en el clustering, un método de minería de datos para agrupar instancias no etiquetadas de conjuntos de datos. La idea es que las instancias recogidas en un mismo grupo tendrán un comportamiento similar [10]. El clustering de series temporales surge como un enfoque útil para minar patrones comunes a partir de datos dependientes del tiempo [11] que se caracterizan por tener una alta dimensionalidad y un gran tamaño.

Centrándonos en el clustering a partir de datos de consumo de energía, son muchas las propuestas enmarcadas en este campo: En [12] se presenta el efecto de las medidas de similitud en la aplicación de la agrupación para descubrir los patrones energéticos de los edificios. Para obtener perfiles de carga típicos de los clientes, en [13] se propone un índice de estabilidad para elegir el algoritmo de agrupamiento que mejor se adapte a este problema de reconocimiento de patrones. Además, se propone otro índice de prioridad (basado en el índice de estabilidad) para determinar el rango de prioridad de los agrupamientos. En [14] desarrolla una técnica de clustering de particiones para extraer información útil de los precios de la electricidad. Mientras que en [15] se usan técnicas de clustering con el objetivo de agrupar y etiquetar las muestras de un conjunto de datos para pronosticar el comportamiento de las series temporales basadas en la similitud de las secuencias de patrones.

En relación a la gestión inteligente de la demanda de electricidad, en [16] los autores proponen un Virtual Power Player como gestor para satisfacer la demanda y reserva de energía eléctrica requerida. En [17] se presenta un análisis de los datos de los contadores inteligentes de los clientes para comprender mejor la demanda máxima y las principales fuentes de variabilidad en su comportamiento.

Además de los métodos clásicos de gestión de datos, el enfoque de big data ha surgido recientemente debido a la disponibilidad de una gran cantidad de datos, sistemas de ficheros distribuidos, y potentes motores de procesamiento distribuido. Esto ha propiciado que muchos de los algoritmos de minería de datos se hayan adaptado al entorno de big data, como por ejemplo los algoritmos de clustering. En cuanto al campo del consumo de energía, en la actualidad han surgido varias grandes soluciones de datos, como las optimizaciones de redes inteligentes en [18] y los patrones de consumo energético en [19].

III. Características del dataset

Los datos originales se encuentran almacenados en 34 tablas, divididas en cientos de archivos CSV. Estas tablas contienen una amplia información sobre consumos, tarifas, contadores, datos geográficos o personales de consumidores eléctricos de Endesa en Cataluña entre 2010 y 2016 para un tamaño total de 1,8 TB.

Los clientes en los que centramos este estudio son aquellos que poseen tarifas 2.0A y 20DHA. Ambas del mercado libre de Endesa para potencias contratadas menores a 10kW, donde se encuentran inmensa mayoría de hogares y pequeños locales. La tarifa 2.0A mantiene un precio fijo durante todo el año, mientras que la 20DHA tiene discriminación horaria de dos periodos. En esta última, los periodos punta y valle marcan dos precios distintos según la hora en la que se consuma la energía: punta de 12h a 22h en invierno y de 13h a 23h en verano; valle de 22h a 12h en invierno y de 23h a 13h en verano.

En la Tabla Ia se observa la distribución de los clientes según su tarifa. Debido a la gran variedad de potencias contratadas, La Tabla Ib muestra una distribución de los clientes por cada rango de potencia. Estos rangos están basados en los valores estándar de potencia que actualmente pueden contratarse al tener contadores inteligentes.

TABLE I: Distribución de clientes

(a) Por tarifa contratada

Tarifa	Clientes
2.0A	102,123
2DHA	6,614

(b) Por potencia contratada

Potencia(kW)	Clientes
[0.1-2.3)	8,972
[2.3-3.45)	21,969
[3.45-4.60)	38,578
[4.60-5.75)	22,328
[5.75-6.90)	8,772
[6.90-8.05)	3,134
[8.05-9.20)	4,946
[9.20-10.0)	82

IV. Análisis realizados

En esta sección se presentan los resultados de los análisis realizados. En el apartado IV-A se detalla el procesado de los datos En el apartado IV-B se analizan cuatro índices de validación de clustering obtener el k óptimo. En el apartado IV-C se muestran los clusters obtenidos con el algortimo k-means. Por último, en el apartado IV-D se realiza una evaluación de los resultados.

Para la realización de los experimentos se han utilizado los siguientes entornos:

- Cluster propio: 72 procesadores Intel Xeon E7-4820, 128 GB RAM y 8 TB de almacenamiento.
- Cluster de EMR (Elastic Map Reduce) de AWS: cinco instancias de m3.2xlarge con 16 procesadores Intel Xeon E5-2670 v2 (Ivy Bridge), 30 GB RAM y 2 SSDs DE 80 GB cada una.

A. Procesado de datos

El primer objetivo es conseguir un conjunto de datos que tenga sentido minar. Primero, almacenaremos la gran cantidad de datos de los que disponemos en un sistema de ficheros distribuido (HDFS) configurado en nuestro cluster. Para reducir el tamaño de los datos, los ficheros CSV se pasan a Parquet, un formato de datos orientado a columnas que los comprime y codifica. Una vez almacenados y formateados, todo el procesamiento de estos datos se realiza de forma distribuida y paralela con Spark. Debido a que estos datos también se procesarán con herramientas online de Amazon Web Service, se almacenan además en el S3, el sistema de almacenamiento en línea de Amazon. Posteriormente se estudian y seleccionan los atributos necesarios para construir un primer dataset. Las tablas procesadas se muestran a continuación (Tabla II):

TABLE II: Tablas procesadas

Tabla	Elementos (millones)	Tamaño (MB)
Clientes	20.6	716
Contratos	40.6	560
Maestro Contratos	33.1	666
Curvas de carga	2,094.44	340,992

Una vez construido este primer dataset, se seleccionan los usuarios con: una potencia contratada igual o inferior a 10kW, una tarifa 2.0A o 20DHA, y que tengan con todas las lecturas de consumo del año 2016. Por último, se descartan las instancias con valores nulos. Este dataset está compuesto por 47,829,235 instancias correspondientes a las 365 curvas de carga de 2016 de 131,039 clientes. De cada consumidor, tenemos 24 lecturas de consumo horario para cada una de sus 365 instancias.

Para construir las series temporales de 2016 es necesario transformar este dataset de instancias diarias en uno de instancias anuales .De esta forma, el nuevo dataset constaría de 131,039 instancias, una por cliente, con: 8760 (365x24) lecturas horarias de 2016, la cups (Código Universal del Punto de Suministro), la tarifa y la potencia contratadas.

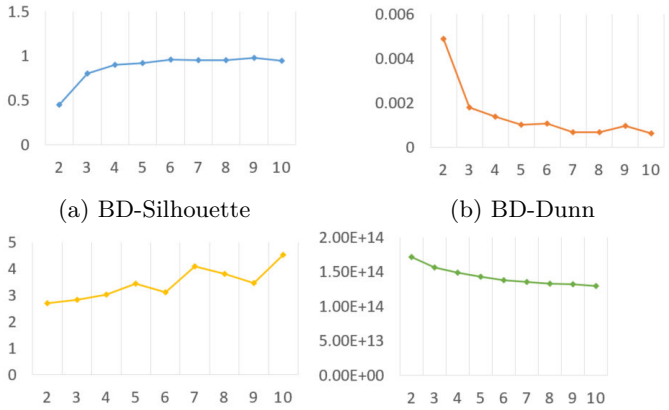
A continuación generamos un nuevo atributo con el que categorizamos a los consumidores en función a la potencia contratada. Por último, construimos un dataset alternativo con diferencias de consumo normalizadas: hallamos la diferencia entre cada par de valores de consumo consecutivo y la dividimos por la media de consumo de ese día. De esta forma tendremos un dataset con los consumos horarios de 2016 y otro con las diferencias normalizadas de consumo del mismo periodo. Estos datasets se utilizarán tanto de forma conjunta como individual en los siguientes apartados con el objetivo de encontrar patrones en los hábitos de consumo eléctrico de los clientes.

B. Determinación del número óptimo de clusters

Antes de aplicar algoritmos de clustering a nuestros datasets es necesario determinar cuál es el número óptimo de clusters (k) a obtener. Para ello, aplicamos a cada uno de los datasets cuatro índices de validación de clustering para big data (BD-CVIs) [4]: BD-Silhouette [4], BD-Dunn [4], Davies-Bouldin [20] y Within Set Sum of Square Errors (WSSSE) [21].

En la Figura 1a se muestra la representación gráfica resultados índice BD-Silhouette. Para este índice los valores óptimos de k son sus máximos, 6 y 9. Estos valores coincides con los máximos de la gráfica correspondiente al índice BD-Dunn (Figura 1b). En el caso del índice Davies-Bouldin los valores óptimos se encuentran en los mínimos,

que vuelven a coincidir en 6 y 9, tal y como se observa en la Figura 1c. Por último, los resultados del índice WSSSE representados en la Figura 1d no arrojan un valor claro. En este índice buscamos una estabilización de valores y, como podemos ver, no hay un valor concreto en lo que esto ocurra. Tras analizar estos resultados, hemos obtenido los valores 6 y 9 como óptimos para la realización del clustering.



(a) BD-Silhouette (b) BD-Dunn

(c) Davies-Bouldin (d) WSSSE

Fig. 1: Índices big data de validación de clustering

Al igual que para el dataset de comsumos, hemos vuelto a aplicar estos índices al dataset de diferencias normalizadas comentado al final del apartado IV-A. En este caso, los resultados para el valor óptimo de k fueron 5 y 7.

C. Clustering

Una vez calculado el número óptimo de clusters, aplicamos a cada dataset la versión implementada en Spark del algoritmo k-means. Esta implementación fue desarrollada para poder extraer patrones en sistemas paralelos y distribuidos. A la hora de ejecutar el algoritmo, le daremos como entrada el objeto RDD (Resilient Distributed Dataset) y el k obtenido anteriormente. Como resultado obtendremos una serie de clusters con elementos de cada uno de los datasets.

Para el dataset de consumos, dos de los clusters obtenidos con k=9 tenían menos de 5 elementos, por lo que se decidió trabajar con el otro valor óptimo obtenido de k=6. En el caso del dataset de diferencias normalizadas, dos de los clusters para k=5 contenían un único elemento. Por este motivo, se optó por tomar el valor k=7.

La distribución de los elementos en los 6 clusters del dataset de consumos se muestra en la Tabla IIIa. De la misma forma, en la Tabla IIIb podemos ver cómo se distribuyen los elementos correspondientes a los 7 clusters del dataset de diferencias normalizadas.

En la Figura 2 se representan las curvas de consumo horario formadas por los centroides de cada uno de los clusters durante una semana de enero de 2016. En ella destaca que la mayoría de los consumidores, agrupados

TABLE III: Clusters de los datasets para k óptimo

(a) Dataset de consumos para k=6

Cluster	Elementos
0	12,029
1	50,643
2	1002
3	138
4	1116
5	43,789

(b) Dataset de diferencias normalizadas para k=7

Cluster	Elementos
0	42,276
1	8241
2	2544
3	18,994
4	28,462
5	7191
6	1029

en los clusters 1 y 5, tienen consumos inferior a 1 kWh. También puede observarse como un grupo reducido de clientes, que conforma el cluster 3, tiene un consumo muy alto por la noche (de 19 a 8 horas) y prácticamente nulo durante el día.

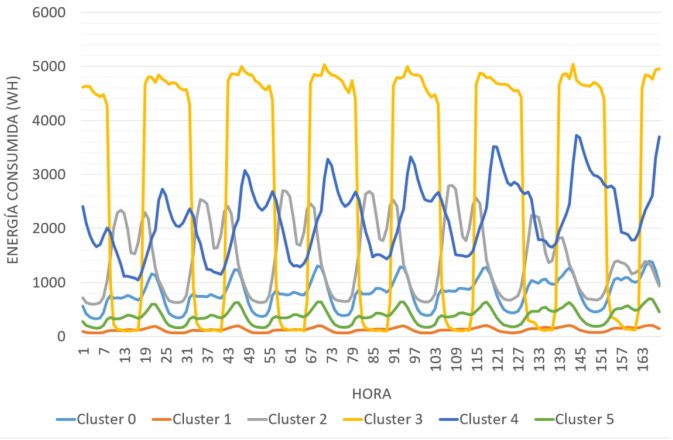


Fig. 2: Curvas de consumo horario de los centroides durante una semana de enero

En la Figura 3 se representan las las curvas de los mismos centroides durante una semana de julio. Se observa que los consumidores de los clusters 0, 1 y 5 mantienen un consumo prácticamente igual al de enero, aunque los picos máximo se acercan más a la media noche. Sin embargo, los del cluster descienden de forma radical hasta equipararse a los del 0. También destacan las curvas de consumo del cluster 3, donde las horas de alto consumo se reducen a 6 (00 a 6 horas).

Por su parte, la Figura 4 muestra las curvas de las diferencias horarias normalizadas en el mismo periodo de tiempo que la Figura 2. En ella podemos observar como los mayores picos de diferencias de consumo entre horas pertenecen a los elementos de los clusters 1,2,5 y 6. Estos clusters resultan ser los menos numerosos, representando un 17% del total de los consumidores. Lo que nos indica que el consumo de la mayoría de los usuarios a lo largo del día mantiene cierta uniformidad.

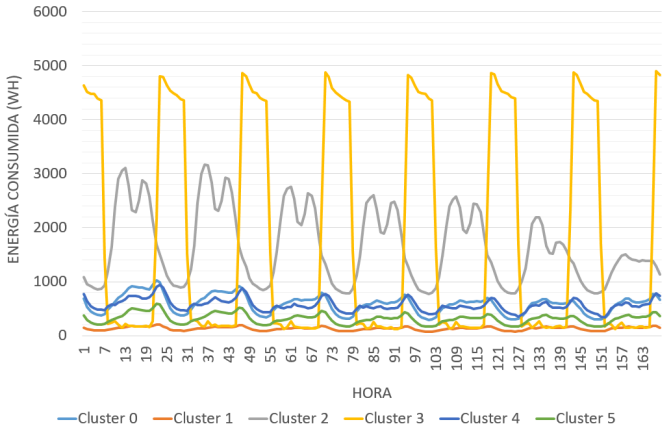


Fig. 3: Curvas de consumo horario de los centroides durante una semana de julio

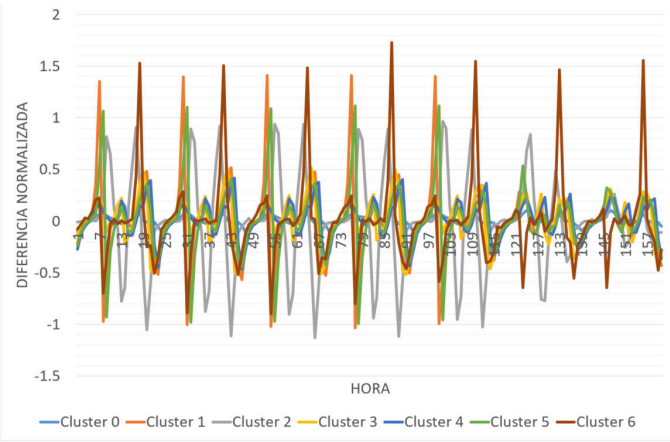


Fig. 4: Curvas de diferencias horarias normalizadas de los centroides

D. Evaluación de resultados

1) Evaluación no supervisada: En esta última fase evaluaremos los resultados obtenidos tras aplicar el algoritmo k-means a los datasets. Para ello, se generan las tablas de contingencia entre los clusters obtenidos para los distintos datasets, cruzando los resultados entre los mismos. Esto nos permite encontrar patrones que definan a los consumidores en relación a las características de los diferentes conjuntos de datos.

La tabla de la Figura 5 muestra por filas los 6 clusters del dataset de consumos (C0 a C5), y por columnas los 7 obtenidos del dataset de diferencias normalizadas (D0 a D6). Estos valores representan los porcentajes relativos al total de cada fila. Es decir, el porcentaje de consumidores de cada cluster de consumos presente en cada uno de los clusters de diferencias normalizadas.

En la Figura 5 destaca que casi la totalidad del cluster C3 está formado por consumidores del cluster D6. Si atendemos a la Figura 2 esto caracterizaría a un grupo de clientes con un consumo nocturno muy alto. Si además

		CLUSTER DIFERENCIAS NORMALIZADAS							
		D0	D1	D2	D3	D4	D5	D6	
CLUSTER CONSUMOS	C0	29.11%	10.20%	3.85%	19.03%	27.53%	9.85%	0.43%	*100.00%*
	C1	53.06%	5.25%	1.87%	13.03%	20.61%	4.74%	1.44%	*100.00%*
	C2	52.20%	1.60%	27.64%	7.98%	7.29%	3.19%	0.10%	*100.00%*
	C3	5.07%	0.00%	0.00%	0.00%	0.00%	0.72%	94.20%	*100.00%*
	C4	56.16%	7.83%	0.09%	7.13%	16.99%	8.36%	3.43%	*100.00%*
	C5	24.51%	9.70%	1.96%	22.71%	32.99%	7.94%	0.18%	*100.00%*

Fig. 5: Tabla de contingencia de valores relativos al total de cada fila

nos fijamos en la Figura 4, observamos que tienen un mínimo y un máximo en las diferencias horarias que se mantiene constante los 7 días de la semana. Esto podría indicar un perfil de consumidor muy concreto, con un consumo energético nocturno alto y uniforme. Por otro lado, podemos observar que la mitad cluster C1, el más numeroso, está formado por consumidores del cluster D0, también el más numeroso. Aquí se identifica un amplio grupo de usuarios de consumo bajo y con pocos cambios, cuyas mayores variaciones en el consumo se producen a las 8h.y 18h.

A continuación vamos a analizar el cluster C5, de un tamaño similar al C1 y con un consumo un poco más alto. En este caso, sólo se compone en un 24.51% de consumidores del cluster D0. Sin embargo, aumenta considerablemente el número de elementos pertenecientes a los clusters D3 y D4. En ellos, las diferencias de consumo horario (algo más altas que en el anterior) se concentran en las 19h. en los consumidores del cluster D3 y las 21 h. en los de D4. Por lo que podemos interpretar que los clientes con un consumo algo mayor que los del cluster C1 son más heterogéneos en cuanto a su comportamiento.

Ahora analizaremos los resultados desde la otra perspectiva. En la Figura 6 se representan los porcentajes relativos al total de cada columna. Es este caso, el porcentaje de consumidores de cada cluster de diferencias normalizadas presente en cada uno de los clusters de consumos.

		CLUSTER DIFERENCIAS NORMALIZADAS						
		D0	D1	D2	D3	D4	D5	D6
CLUSTER CONSUMOS	C0	8.28%	14.89%	18.20%	12.05%	11.63%	16.48%	5.05%
	C1	63.56%	32.29%	37.15%	34.74%	36.68%	33.39%	70.65%
	C2	1.24%	0.19%	10.89%	0.42%	0.26%	0.45%	0.10%
	C3	0.02%	0.00%	0.00%	0.00%	0.00%	0.01%	12.63%
	C4	1.51%	1.08%	0.04%	0.43%	0.68%	1.32%	3.79%
	C5	25.39%	51.55%	33.73%	52.36%	50.76%	48.35%	7.77%
		100.00%	*100.00%*	*100.00%*	*100.00%*	*100.00%*	*100.00%*	*100.00%*

Fig. 6: Tabla de contingencia de valores relativos al total de cada columna

Si nos fijamos en la Figura 6, lo primero que podemos destacar es que todos los clusters de diferencias normalizadas están compuestos mayoritariamente por los clusters C1 y C5. Algo normal debido entre los dos aglutinan el 86% del total de consumidores. Los clusters D0 y D1 tienen una mayoría de elementos de C1, mientras que los clusters D1, D3, D4 y D5 tienen una mayor presencia de elementos del C5. Por otro lado, en el D2 los clusters C1

y C5 no tienen una representación tan alta debido al 10% de elementos del cluster C2.

Centrándonos en el cluster D6, vemos que el 70.65% de sus clientes pertenecen al cluster C1. Mientras que observando la Figura 5 se denota que sólo el 1.44% de los del cluster C1 de consumos pertenecen al cluster D6.

2) Evaluación semi-supervisada: En este apartado tomaremos como referencia el rango de potencia contratada al que pertenece cada consumidor. Por lo que cruzamos elementos de los clusters del dataset de comsumo (Ver Tabla IIIa) con los de cada rango de potencia (ver Tabla Ib). El objetivo es encontrar relaciones entre los distintos tipos de consumidores y sus potencias contratadas. En la tabla de contingencia de la Figura 7 se representan los clusters del dataset de consumos por filas (C0 a C5) y los rangos de potencia por columnas (P1 a P8). Estos valores representan los porcentajes relativos al total de cada fila. Es decir, el porcentaje de consumidores de cada cluster de consumos presente en cada uno de los rangos de potencia.

		RANGO DE POTENCIA								
		P1	P2	P3	P4	P5	P6	P7	P8	
CLUSTER CONSUMOS	C0	2.65%	9.51%	27.46%	23.09%	17.67%	6.66%	12.80%	0.17%	*100.00%*
	C1	13.76%	26.37%	34.08%	17.74%	4.43%	1.50%	2.08%	0.03%	*100.00%*
	C2	2.69%	5.29%	12.28%	16.87%	14.87%	19.56%	26.85%	1.60%	*100.00%*
	C3	3.62%	0.72%	10.87%	21.74%	10.14%	47.10%	4.35%	1.45%	*100.00%*
	C4	4.15%	9.01%	12.54%	15.19%	26.50%	19.08%	12.90%	0.62%	*100.00%*
	C5	3.67%	16.70%	40.50%	23.28%	8.89%	2.50%	4.41%	0.05%	*100.00%*

Fig. 7: Tabla de contingencia de valores relativos al total de cada fila

Si observamos los clusters C1 y C5, los de menor consumo (ver Figura 2), comprobamos que la mayoría de sus elementos pertenece a los rangos P1 a P4, las potencias bajas. Mientras, en los clusters C2 y C4 aumentan de forma considerable los consumidores con potencias más altas (P5, P6 y P7). Como es lógico, la mayoría de los clientes con consumo bajo tienen contratadas potencias medias y bajas. Además, más de la mitad de los clientes con un consumo alto contrató una potencia alta. Un caso particular es el C3, donde casi la mitad de sus elementos pertenecen a P6. Esto indica que cerca del 50% de los clientes con alto consumo nocturno tienen contratados entre 6.90 y 8.05kW.

Ahora vamos a analizarlo desde el punto de vista de la potencia contratada. Si observamos la Figura 8, se representan los porcentajes relativos al total de cada columna. Es decir, el porcentaje de consumidores de cada rango de potencia presente en cada uno de los clusters de consumos.

Podemos ver como más del 85% de los consumidores de los rangos P1 a P4 pertenecen a los clusters C1 y C5. Por lo que la relación entre estos grupos de clusters y rangos de potencia existe en ambos sentidos: Clientes con bajo consumo contrató un nivel potencia baja o media y viceversa. En el análisis de la Figura 7 vimos como los clientes con alto consumo tenían potencias altas contratadas. Pero, si nos fijamos en los rangos P5 a P8,

		RANGO DE POTENCIA							
		P1	P2	P3	P4	P5	P6	P7	P8
CLUSTER CONSUMOS	C0	3.54%	5.20%	8.55%	12.42%	24.34%	25.53%	31.10%	24.39%
	C1	77.66%	60.80%	44.75%	40.25%	25.72%	24.31%	21.33%	15.85%
	C2	0.30%	0.24%	0.32%	0.76%	1.71%	6.25%	5.44%	19.51%
	C3	0.06%	0.00%	0.04%	0.13%	0.16%	2.07%	0.12%	2.44%
	C4	0.52%	0.46%	0.37%	0.77%	3.44%	6.89%	2.95%	8.54%
	C5	17.92%	33.28%	45.98%	45.66%	44.63%	34.94%	39.06%	29.27%
		100.00%	*100.00%*	*100.00%*	*100.00%*	*100.00%*	*100.00%*	*100.00%*	*100.00%*

Fig. 8: Tabla de contingencia de valores relativos al total de cada columna

observamos que éstos también están compuestos entre un 45% y un 70% por clientes con bajo consumo. Por lo que, aunque los clientes con alto consumo tienen potencias altas contratadas, la mayoría de consumidores con estas potencias tienen un bajo consumo. Al igual que antes, nos vamos a centrar el caso de P6 y C3. Mientras que el 47.1% de los consumidores de C3 pertenecían a P6, sólo el 2.07% de los de P6 se encuentran en C3. Esto refuerza la conclusión anterior, ya que más del 85% de los consumidores con potencia entre 6.9 y 8.05kW tienen consumos que rondan entre los 0.5 y 1.5kWh.

V. Conclusiones y trabajo futuro

En este trabajo se ha presentado la aplicación de técnicas big data para el análisis de datos de consumidores eléctricos. La caracterización de estos clientes obtenida como resultado da lugar a las siguientes conclusiones:

- Más del 85% de los clientes presentan curvas de carga de consumo donde los valores máximos no superan el kWh.
- Existe grupo de 138 clientes con un consumo nocturno muy alto. Y, aunque la cantidad de horas de consumo durante el verano es muy inferior al invierno, siempre se producen en periodos valle. El 97.8% de estos usuarios tienen contratada una tarifa adaptada a sus consumos, con discriminación horaria.
- En su inmensa mayoría, los clientes con un bajo consumo tenían contratada una baja potencia y viceversa.
- Los usuarios con un consumo medio-alto contrataron una potencia media-alta. Sin embargo, el 77.32% de los clientes que contrataron estos niveles de potencia, consumió valores de energía que no llegaron a 1kWh. Por lo que más de 3/4 partes de estos clientes tienen potencias contratadas muy por encima de lo que necesitan.
- Durante todo el año, los picos de consumo que se alcanzan por las mañanas y al mediodía se producen en horarios valle. Lo mismo ocurre con los picos nocturnos en verano, ya que estos aparecen entre las 23 y la 1. Esto indica que las tarifas de discriminación horaria podrían ser beneficiosas para los clientes con estos hábitos de consumo. Sin embargo, sólo el 5% (4,784 de 95,569 clientes) tienen contratada este tipo de tarifa.

- En trabajos futuros se caracterizará a los consumidores en función de sus consumos y tarifas. Además, se analizarán y recomendarán las tarifas y potencias óptimas a contratar de forma personalizada para cada tipo de cliente.

References

[1] K. Shvachko, H. Kuang, S. Radia, and R. Chansler, "The hadoop distributed file system," in *2010 IEEE 26th Symposium on Mass Storage Systems and Technologies (MSST)*, May 2010, pp. 1–10.

[2] M. Zaharia, M. Chowdhury, M. J. Franklin, S. Shenker, and I. Stoica, "Spark: Cluster computing with working sets," vol. 10, pp. 10–10, 07 2010.

[3] X. Meng, J. Bradley, B. Yavuz, E. Sparks, S. Venkataraman, D. Liu, J. Freeman, D. Tsai, M. Amde, S. Owen, D. Xin, R. Xin, M. J. Franklin, R. Zadeh, M. Zaharia, and A. Talwalkar, "Mllib: Machine learning in apache spark," *Journal of Machine Learning Research*, vol. 17, no. 34, pp. 1–7, 2016.

[4] J. M. Luna-Romera, J. García-Gutiérrez, M. Martínez-Ballesteros, and J. C. Riquelme Santos, "An approach to validity indices for clustering techniques in big data," *Progress in Artificial Intelligence*, vol. 7, no. 2, pp. 81–94, Jun 2018.

[5] H. T. Haider, O. H. See, and W. Elmenreich, "A review of residential demand response of smart grid," *Renewable and Sustainable Energy Reviews*, vol. 59, pp. 166 – 178, 2016.

[6] I. Hussain, S. Mohsin, A. Basit, Z. A. Khan, U. Qasim, and N. Javaid, "A review on demand response: Pricing, optimization, and appliance scheduling," *Procedia Computer Science*, vol. 52, pp. 843 – 850, 2015.

[7] H. T. Haider, O. H. See, and W. Elmenreich, "Residential demand response scheme based on adaptive consumption level pricing," *Energy*, vol. 113, pp. 301 – 308, 2016.

[8] F. Martínez-Álvarez, A. Troncoso, G. Asencio-Cortés, and J. C. Riquelme, "A survey on data mining techniques applied to electricity-related time series forecasting," *Energies*, vol. 8, no. 11, pp. 13 162–13 193, 2015.

[9] T. chung Fu, "A review on time series data mining," *Engineering Applications of Artificial Intelligence*, vol. 24, no. 1, pp. 164 – 181, 2011.

[10] S. Aghabozorgi, A. S. Shirkhorshidi, and T. Y. Wah, "Time-series clustering – a decade review," *Information Systems*, vol. 53, pp. 16 – 38, 2015.

[11] H. Wang, W. Wang, J. Yang, and P. Yu, "Clustering by pattern similarity in large data sets," vol. 3, 10 2002.

[12] F. Iglesias and W. Kastner, "Analysis of similarity measures in times series clustering for the discovery of building energy patterns," *Energies*, vol. 6, no. 2, pp. 579–597, 2013.

[13] T. Zhang, G. Zhang, J. Lu, X. Feng, and W. Yang, "A new index and classification approach for load pattern analysis of large electricity customers," *IEEE Transactions on Power Systems*, vol. 27, no. 1, pp. 153–160, Feb 2012.

[14] F. Martínez-Álvarez, A. Troncoso, J. C. Riquelme, and J. M. Riquelme, "Partitioning-clustering techniques applied to the electricity price time series," in *Proceedings of the 8th International Conference on Intelligent Data Engineering and Automated Learning*, ser. IDEAL'07. Berlin, Heidelberg: Springer-Verlag, 2007, pp. 990–999.

[15] F. M. Alvarez, A. Troncoso, J. C. Riquelme, and J. S. A. Ruiz, "Energy time series forecasting based on pattern sequence similarity," *IEEE Transactions on Knowledge and Data Engineering*, vol. 23, no. 8, pp. 1230–1243, Aug 2011.

[16] P. Faria, Z. Vale, and J. Baptista, "Demand response programs design and use considering intensive penetration of distributed generation," *Energies*, vol. 8, no. 6, pp. 6230–6246, 2015.

[17] S. Haben, C. Singleton, and P. Grindrod, "Analysis and clustering of residential customers energy behavioral demand using smart meter data," *IEEE Transactions on Smart Grid*, vol. 7, no. 1, pp. 136–144, Jan 2016.

[18] J.-S. Chou and N.-T. Ngo, "Smart grid data analytics framework for increasing energy savings in residential buildings," *Automation in Construction*, vol. 72, pp. 247 – 257, 2016.

[19] R. Perez-Chacon, R. L. Talavera-Llames, F. Martinez-Alvarez, and A. Troncoso, "Finding electric energy consumption patterns in big time series data," in *Distributed Computing and Artificial Intelligence, 13th International Conference*, S. Omatu, A. Semalat, G. Bocewicz, P. Sitek, I. E. Nielsen, J. A. García García, and J. Bajo, Eds. Cham: Springer International Publishing, 2016, pp. 231–238.

[20] D. L. Davies and D. Bouldin, "A cluster separation measure," vol. PAMI-1, pp. 224 – 227, 05 1979.

[21] "Spark clustering rdd based api documentation for spark 2.3.0. 2017." https://spark.apache.org/docs/2.3.0/mllib-clustering.html, accessed: 2018-06-11.

Un procedimiento efectivo para descomponer y modelar series temporales en agricultura

Francisco Aragón, Francisco Javier Baldán, Manuel Parra, José M. Benítez
Depto. Ciencias de la Computación e Inteligencia Artificial, DICITS, DaSCI
Universidad de Granada
Granada, España
{far, fjbaldan, manuelparra, J.M.Benitez}@decsai.ugr.es

***Resumen*—En este trabajo proponemos una forma innovadora de abordar casos reales de predicción de la producción de cultivos en una cooperativa. Nuestro enfoque consiste en la descomposición de la serie temporal original de los cultivos en sub-series temporales según una serie de factores, con el objetivo de generar un modelo predictivo del cultivo a partir de los modelos predictivos parciales de las sub-series. El ajuste de los modelos se realiza mediante un conjunto de técnicas estadísticas y de Aprendizaje Automático. Esta metodología se ha comparado con una metodología intuitiva que consiste en una predicción directa de las series temporales. Los resultados muestran que nuestro enfoque logra un mejor rendimiento de predicción que la manera directa, por lo que aplicar una metodología de descomposición es más adecuada para este problema que la no descomposición.**

***Palabras Clave*—agricultura, predicción, modelos predictivos, series temporales, descomposición**

I. Introducción

La agricultura es una actividad económica que resulta de vital importancia en prácticamente todos los países del mundo. En los últimos años la mejora del rendimiento agrícola (cantidad de producción obtenida por área cultivada) había sido propiciada por avances en la maquinaria empleada, nuevas técnicas de siembra, así como mediante la mejora de semillas y agroquímicos o un mejor control de plagas y enfermedades. Pero ahora, la agricultura comercial se ha vuelto una actividad de alta tecnología, en la que los avances tecnológicos informáticos también tienen su aplicación ya que permiten generar información de alta calidad sobre los procesos productivos.

Por lo tanto, la agricultura puede beneficiarse del auge de las técnicas englobadas dentro del ámbito de la Inteligencia Artificial y del Análisis de Datos, para crear modelos predictivos que predigan situaciones futuras que sean de ayuda para mejorar tanto la productividad de las cosechas como la toma de decisiones en todo lo relativo a ellas. Dentro del ámbito de la agricultura una gran cantidad de procesos tales como la producción o la aparición de plagas entre otras, pueden modelarse como una colección de observaciones habitualmente ordenadas a lo largo de un periodo de tiempo, es decir, mediante una serie temporal. Por medio de la aplicación sobre las series temporales de un conjunto de técnicas estadísticas y de Aprendizaje Automático se pueden generar modelos predictivos que sean capaces de extraer sus regularidades y de realizar predicciones.

Estos modelos predictivos resultantes de los análisis de los datos, pueden ser de gran importancia en las cooperativas y asociaciones de las que dependen los pequeños agricultores a la hora de la toma de decisiones en una gran cantidad de situaciones cotidianas relacionadas por ejemplo con la gestión de plagas, de los cultivos en el campo, o la gestión de la producción esperada ya sea para ofertarla en el mercado, para contratar el personal necesario para tratarla o para obtener el material necesario para su preparación y embalaje. La eficiencia en todas estas situaciones va a depender de la capacidad de transformar los datos brutos en información precisa que permita tomar la decisión acertada en cada una de ellas, siendo la gestión de la producción una de las actividades de mayor importancia. Por ello, disponer de información fiable sobre las expectativas de producción es crítico en el sector agrícola.

En este artículo se propone descomponer las series temporales del rendimiento de un cultivo en un conjunto de sub-series temporales en función de una serie de factores, siendo clave la semana de plantación, con la intención de que con la descomposición se obtengan sub-series con patrones más establecidos que la serie original general. El análisis y ajuste de los modelos predictivos de las sub-series temporales se realiza mediante una batería de métodos estadísticos como son los métodos ARIMA [3] y de Aprendizaje Automático como son las redes neuronales entre otras, para a partir de ellos, realizar una predicción general de los kilogramos del cultivo. La metodología propuesta se ha evaluado en varios casos de estudio reales, y se ha comparado con una metodología directa que consiste en una predicción de las series temporales sin descomponer. Los resultados muestran como se obtiene una mejora de la predicción al aplicar la metodología planteada con respecto a la serie sin descomponer.

El resto del artículo está organizado de la siguiente manera: la sección 2 describe el estado actual en que se encuentra la aplicación de modelos predictivos sobre problemas de predicción de cosechas modelados como series temporales. La sección 3 describe el problema, la forma en que ha llevado a cabo la descomposición, y la metodología empleada para resolverlo. En la sección 4 se detalla la experimentación realizada, se muestra el conjunto de datos y los resultados

obtenidos. En la sección 5 se discuten los resultados obtenidos. Por último, la sección 6 presenta las conclusiones obtenidas tras la realización del trabajo.

II. Estado del arte

En el ámbito de la agricultura, los investigadores han propuesto numerosos modelos y procedimientos para mejorar la predicción de cosechas. La mayoría de ellos han tenido un enfoque multivariante, en los que se trata de incorporar a la predicción el impacto que numerosas variables relativas a cultivos tales como la lluvia, la temperatura, los fertilizantes o la calidad del suelo tienen en la cosecha. Los métodos más ampliamente empleados para realizar este tipo de modelos multivariantes son las redes neuronales, más en concreto en su variante *feed-fordward*. En [10] y [12] se desarrollan estudios sobre si redes neuronales con variables relacionadas al cultivo son adecuadas para realizar la predicciones futuras, realizan una evaluación de la capacidad predictiva de diferentes parametrizaciones de la red, y comparan los resultados con otros modelos de regresión. En otros trabajos como en [14] se ha realizado un estudio concreto de las variables óptimas para la predicción de cosecha de maíz. Existe un déficit de trabajos en cuanto a la predicción de cosechas que tengan un enfoque multivariante y que además empleen métodos distintos a redes neuronales. Algunos de esos trabajos son [6] y [9], que demuestran la capacidad de métodos de *Random Forests (RF)* para la estimación de cultivos, y en [4] para máquinas de soporte vectorial. En [16] se realiza una comparación entre diferentes metodos regresivos como redes neuronales, funciones de base radial o árboles de regresión.

En cuanto a un enfoque univariante, los métodos ARIMA son los más ampliamente usados en la predicción de cosechas. En [15] se emplean los métodos ARIMA para predecir el área y la producción de trigo en los próximos años. También los alisados exponenciales los acompañan en algunos trabajos como en [1]. En [5] se realiza una comparación sobre la capacidad predictiva entre los métodos ARIMA y los alisados exponenciales.

La mayoría de los trabajos que se han realizado en este ámbito se han realizado desde un punto de vista multivariante, centrando el estudio en la importancia del impacto de los factores endógenos sobre la variable a predecir. Las redes neuronales y los métodos ARIMA son los más empleados, siendo también aplicados aunque en menor medida las máquinas de soporte vectorial o los métodos *Random Forest*. Son menos los trabajos en los que se ha empleado un enfoque univariante que trabaje directamente sobre la serie temporal. Aún así, los trabajos tratan de una manera muy superficial a la serie temporal ya que no tratan de adaptarla o modificarla para obtener una serie más fácil de predecir, sino que se centran directamente en aplicar un método concreto, con diferentes entradas o diferentes parametrizaciones para obtener una mejor predicción. Además, los trabajos solo emplean un método para el análisis de la serie, obviando la posibilidad de combinar varios métodos sobre la misma serie mediante una subdivisión de la misma.

III. Propuesta

Para hacer una declaración más precisa se formaliza el problema de predicción abordado, por lo que a continuación se detalla nuestra propuesta.

III-A. Descripción del problema y formulación

El problema se ubica en el ámbito de una cooperativa de frutas y hortalizas. La cooperativa está compuesta de un conjunto de parcelas, pertenecientes a los agricultores asociados, y la producción se divide en campañas, que representan los años agrícolas, de septiembre a julio. Cada parcela planta un producto específico durante una campaña. La superficie de cultivo de las parcelas se mide en metros cuadrados y cada parcela tiene su propia superficie. Los productos que se plantan en la cooperativa son diferentes variedades de frutas y verduras y su producción se mide en kilogramos. La producción total de un producto está formada por la suma de la producción de cada parcela que planta el producto. Algunos de los productos se plantan durante un cierto periodo de tiempo, siguiendo un patrón de siembra similar en cada campaña, mientras que otros se plantan de forma irregular a lo largo de la campaña. La producción durante el ciclo de vida de un cultivo suele ser similar para el mismo producto, por lo que para los productos plantados en periodos cercanos la producción sigue un patrón similar.

El principal interés de la cooperativa reside en conocer la producción en unidades de peso (p.e. kilogramos) que producirá cada producto para la próxima semana, de manera que la cooperativa pueda manejar adecuadamente el volumen de producción que se tendrá. Por lo tanto, la mejor manera de medir la producción es de forma semanal, y el problema a tratar consiste en la predicción de la producción en kilogramos de producto para la próxima semana. Este problema se abordará descomponiendo las series temporales de producción en sub-series temporales de semanas de siembra significativas, y empleando sobre ellas técnicas de Inteligencia Artificial y métodos estadísticos para generar modelos predictivos cuyas predicciones se irán agregando para obtener finalmente la producción global.

III-B. Descomposición de las series temporales

La idea principal del enfoque de descomposición en sub-series temporales es capturar el comportamiento similar que debe tener la producción de un producto que ha sido plantado en el mismo corto periodo de tiempo, ya que su ciclo de vida será muy parecido. Por lo tanto, el principal criterio de descomposición está tomado de uno de los factores más influyentes en la producción: la fecha de siembra. Las fechas de plantación se agrupan por semanas. La descomposición se realiza en semanas de siembra significativas y habrá tantas sub-series temporales como semanas de plantación significativas se establezcan. Las semanas significativas son aquellas que presentan un gran número de parcelas en plantación y por tanto la mayor parte de la producción del producto. Para determinar el conjunto de semanas significativas, se computan

las frecuencias de las semanas de siembra durante las campañas anteriores para analizar el número de parcelas que han plantado el producto en cada semana, y por lo tanto elegir las semanas con una mayor frecuencia de parcelas en plantación. Las semanas cuyas frecuencias de siembra son pequeñas se agrupan en una o dos series temporales adicionales.

Para los casos particulares bajo estudio, la descomposición se hizo como se expresa en la Tabla I. La Tabla I revela la elección de semanas significativas para los productos, y por lo tanto el número de sub-series temporales. El producto 1 tiene tres importantes semanas consecutivas de plantación; 37, 38 y 39 que cubren el 70 % de las parcelas. Para el producto 2 las semanas elegidas presentan cierta dispersión ya que tiene sus semanas significativas entre la 8 y la 11 y entre la 31 y la 38, representando el 67 % del total de parcelas. En general, para ambos productos predomina la plantación en las semanas correspondientes a los meses de agosto y septiembre.

Tabla I
Semanas de plantación seleccionadas para cada sub-serie

Nº de la subserie	Semanas de plantación de la subserie	
	Producto 1	**Producto 2**
Sub1	37	8-9
Sub2	38	10-11
Sub3	39	31-32
Sub4	-	33-34
Sub5	-	35-36
Sub6	-	37-38
Sub7	-	Entre la 12 y la 30
Sub8	-	No entre la 12 y la 30

III-C. Metodología

Para resolver el problema, proponemos desarrollar modelos que predigan la producción de un producto para la semana siguiente. Esta producción se modela como una serie temporal del rendimiento histórico semanal del producto. El rendimiento del cultivo se utiliza como valor de la serie temporal en lugar de los kilogramos, ya que elimina el efecto de la influencia que tiene la superficie sobre el número total de kilogramos producidos por una parcela en un periodo. El rendimiento global de cada producto se obtiene dividiendo la suma total de los kilogramos producidos por las parcelas en producción en cada periodo de tiempo por la suma de las superficies de las parcelas que contribuyen en ese periodo de tiempo.

$$\text{Rendimiento de cultivo} = \frac{\sum kg}{\sum \text{superficie}} \tag{1}$$

La metodología propuesta para desarrollar los modelos predictivos se basa en la descomposición de las series temporales del rendimiento de los cultivos de un producto en sub-series temporales. En cada sub-serie temporal solo se tratan los cultivos que han sido plantados en un periodo de tiempo muy corto, por lo que como paso esencial para nuestra metodología se deben analizar y agrupar las semanas de plantación para aprovechar la homogeneidad en la duración y producción del cultivo y además cubrir el mayor número de parcelas posible. Como resultado de esto, las sub-series temporales que se obtengan presentarán comportamientos más predecibles.

Una vez que se tienen las sub-series parciales de un producto, se ajusta un modelo predictivo para cada una de ellas. El proceso de construcción del modelo predictivo sigue los siguientes pasos:

1. Preprocesamiento: Se realiza una imputación de valores faltantes en las sub-series temporales en caso de que existan, para que todas las sub-series temporales tengan la misma longitud. Además, se realiza una normalización min-max entre el rango [0,1]:

$$\text{normalización } TS = \frac{TS - min_{TS}}{max_{TS} - min_{TS}} \tag{2}$$

   donde TS denota la sub-serie temporal original, y min_{TS} y max_{TS} los valores mínimo y máximo de la sub-serie temporal. Por último, las series parciales se dividen en datos de entrenamiento y datos de prueba.
2. Modelado: Se ajustan modelos predictivos de las sub-series temporales de entrenamiento utilizando un conjunto de técnicas estadísticas y de *machine learning* (ARIMA, alisados exponenciales, máquinas de soporte vectorial, *random forest*, redes neuronales, redes neuronales parcialmente recurrentes y modelos aditivos). Para cada una de las técnicas anteriores se generan un conjunto de modelos en función de las diferentes parametrizaciones empleadas.
3. Validación: El paso de validación determina en primer lugar cuál es el modelo más adecuado para cada técnica y, a partir de él, cuál es el mejor modelo global. El mejor modelo será aquel que obtenga una menor medida de error y por tanto una mayor capacidad predictiva. Para evaluar el error y la capacidad predictiva de cada uno de los modelos generados se aplica una validación cruzada *leave-one-out* [7] sobre las sub-series temporales de prueba. Para esta variante de la validación cruzada, existe un único valor de prueba para cada iteración, con la particularidad de que el conjunto de entrenamiento está formado por los valores que se producen temporalmente antes del valor de prueba. Así pues, no se usan valores futuros al valor de prueba en el entrenamiento del modelo. En cada iteración, el valor de prueba de la iteración anterior se incorpora al conjunto de entrenamiento, y el siguiente valor temporal que antes no ha sido usado se incorpora como valor de prueba.
4. Predicción: Después de completar los pasos anteriores, se tiene el modelo predictivo más adecuado para cada sub-serie temporal. A partir de estos modelos se genera una predicción para la semana siguiente del rendimiento de cultivo en cada sub-serie temporal. La predicción en kilogramos se obtiene multiplicando el rendimiento previsto obtenido de la cosecha y la superficie estimada de cultivo de las parcelas:

$$kg = \text{rendimiento*superficie} \tag{3}$$

El área de cultivo se estima a partir de las parcelas que han plantado en la presente campaña. Para ello se añade la superficie de todas las parcelas plantadas, menos la superficie de aquellas parcelas que han plantado y que no han podido producir al llevar plantadas menos días del tiempo que el producto necesita para empezar a producir, y aquellas parcelas que ya han terminado de producir.

5. Agregación: Para obtener la predicción global de kilogramos, se suman los valores pronosticados de kilogramos de cada sub-serie temporal.

$$KG = \sum_{i=1}^{n} kg_i \tag{4}$$

Dónde $i = \{1, 2, ...n\}$ es el número de sub-series temporales.

IV. Estudio empírico

En esta sección, se describe el estudio empírico que hemos diseñado para evaluar la validez y el desempeño de nuestra propuesta de predicción para el problema abordado. La metodología se ha aplicado a casos reales. Se detallan los conjuntos de datos, las medidas de rendimiento y los enfoques competitivos. Después, se presentan los resultados empíricos obtenidos tras la realización de los experimentos.

IV-A. Procedimiento

Para comprobar la validez, el enfoque propuesto en este trabajo se ha comparado con un enfoque de predicción directo del problema. Esta estrategia directa consiste en aplicar algunos métodos de predicción a la serie temporal completa sin ningún tipo de descomposición. Las diferentes técnicas de predicción empleadas son ampliamente utilizadas en el estado del arte actual. En particular, los modelos considerados son: ARIMA, redes neuronales, máquinas de soporte vectorial, alisados exponenciales, redes neuronales parcialmente recurrentes, *Random Forest* y el modelo aditivo *Facebook Prophet* [17]. Argumentamos que este enfoque directo obtiene una menor capacidad predictiva, y por lo tanto, un enfoque más efectivo es considerar la descomposición basada en la semana de siembra.

IV-B. Configuración

Las series temporales de los productos y las sub-series temporales son univariantes. El periodo de las series temporales y las sub-series temporales se ha establecido en 52. Cada observación de la serie temporal y las sub-series temporales representa un valor semanal. Los valores de las series temporales se dividen en conjunto de entrenamiento y conjunto de pruebas. El porcentaje de valores en entrenamiento para todas las series temporales es el primer 65 % de los valores totales, dejando el 35 % restante como conjunto de pruebas. La longitud de la serie temporal del producto 1 es de 248, 161 valores para entrenamiento y 87 para la prueba. Para el segundo producto la longitud es de 222, 144 valores para el entrenamiento y 78 para la prueba. Se ha utilizado el error cuadrático medio (RMSE) y el error absoluto medio (MAE) como medida del rendimiento:

$$RMSE = \sqrt{\frac{1}{T}\sum_{t=1}^{T}(o_t - \bar{pe}_t)^2} \tag{5}$$

$$MAE = \frac{1}{T}\sum_{t=1}^{T}|o_t - \bar{pe}_t| \tag{6}$$

donde o_t denota el valor de rendimiento original de la sub-serie temporal, pe_t el valor de rendimiento predicho y T el número de valores de prueba. Por otro lado, todo el procesamiento se ha implementado en el lenguaje de programación R, utilizando gran cantidad de los paquetes disponibles en el repositorio CRAN [2], [8], [11], [13], [17].

IV-C. Conjunto de datos

Los datos de producción se han recogido de una cooperativa situada en el sur de España. Para este trabajo se han seleccionado dos cultivos:

- Producto 1. Vegetal de temporada. La producción de rendimiento para este producto está disponible desde septiembre de 2013 hasta abril de 2018, cubriendo las campañas 2013, 2014, 2015, 2016 y 2017. La plantación principal tiene lugar en el mes de septiembre, pero también se realizan pequeñas plantaciones en los meses de agosto y febrero.
- Producto 2. Fruta de temporada. La producción de rendimiento para este producto está disponible desde enero de 2014 hasta abril de 2018, cubriendo la mitad de la campaña de 2013 y las campañas completas de 2014, 2015 y 2016 y gran parte de la de 2017. La plantación principal ocurre en los meses de agosto-septiembre y en los meses de febrero-marzo.

IV-D. Resultados

La Tabla II muestra el RMSE en prueba obtenido con nuestra metodología propuesta y con el enfoque directo. La Tabla III muestra el MAE. En las filas de la primera columna de ambas tablas se muestra la metodología empleada, ya sea la metodología propuesta en este trabajo o la metodología directa. La segunda columna especifica el método de modelado empleado por la metodología, en el caso de la metodología propuesta se han empleado una combinación de todos los métodos para ajustar la serie, pero en el caso de la metodología directa se ha empleado solo un método en cada caso. La columna 3 se subdivide en dos columnas siendo cada una referente al error cometido en cada producto. Todos los valores RMSE y MAE están normalizados en el rango [0-1].

V. Análisis y discusión de los resultados

Los resultados de la Tabla II y Tabla III muestran como nuestro enfoque logra una mejor capacidad predictiva que el enfoque directo para cada uno de los casos reales al obtener un error menor tanto en RMSE como en MAE.

Tabla II
RMSE normalizado obtenido por nuestra propuesta y de manera directa sobre el subconjunto de prueba

Metodología	Modelo usado	Error cometido (RMSE)	
		Producto 1	Producto 2
Propuesta	Todos	**0.0371**	**0.0368**
Directa	ARIMA	0.0536	0.0458
Directa	Red Neuronal	0.0502	0.0446
Directa	SVM	0.0643	0.0685
Directa	Alisado Exponencial	0.0565	0.0475
Directa	Red Neuronal Parcialmente Recurrente	0.0457	0.0445
Directa	Random Forest	0.0658	0.0487
Directa	Modelo Aditivo	0.0600	0.0579

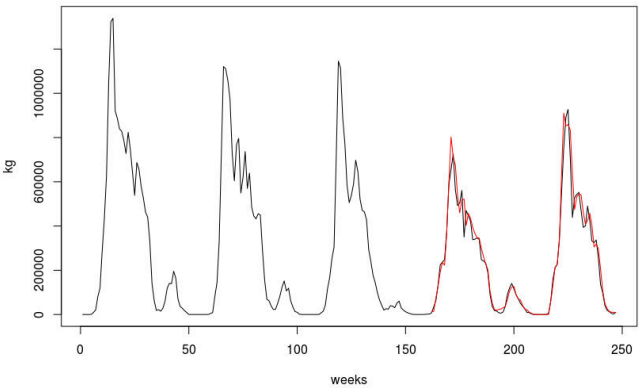
Tabla III
MAE normalizado obtenido por nuestra propuesta y de manera directa sobre el subconjunto de prueba

Metodología	Modelo usado	Error cometido (MAE)	
		Producto 1	Producto 2
Propuesta	Todos	**0.0214**	**0.0247**
Directa	ARIMA	0.0346	0.0400
Directa	Red Neuronal	0.0331	0.0292
Directa	SVM	0.0485	0.0403
Directa	Alisado Exponencial	0.0384	0.0350
Directa	Red Neuronal Parcialmente Recurrente	0.0346	0.0302
Directa	Random Forest	0.0447	0.0343
Directa	Modelo Aditivo	0.0488	0.0488

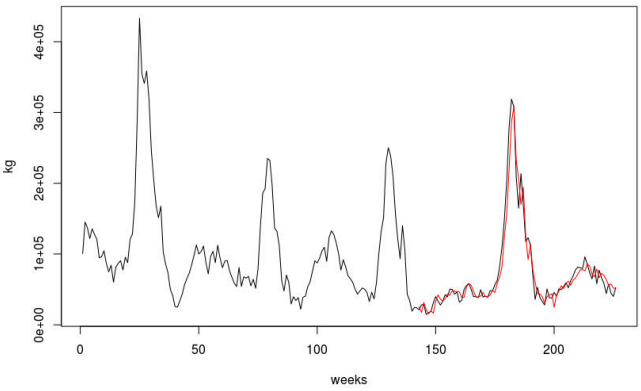
La mejora al emplear el enfoque con descomposición es considerable para el producto 1. El RMSE obtenido con nuestra metodología es de 0.0371, mientras que el menor RMSE obtenido con la metodología directa en cualquiera de sus variantes es de 0.0457 conseguido aplicando una red neuronal parcialmente recurrente. Para los ajustes con el resto de métodos en la metodología directa el RMSE es superior a 0.05 e incluso a 0.06. Por otro lado nuestra metodología obtiene un MAE más bajo de 0.0214, mientras que el menor MAE obtenido con la metodología directa es de 0.0331.

Mientras que para el producto 2 nuestra metodología ofrece también una mejor capacidad predictiva que la metodología directa. El RMSE más bajo obtenido con nuestro enfoque es del 0.0368 mientras que el menor de los RMSE obtenidos por el enfoque directo directo es de 0.0445, conseguido también con una red neuronal parcialmente recurrente. El resto de ajustes con el enfoque directo ofrecen un RMSE similar al 0.0445 conseguido por la red neuronal parcialmente recurrente, salvo varias excepciones. Con respeto al MAE el menor conseguido por nuestra metodología es de 0.0247, y el menor del enfoque directo es de 0.0292. El resto de variantes del enfoque directo obtienen un MAE superior al 0.03.

Así pues estos resultados muestran que tanto para el caso del producto 1 como el producto 2 los errores cometidos por nuestro enfoque son siempre menores que los errores obtenidos por el enfoque directo, siendo la mejora obtenida considerable.



(a) Producto 1



(b) Producto 2

Figura 1. Serie temporal de los kilogramos de producción real y predicción realizada por nuestra metodología (línea roja) en los casos de prueba para los dos productos analizados

VI. Conclusiones

En este trabajo se ha presentado una forma innovadora de realizar el análisis de una serie temporal del rendimiento de la producción de productos reales de una cooperativa hortofrutícula mediante la descomposición de la serie temporal general en sub-series temporales según la semana de plantación, siendo el modelo predictivo final el resultado de la mezcla entre los mejores modelos ajustados para cada sub-serie a partir de un conjunto de métodos estadísticos y de Aprendizaje Automático. Además, esta metodología ha sido comparada con otra implementación posible del mismo problema.

Los resultados de los experimentos han mostrado que la metodología propuesta mejora la capacidad predictiva de la metodología comparada, ya que obtiene un error menor para ambos productos analizados. Además, siempre mejora el error obtenido en cada una de las comparaciones. Por lo que se puede concluir que una descomposición de la serie temporal según las semanas significativas de plantación se da lugar a

que se obtenga una mejora significativa en la predicción con respecto a la serie sin descomponer.

Agradecimientos

Esta investigación ha sido parcialmente financiada por los Proyectos de Investigación Nacional TIN2013-47210-P, y TIN2016-81113-R y por el Proyecto de Investigación de Excelencia de la Junta de Andalucía P12-TIC-2958.

Referencias

[1] Akram, M., Bhatti, I., Ashfaq, M., Khan, A.A. *Hierarchical Forecasts of Agronomy-Based Data*, American Journal of Mathematical and Management Sciences, 36(1), 49-65, 2017

[2] Bergmeir C. and Benítez J.M. *Neural Networks in R Using the Stuttgart Neural Network Simulator: RSNNS*. Journal of Statistical Software, 46(7), 1-26, 2012

[3] Box G.E.P. and Jenkins G.M. *Time Series Analysis, Forecasting and Control*, 1970

[4] Brdar S., Culibrk D., Marinkovic B., Crnobarac J., Crnojevic V. *Support Vector Machines with Features Contribution Analysis for Agricultural Yield Prediction*, Second International Workshop on Sensing Technologies in Agriculture, Forestry and Environment, 43-47, 2011

[5] Choudhury, A. and Jones, J. *Crop yield prediction using time series models*, Journal of Economics and Economic Education Research., 15, 53-68, 2014.

[6] Fukuda S., Spreer W., Yasunaga E., Yuge K., Sardsud V. and Müller J. *Random Forests modelling for the estimation of mango (Mangifera indica L. cv. Chok Anan) fruit yields under different irrigation regimes*, Agricultural Water Management, 116(1), 142-150,2013

[7] Hastie T., Tibshirani R., Friedman, J. *The Elements of Statistical Learning*, Springer, New York, 2009.

[8] Hyndman R.J. and Khandakar Y. *Automatic time series forecasting: the forecast package for R*. Journal of Statistical Software, 26(3), 1-22, 2008

[9] Jeong J.H., Resop J., Mueller N., Fleisher D.H., Yun K.,Butler E., Timlin D., Shim K., Gerber J., Reddy V., and Kim S.H. *Random Forests for Global and Regional Crop Yield Predictions*, PLoS One. 2016; 11(6): e0156571.

[10] Ji, B., Sun Y., Yang S. and Wan J. *Artificial neural networks for rice yield prediction in mountainous regions*, Journal of Agricultural Science, 145: 249-26, 2007.

[11] Karatzoglou A., Smola A., Hornik K. and Zeileis A. *kernlab - An S4 Package for Kernel Methods in R*. Journal of Statistical Software, 11(9), 1-20, 2004

[12] Kaul M., Hill R.L., and Walthall C. *Artificial neural networks for corn and soybean yield prediction*, Agricultural Systems, 85(1): 1-18, 2005.

[13] Liaw A. and Wiener M. *Classification and Regression by randomForest*. R News, 2(3), 18-22, 2002

[14] Liu J., Goering C.E., and Tian L. *A neural network for setting target corn yields*, Transactions of the ASAE. Vol. 44(3): 705–713, 2007.

[15] Najeeb I., Khuda B., Asif M. *Use of the ARIMA Model for Forecasting Wheat Area and Production in Pakistan*, Journal of Agriculture ans Social Sciences, 1(2), 120-122, 2015

[16] Ruß G. *Data Mining of Agricultural Yield Data: A Comparison of Regression Models*, In: Perner P. (eds) Advances in Data Mining. Applications and Theoretical Aspects, ICDM 2009. Lecture Notes in Computer Science, vol 5633.

[17] Taylor S. and Letham B. *prophet: Automatic Forecasting Procedure. R package version 0.1*. 2017

[18] Wuo W., Xue H. *An incorporative statistic and neural approach for crop yield modelling and forecasting*, Neural Computing and Applications, 21(1): 109–117, 2012.

¿Requiere la clasificación de series temporales métodos específicos?

Amaia Abanda[1,2]
[1] *Basque Center for Applied Mathematics (BCAM)*
[2] *Intelligent Systems Group (ISG)*
Department of Computer Science and Artificial Intelligence,
University of the Basque Country UPV/EHU
Bilbao, Spain
aabanda@bcamath.org

Usue Mori[2,3]
[2] *Intelligent Systems Group (ISG)*
Department of Computer Science and Artificial Intelligence
[3] *Department of Applied Mathematics*
Statistics and Operational Research
University of the Basque Country UPV/EHU
Bilbao, Spain
usue.mori@ehu.es

Jose A. Lozano[1,2]
[1] *Basque Center for Applied Mathematics (BCAM)*
[2] *Intelligent Systems Group (ISG)*
Department of Computer Science and Artificial Intelligence,
University of the Basque Country UPV/EHU
Bilbao, Spain
ja.lozano@ehu.eus

***Resumen*—La clasificación de series temporales tiene la peculiaridad, respecto a otros problemas de clasificación supervisada, de que las observaciones de las series o variables predictoras tienen un orden específico. La mayoría de las soluciones propuestas en la literatura consideran que este orden es discriminatorio para la clasificación y, por tanto, emplean métodos específicos que tienen en cuenta el orden. El objetivo de esta investigación es explorar, de una manera preliminar, si realmente es siempre necesario el uso de métodos específicos para series temporales o si hay algunos casos en los que los métodos de clasificación no específicos, habitualmente utilizados para datos de tipo convencional, obtienen mejores resultados que los específicos. La experimentación llevada a cabo en 40 bases de datos del repositorio UCR muestra como en los casos en los que el orden de las observaciones no es relevante para la clasificación, los clasificadores no específicos consiguen mejorar la precisión, mientras que en los casos en los que el orden es un factor clave no lo consiguen.**

***Index Terms*—Clasificación, series temporales, orden, vector, información temporal, métodos específicos.**

I. Introduction

Una serie temporal es una secuencia de datos u observaciones que viene ordenada respecto al tiempo -la mayoría de las veces-, o respecto a otros aspectos como el espacio, por ejemplo. Es un tipo particular de datos precisamente porque tiene una naturaleza ordinal que la mayoría de datos no tienen. Las series temporales aparecen naturalmente en diferentes áreas como la bio-informática o economía [1], y están adquiriendo una gran relevancia en el ámbito del aprendizaje automático debido al gran reto que supone trabajar con este tipo especial de datos. De esta manera, continuamente están surgiendo nuevos métodos para representar, indexar, agrupar y clasificar series temporales [2]. El presente trabajo se sitúa dentro del área de la clasificación de series temporales -Time Series Classification (TSC)-, donde cada serie del conjunto de entrenamiento tiene asociada una clase, y el objetivo es encontrar una función tal que, dada una serie nueva, sea capaz de predecir cuál es su clase. La diferencia principal con el problema clásico de clasificación es que, mientras en éstos las instancias vienen descritas por atributos sin orden específico, en la clasificación de series temporales las instancias vienes definidas por las propias series temporales completas [3][4]. Además de que las variables predictoras están ordenadas en este caso, las series pueden tener un número muy alto de observaciones, las longitudes pueden ser variables dentro de una misma base de datos y, dependiendo del contexto, pueden ser observaciones con mucho ruido [5].

La peculiaridad de las series temporales, su posible interpretabilidad y su significado semántico han llevado a la comunidad científica a asumir que éstas tienen que ser tratadas con métodos específicamente diseñados para series [4][6][7][8]. Estos métodos tienen en cuenta la información temporal que intrínsecamente contienen las series para extraer atributos, construir modelos, medidas de similitud, etc. útiles para la clasificación. El orden de las observaciones ha sido hasta ahora una de las características más comúnmente asumidas como información temporal discriminatoria; en los métodos basados en distancias elásticas [3][9][10], en los métodos basados en modelos autoregresivos [11][12] en los modelos ocultos de Márkov [13], o en los métodos extraen atributos temporales de las series [5].

Esta asunción implica que los métodos no específicos, es decir, los métodos basados en clasificadores tradicionales

que no tienen en cuenta la información temporal, no deberían ser competitivos con los específicos en cuanto a precisión se refiere. En este trabajo se pretende explorar hasta qué punto esta afirmación es cierta y, sobre todo, en qué casos se cumple o no se cumple.

Este trabajo parte de la clasificación de series temporales basada en distancias [14], donde se utiliza la distancia/similitud entre series como criterio para la clasificación; dos series que sean similares pertenecerán a la misma clase. En particular, el método que hasta ahora más se ha utilizado para en la clasificación de series temporales basada en distancias es el 1-Nearest Neighbour (1NN), clasificador que aunque haya demostrado buenos resultados en el ámbito [2][9][15][16], no es muy robusto. En cuanto a las distancias utilizadas, en el ámbito de la clasificación de series temporales, podemos distinguir entre distancias que tienen en cuenta el orden de las observaciones y distancias que no lo tienen, pero muchas de las distancias situadas en el segundo grupo son muy costosas computacionalmente. En este contexto, sería interesante saber en qué casos no es necesario el uso de métodos específicos -casos en los que se podrían utilizar clasificadores estándares de una manera eficiente y con mejores resultados-.

El trabajo se organiza de la siguiente forma: en la Sección II se introducen algunos conceptos básicos para contextualizar la clasificación de series temporales basada en distancias, en la Sección III se presenta la propuesta y la experimentación realizada y, finalmente, en la Sección IV presentamos las principales conclusiones obtenidas.

II. Conceptos básicos y motivación

En esta sección se presenta un breve resumen de los conceptos básicos necesarios para abordar la clasificación de series temporales basada en distancias y se introducen algunas de las distancias más conocidas para series.

II-A. Clasificación de series temporales

Los métodos de clasificación de series temporales se pueden dividir entre tres principales categorías [15]: métodos basados en atributos, basados en modelos y basados en distancias. Dentro de la primera línea, los investigadores tratan de extraer características fundamentales de las series para obtener una nueva representación no-ordenada y de menor dimensión que, además, contenga la información más relevante de las series. Esta transformación evita tener que aprender directamente con las series originales y las traslada a un nuevo espacio donde las características discriminatorias pueden ser más detectables [17]. Algunos ejemplos de estas representaciones incluyen Discrete Fourier Transformation (DFT) [5], Discrete Wavelet Transformation (DWT) [18] o Piecewise Aggregate Approximation (PAA) [19]. Por otro lado, en los métodos basados en modelos, se asume que todas las series de una clase han sido generadas por un modelo subyacente y a una serie nueva se le asigna la clase del modelo al que mejor se ajuste. Algunos de los modelos más utilizados son los modelos auto-regresivos [11] [12] o los modelos ocultos de Markov [13]. Por último, en los métodos basados en distancias los investigadores tratan de definir una similitud, o disimilitud, entre series que tenga en cuenta diferentes características semánticas o temporales. La definición de una distancia adecuada es una cuestión crucial, ya que cada distancia refleja diferentes características de las series. Una vez elegida la distancia adecuada, la clasificación se lleva a cabo empleando métodos basados en distancias, en la mayoría de los casos clasificadores basados en el vecino más cercano (1NN). Entre las distancias para series temporales más conocidas están la distancia Euclídea, Dynamic Time Warping (DTW) [20] o Edit Distance with RealPenalty (ERP) [21].

II-B. Distancias de series temporales

Las distancias entre series temporales suelen categorizarse habitualmente en dos grupos principales mostradas en la Figura 1 [22]; las *medidas rígidas*, que se refieren a aquellas medidas que comparan el punto i-ésimo de una serie con el punto i-ésimo de la otra, y las *medidas elásticas*, que tratan de crear un mapeo no lineal entre las series para alinearlas, permitiendo una comparación de uno-a-varios puntos. La mayor diferencia entre estos dos tipos de medidas es que las medidas rígidas, al no considerar los puntos de alrededor, tratan la serie como si fuera un vector y por tanto, no consideran la ordenación de las mediciones. De esta forma, aún desordenando la serie, obtendríamos los mismos resultados de clasificación. Las medidas elásticas, por el contrario, consideran que el orden de las observaciones es determinante y hacen uso de la misma en su cálculo. Por tanto, otra ordenación de las observaciones podría cambiar por completo los resultados de la clasificación.

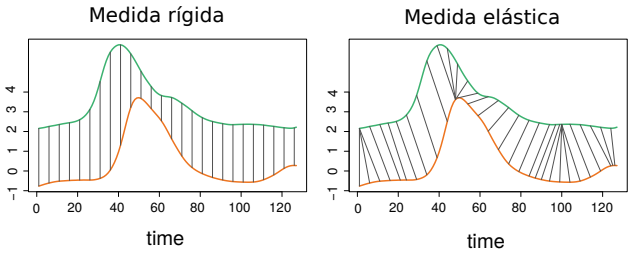


Figura 1: Alineamientos de medidas rígidas y elásticas

La distancia rígida más habitual es la denominada distancia Euclídea. Se define de la siguiente forma: dadas dos series temporales T=(t_1,...,t_n) y S=(s_1, ..., s_n),

$$ED(T,S) = \sqrt{\sum_{i=1}^{n}(t_i - s_i)^2}$$

Esta distancia es ampliamente conocida debido que cumple buenas propiedades, como ser métrica, así como su bajo coste computacional y simplicidad, razones por las

que es habitualmente utilizada en numerosos problemas de aprendizaje automático. Sin embargo, tiene también algunos inconvenientes como que sólo acepta dos series que tengan la misma longitud, o como se ha comentado anteriormente, que no considera el orden de las observaciones.

En vista de las carencias de la distancia Euclídea, en los últimos años se han presentado numerosas distancias elásticas específicamente diseñadas para medir la disimilitud entre series temporales. La más habitual entre ellas es la Dynamic Time Warping (DTW). Esta distancia alinea las dos series con tal de minimizar la distancia entre ellas y, por lo tanto, es robusta frente a desfases y distorsiones en el tiempo. Además, a diferencia de la distancia Euclídea, permite calcular la distancia entre series de diferentes longitudes. Dadas dos series temporales T y S, el objetivo de la DTW es buscar la alineación óptima que minimice la distancia entre ellas. Para ello, el primer paso es construir una matriz de distancias de n×m donde cada posición (i, j) contiene la distancia $(t_i - s_j)^2$, que representa el coste de alinear la observación i-ésima de T con la observación j-ésima de S. Así, un alineamiento entre dos series se define como un camino π en la matriz de distancias, que tiene que cumplir ciertas restricciones (continua, monótona creciente y puntos de inicio y final fijos) (ver Figura 2). El alineamiento óptimo entre dos series es aquel que minimiza la distancia acumulativa. Finalmente, la distancia DTW(T,S) se define como la distancia acumulativa del camino óptimo entre T y S. En la práctica, este camino óptimo se encuentra mediante métodos de programación dinámica [23] y uno de las mayores desventajas de ésta distancia es que este proceso tiene un coste computacional de $\mathcal{O}(N^2)$, siendo N el número de series.

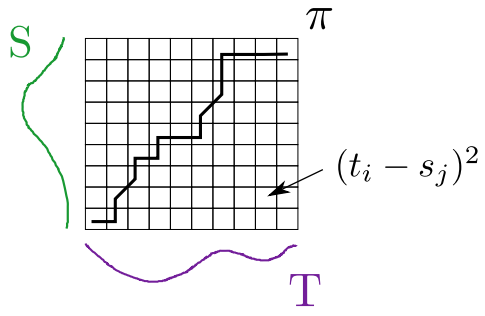


Figura 2: Matriz de distancias y camino óptimo de DTW

La mayoría de los métodos de clasificación de series temporales asumen que el orden de las observaciones es fundamental para discriminar entre clases y emplean medidas elásticas como la DTW [24][16][10]. El objetivo de este trabajo es realizar un estudio preliminar para explorar si realmente es, o en qué casos, la información temporal -en particular, el orden de las observaciones- fundamental para la clasificación. Esta cuestión proporcionaría información importante sobre la utilidad de clasificadores específicos para series temporales, ya que si el orden de las observaciones no es relevante, no habría necesidad de usar métodos específicos -clasificadores basados en la temporalidad-, sino que las series podrían entenderse como simples vectores y podrían utilizarse también otros clasificadores clásicos, obteniendo quizás resultados mejores.

III. Propuesta y experimentación

Como hemos expuesto anteriormente, el objetivo de este trabajo es explorar si realmente la información temporal y, en particular, el orden de las observaciones, juega siempre un papel fundamental en la clasificación de series temporales. Además, la hipótesis principal es que en aquellos casos donde el orden no es discriminatorio, los clasificadores no específicos podrían funcionar mejor, debido a que son más complejos y robustos que el 1-NN. Es decir, en los casos donde el orden no es relevante, las series pueden ser interpretadas como vectores (*feature vectors*) y, por tanto, las observaciones puede entenderse como dimensiones que no tienen un orden pre-definido. En los casos donde el orden es un aspecto discriminatorio para la clasificación, el hecho de que cada observación esté en una posición concreta es relevante y por tanto no se pueden interpretar las series como vectores. Los clasificadores no específicos, o tradicionales, esperan como entrada un vector sin orden, por lo que en el primer caso podrían funcionar bien mientras que en el segundo, al no ser capaces de tener en cuenta el orden, deberían obtener peores resultados.

El estudio parte de la clasificación basada en distancias y, más particularmente, tomamos el clasificador 1-NN como referencia. Como distancia que no considera el orden de las observaciones se utiliza la distancia Euclídea, mientras que como distancia que sí considera el orden consideraremos la DTW.

Asumiremos que en los casos en los que la distancia Euclídea obtiene mayor precisión que la DTW, el orden de las observaciones no es relevante para la discriminación, mientras que en las que la distancia DTW obtiene mayor precisión sí lo es. El objetivo es explorar si en los casos en los que 1-NN-EUC (método no específico) tiene mayor precisión que el 1-NN-DTW (método específico), los clasificadores estándares (métodos no específicos), como el Support Vector Machine o naive Bayes, mejoran más (veces) los resultados que en los casos en los que el 1-NN-DTW tiene mayor precisión que el 1-NN-EUC. En particular, se quiere comprobar si, en el caso de que el 1-NN-EUC obtiene mejores resultados que el 1-NN-DTW, los clasificadores estándares pueden llegar a mejorar los resultados obtenidos con el 1-NN.

Con el objetivo de poder contrastar las hipótesis planteadas, la clasificación se ha llevado a cabo utilizando, por un lado, 1-NN-Euclídea y 1-NN-DTW y, por otro, 3 clasificadores estándares (no específicos): Support Vector Machine (SVM), naive Bayes (NB) y Random Forest (RF). No se han ajustado los parámetros de ninguno para poder establecer una base de referencia. Los experimentos se han

realizado con 40 bases de datos del repositorio de series temporales UCR [25], un repositorio de bases de datos de series temporales que es habitualmente utilizada como referencia para evaluar nuevos métodos de clasificación de series temporales.

En el Cuadro I se muestran los resultados de la clasificación de las bases de datos donde el 1-NN-EUC obtiene mayor precisión que el 1-NN-DTW (15/40), mientras que en el Cuadro II la de aquellas en las que el 1-NN-DTW obtiene mayor precisión que 1-NN-EUC (25/40). Estas proporciones eran de esperar debido a que el 1-NN-DTW se considera uno de los métodos que mejores resultados obtienen en la clasificación de series temporales [2][9][16] [15]. Se han separado las bases de datos en estas dos tablas para poder visualizar mejor si hay una tendencia diferente en cada uno de los casos.

En el primer cuadro, donde el método no específico obtiene mejores resultados que el específico, se puede observar que los clasificadores estándares superan en algunos casos (12/15) al 1-NN. En el segundo cuadro, en cambio, donde el método específico funciona mejor que el no específico, los clasificadores estándares no superan al 1-NN en casi ningún caso (6/25).

BBDD	EUC	DTW	SVM	NB	RF
Adiac	0.61	0.52	0.40	0.57	**0.64**
Beef	0.53	0.50	0.50	0.47	**0.57**
Chlothe	0.65	0.62	0.58	0.35	**0.71**
CinC	0.90	0.69	0.73	0.88	0.76
ECG	0.88	0.80	0.87	0.77	0.83
ECGF5	0.80	0.75	**0.84**	0.78	**0.80**
GunPoint	0.91	0.89	0.81	0.78	**0.92**
Haptics	0.37	0.35	**0.45**	**0.44**	**0.46**
SonyAI	0.70	0.66	**0.78**	**0.94**	0.68
SonyAI II	0.86	0.82	0.82	0.79	0.80
Swedish	0.79	0.75	**0.88**	**0.86**	**0.88**
WaveX	0.74	0.73	**0.76**	0.66	**0.76**
WaveY	0.66	0.64	**0.67**	0.56	**0.69**
Wafer	0.99	0.98	**0.99**	0.70	**0.99**
FetalECG2	0.88	0.83	**0.90**	0.83	**0.91**

Cuadro I: Resultados de clasificación de las bases de datos donde 1-NN-Euc obtiene mayor precisión que 1-NN-DTW.

Con fin de entender mejor los resultados, el Cuadro III sintetiza y compara los dos cuadros anteriores. En él se refleja para cada una de las tablas (representadas por las filas EUC y DTW) el número de veces en el que al menos uno -y al menos dos- de los clasificadores no-específicos empleados supera la precisión del 1-NN. Se puede apreciar que en los casos del Cuadro I (fila EUC), los métodos no-específicos tienden a superarlo al menos una vez en el 80 % de los casos, mientras que los casos del Cuadro II (fila DTW), este porcentaje baja hasta el 24 %. Lo mismo sucede si contamos las veces en las que los métodos no-específicos superan al menos 2 veces al 1NN: el valor es 53 % para las bases de datos donde la Euclídea funciona mejor y 4 % para las del DTW.

Por lo tanto, se puede apreciar una tendencia a que los

BBDD	EUC	DTW	SVM	NB	RF
50Words	0.63	0.71	0.62	0.58	0.66
CBF	0.85	0.97	0.87	0.90	0.91
Coffee	0.75	0.79	0.71	0.68	0.68
Cricket_X	0.57	0.73	0.54	0.44	0.62
Cricket_Y	0.64	0.75	0.63	0.53	0.74
Cricket_Z	0.62	0.73	0.58	0.42	0.66
Diatom	0.93	0.96	0.90	0.87	0.91
Face_all	0.71	0.75	0.74	0.69	**0.81**
Face_four	0.78	0.84	0.64	**0.90**	0.77
FacesUCR	0.77	0.92	0.73	0.74	0.79
Fish	0.78	0.85	0.78	0.66	0.79
InlineSkate	0.34	0.38	0.25	0.22	0.35
Lightning2	0.75	0.80	0.70	0.67	0.75
Lightning7	0.58	0.77	0.63	0.71	0.75
MALLAT	0.91	0.92	0.77	0.87	**0.93**
MedicalImg	0.68	0.73	0.57	0.44	**0.73**
MoteStrain	0.88	0.90	0.85	0.84	0.89
OSU Leaf	0.52	0.63	0.52	0.38	0.51
Symbols	0.90	0.95	0.76	0.64	0.87
Synthetic	0.88	0.95	0.98	**0.96**	**0.96**
Trace	0.76	0.99	0.71	0.80	0.86
Two Lead	0.75	0.93	0.72	0.69	0.73
Two Patterns	0.91	0.99	0.89	0.46	0.85
WaveY	0.65	0.65	0.70	0.56	**0.71**
WordSyn	0.62	0.67	0.53	0.48	0.57

Cuadro II: Resultados de clasificación de las bases de datos donde 1-NN-Dtw obtiene mayor precisión que 1-NN-Euc.

métodos no-específicos funcionen mejor en los casos en los que la distancia Euclídea funciona mejor que la DTW con el 1-NN. Es decir, en las bases de datos en las que el orden de las observaciones no es relevante para clasificar y los métodos no-específicos -que no tienen en cuenta el orden- funcionan mejor que en las bases de datos que en las que el orden si es relevante.

	$\geq$ 1 vez	$\geq$ 2 veces
EUC	12/15 = 80 %	8/15 = 53 %
DTW	6/25 = 24 %	1/25 = 4 %

Cuadro III: Comparativa de las veces que los clasificadores estándares superan el 1-NN en el método no específico (EUC) y específico (DTW).

Para evaluar estadísticamente los Cuadros I y II, se han realizado mútliples test estadísticos emparejados de Wilcoxon utilizando el paquete 'scmamp' [26] para R. En el primer caso, se ha realizado 3 test estadísticos para comprobar si hay diferencias significativas entre los métodos del Cuadro I. En particular, s han comparado las siguientes columnas: 1-NN-EUC con SVM, 1-NN-EUC con NB y 1-NN-EUC con RF. Después de corregir los p-valores obtenidos de cada uno de ellos (0.34,0.09 y 0.28, correspondientemente), no podemos rechazar la hipótesis nula y, por tanto, no podemos afirmar que haya diferencias significativas entre los métodos. Es decir, no hay evidencias estadísticas de que los métodos no-específicos y el 1-NN tengan una comportamiento diferente. En el segundo caso, se han comparado los resultados de los métodos del Cuadro II; en particular, comparando el 1-NN-DTW con SVM,

1-NN-DTW con NB y 1-NN-DTW con RF. Después de corregir los p-valores obtenidos (0.00004, 0.00004 y 0.0008) podemos rechazar la hipótesis nula y afirmar que hay diferencias significativas entre los métodos. En particular, podemos afirmar que los métodos no-específicos tienen peores resultados de clasificación que el 1-NN-DTW. Resumiendo, en los casos donde la distancia Euclídea funciona mejor que la DTW (el orden no tiene importancia), los métodos no-específicos se comportan parecido al 1-NN, mientras que en los casos donde la DTW funciona mejor (el orden sí tiene importancia), funcionan peor que el 1-NN.

Para terminar, se ha realizado otros 2 test estadísticos para comparar si hay diferencias significativas entre los resultados del 1-NN y el mejor de los 3 métodos no-específicos en cada caso (Cuadro IV). En ambos casos, los test afirman que hay diferencias significativas con p-valores 0.01 y 0.002. En los casos donde el 1-NN-EUC obtiene mejores resultados que el 1-NN-DTW, el test indica que al menos uno de los 3 métodos no-específicos obtiene mejores resultados que el 1-NN. En los casos donde el 1-NN-DTW obtiene mejores resultados que el 1-NN-EUC, por el contrario, el test muestra que los 3 métodos específicos obtiene peores resultados que el 1-NN.

BBDD	EUC	Mejor
Adiac	0.61	**0.64**
Beef	0.53	**0.57**
Chlothe	0.65	**0.71**
CinC	**0.90**	0.88
ECG	**0.88**	0.87
ECGF5	0.80	**0.84**
GunPoint	0.91	**0.92**
Haptics	0.37	**0.46**
SonyAI	0.70	**0.94**
SonyAI II	**0.86**	0.82
Swedish	0.79	**0.88**
WaveX	0.74	**0.76**
WaveY	0.66	**0.69**
Wafer	**0.99**	**0.99**
FetalECG2	0.88	**0.91**

BBDD	DTW	Mejor
50Words	0.71	0.66
CBF	**0.97**	0.91
Coffee	**0.79**	0.71
Cricket_X	**0.73**	0.62
Cricket_Y	**0.75**	0.74
Cricket_Z	**0.73**	0.66
Diatom	**0.96**	0.91
Face_all	0.75	**0.81**
Face_four	0.84	**0.90**
FacesUCR	**0.92**	0.79
Fish	**0.85**	0.79
InlineSkate	**0.38**	0.35
Lightning2	**0.80**	0.75
Lightning7	**0.77**	0.75
MALLAT	**0.92**	0.93
MedicalImg	**0.73**	**0.73**
MoteStrain	**0.90**	0.89
OSU Leaf	**0.63**	0.52
Symbols	**0.95**	0.87
Synthetic	0.95	**0.98**
Trace	**0.99**	0.86
Two Lead	**0.93**	0.73
Two Patterns	**0.99**	0.89
WaveY	0.65	**0.71**
WordSyn	**0.67**	0.57

Cuadro IV: Resultados de clasificación del 1-NN y el mejor de los 3 clasificadores estándares en cada base de datos.

IV. Conclusiones

En este trabajo el objetivo era explorar de una manera preliminar si realmente es necesario siempre utilizar los métodos específicos que tienen en cuenta la información temporal, en particular, el orden de las observaciones, para la clasificación de series temporales. La mayoría de los métodos habitualmente utilizados en clasificación de series temporales, especialmente aquellos basados en distancias, asumen, por defecto, que el orden es un factor discriminatorio para la clasificación y que, por tanto, interpretar las series como vectores sin orden no aportaría beneficios. Está cuestión limita el uso de clasificadores estándares que no están basados en la temporalidad (por ejemplo SVM, naive Bayes o Random Forest), ya que si el orden de las series es crucial, utilizar estos clasificadores que no lo tienen en cuenta no tendría sentido. Sin embargo, la hipótesis de este estudio es que no hay que asumir siempre que el orden es discriminatorio y hay casos en los que los clasificadores estándares pueden funcionar igual o mejor que el tan valorado 1-NN.

Para evaluar está hipótesis se han tomado como métodos de referencia el 1-NN-EUC (que no tiene en cuenta el orden de las observaciones) y el 1-NN-DTW (que sí tiene en cuenta el orden de las observaciones). De esta manera, en la experimentación se observa que en los casos donde el 1-NN-EUC obtiene mejores resultado que el 1-NN-DTW, los clasificadores estándares pueden mejorar la precisión de clasificación (en el 80 % de los casos), mientras que en los otros casos los clasificadores estándares no consiguen mejorarlo (solo en el 24 % de los casos lo hacen).

Este resultado era esperado ya que si en una base de datos el 1-NN-EUC obtiene mayor precisión que el 1-NN-DTW se puede suponer que el orden de las observaciones no es un factor determinante para la clasificación. En ese caso, los clasificadores estándares, que son más complejos y robustos que el 1-NN, y no tienen en cuenta en orden, deberían funcionar bien. Por otro lado, si en una base de datos el 1-NN-DTW obtiene mayor precisión que el 1-NN-EUC se puede interpretar que el orden de las observaciones sí es un factor determinante para la clasificación, y tiene sentido que los clasificadores estándares, que no tienen en cuenta el orden de las observaciones, rindan peor. De hecho, los test estadísticos realizados a los resultados confirman las hipótesis. En otras palabras, cuando el orden de las observaciones no es un factor discriminatorio, los clasificadores estándares mejoran el resultado del 1-NN, mientras que cuando el orden es sí es relevante no lo consiguen.

En conclusión, no es que los clasificadores estándares, y métodos no específicos en general, sean inadecuados para la clasificación de series temporales, si no que dependiendo de las características discriminatorias de cada base de datos (en particular, la relevancia del orden de las observaciones para la clasificación), pueden llegar a ser beneficiosos. Por ello, sería interesante entender más en profundidad cuáles son las características temporales discriminatorias de las series, seguir estudiando la importancia del orden para la clasificación y buscar otras maneras de evaluar la influencia de este factor. Además, sería también enriquecedor seguir explorando el uso de clasificadores estándares, o métodos no específicos, para series temporales, ya que son mucho más robustos que el

1-NN y dependiendo de la distancia empleada en el 1-NN, mucho más rápidos.

Agradecimientos

Esta investigación esta subvencionada por el Gobierno Vasco a través del programa BERC 2018-2021, así como del Ministerio de Economía y Competitividad (MINECO) mediante la acreditación de excelencia BCAM Severo Ochoa SEV-2013-0323. También del proyecto TIN2017-82626-R, financiado por AEI/FEDER (UE) y acrónimo "GECECPAST". Además, gracias al Programa de Grupos de Investigación 2013-2018 (IT-609-13) (Gobierno Vasco) y TIN2016-78365-R (MINECO). A. Abanda está subvencionada con la beca BES-2016-076890.

Referencias

[1] E. Keogh and S. Kasetty, "On the need for time series data mining benchmarks," *Proceedings of the 8th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, p. 102, 2002. [Online]. Available: http://dl.acm.org/citation.cfm?id=775047.775062

[2] P. Esling and C. Agon, "Time-series data mining," *ACM Computing Surveys*, vol. 45, no. 1, pp. 1–34, 2012. [Online]. Available: http://dl.acm.org/citation.cfm?doid=2379776.2379788

[3] A. Bagnall, J. Lines, A. Bostrom, J. Large, and E. Keogh, "The great time series classification bake off: a review and experimental evaluation of recent algorithmic advances," *Data Mining and Knowledge Discovery*, vol. 31, no. 3, pp. 606–660, 2017.

[4] T. C. Fu, "A review on time series data mining," *Engineering Applications of Artificial Intelligence*, vol. 24, no. 1, pp. 164–181, 2011. [Online]. Available: http://dx.doi.org/10.1016/j.engappai.2010.09.007

[5] C. Faloutsos, M. Ranganathan, and Y. Manolopoulos, "Fast subsequence matching in time-series databases," *ACM SIGMOD International Conference on Management of Data*, pp. 419–429, 1994.

[6] M. Cuturi and J. Vert, "A kernel for time series based on global alignments," *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. 1, pp. 413–416, 2007. [Online]. Available: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=4217433

[7] R. Povinelli, "Time series classification using Gaussian mixture models of reconstructed phase spaces," *Knowledge and Data ...*, vol. 16, no. 6, pp. 779–783, 2004. [Online]. Available: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=1294898

[8] D. Eads, D. Hill, S. David, S. Perkings, J. Ma, R. Porter, and J. Theiler, "Genetic Algorithms and Support Vector Machines for Time Series Classi cation," *Applications and Science of Neural Networks, Fuzzy Systems, and Evolutionary Computation*, vol. 4787, 2002.

[9] X. Wang, A. Mueen, H. Ding, G. Trajcevski, P. Scheuermann, and E. Keogh, "Experimental comparison of representation methods and distance measures for time series data," *Data Mining and Knowledge Discovery*, vol. 26, no. 2, pp. 275–309, 2013.

[10] J. Lines and A. Bagnall, "Time series classification with ensembles of elastic distance measures," *Data Mining and Knowledge Discovery*, vol. 29, no. 3, pp. 565–592, 2015.

[11] A. Bagnall and Gareth Janacek, "A run length transformation for discriminating between auto regressive time series," *Journal of Classification*, vol. 31, no. October, pp. 274–295, 2014.

[12] M. Corduas and D. Piccolo, "Time series clustering and classification by the autoregressive metric," *Computational Statistics and Data Analysis*, vol. 52, no. 4, pp. 1860–1872, 2008.

[13] P. Smyth, "Clustering sequences with hidden Markov models," *Advances in Neural Information Processing Systems*, vol. 9, pp. 648–654, 1997. [Online]. Available: http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.44.3648

[14] A. Abanda, U. Mori, and J. A. Lozano, "A review on distance based time series classification," *https://arxiv.org/abs/1806.04509*, pp. 1–28, 2018. [Online]. Available: http://arxiv.org/abs/1806.04509

[15] Z. Xing, J. Pei, and E. Keogh, "A brief survey on sequence classification," *ACM SIGKDD Explorations Newsletter*, vol. 12, no. 1, p. 40, 2010. [Online]. Available: http://portal.acm.org/citation.cfm?doid=1882471.1882478

[16] Y. Chen, B. Hu, E. Keogh, and G. E. Batista, "DTW-D: Time Series Semi-Supervised Learning from a Single Example," *Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, p. 383, 2013. [Online]. Available: http://dl.acm.org/citation.cfm?doid=2487575.2487633

[17] A. Bagnall, J. Lines, J. Hills, and A. Bostrom, "Time-series classification with COTE: The collective of transformation-based ensembles," *Proceedings of 32nd ICDE International Conference on Data Engineering*, vol. 27, no. 9, pp. 1548–1549, 2016.

[18] I. Popivanov and R. J. Miller, "Similarity Search Over Time-Series Data Using Wavelets," *Proceedings 18th International Conference on Data Engineering (ICDE)*, pp. 212–221, 2002.

[19] E. Keogh, K. Chakrabarti, M. Pazzani, and S. Mehrotra, "Dimensionality Reduction for Fast Similarity Search in Large Time Series Databases," *Knowledge and Information Systems*, vol. 3, no. 3, pp. 263–286, 2001. [Online]. Available: http://link.springer.com/10.1007/PL00011669

[20] D. Berndt and J. Clifford, "Using dynamic time warping to find patterns in time series," *Workshop on Knowledge Knowledge Discovery in Databases*, vol. 398, pp. 359–370, 1994. [Online]. Available: http://www.aaai.org/Papers/Workshops/1994/WS-94-03/WS94-03-031.pdf

[21] L. Chen and R. Ng, "On The Marriage of Lp-norms and Edit Distance," in *International conference on Very large data bases*, 2004, pp. 792–803.

[22] H. Kaya and . Gündüz-Ö üdücü, "A distance based time series classification framework," *Information Systems*, vol. 51, pp. 27–42, 2015.

[23] H. Sakoe and S. Chiba, "Dynamic Programming Algorithm Optimization for Spoken Word Recognition," *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. 26, no. 1, pp. 43–49, 1978.

[24] X. Xi, E. Keogh, C. Shelton, L. Wei, and C. A. Ratanamahatana, "Fast time series classification using numerosity reduction," *Proceedings of the 23rd ICML International Conference on Machine learning*, pp. 1033—-1040, 2006. [Online]. Available: http://dl.acm.org/citation.cfm?id=1143974

[25] Y. Chen, E. Keogh, B. Hu, N. Begum, A. Bagnall, A. Mueen, and G. E. Batista, "The UCR Time Series Classification Archive," 2015. [Online]. Available: www.timeseriesclassification.com

[26] B. Calvo and G. Santafe, "scmamp: Statistical comparison of multiple algorithms in multiple problems," *The R Journal*, vol. Accepted for publication, 2015.

Reglas de Asociación en Flujos de Datos para Monitorizar Actividad de Teléfonos Móviles

Elena Ruiz, Jorge Casillas
DaSCI (Centro de Investigación en Ciencia
de Datos e Inteligencia Computacional)
Universidad de Granada, Granada, España
Email: {eruiz, casillas}@decsai.ugr.es

***Abstract*—Los algoritmos de minería de flujo de datos trabajan sobre datos con altas tasas de llegada, que evolucionan a lo largo del tiempo y requieren respuesta en tiempo real. Este tipo de técnicas, que procesan los datos al vuelo, han captado la atención tanto del ámbito científico como del industrial. El aprendizaje descriptivo en flujos de datos nos permite tener un modelo que se adapta a la evolución de los datos para explicar qué está pasando en tiempo real. En este trabajo, mostramos el potencial de este campo usando información real registrada a través de teléfonos móviles durante meses (por el MIT Human Dynamics Lab). El objetivo es evolucionar de forma dinámica reglas de asociación que expliquen la actividad del usuario en cualquier momento de forma muy eficiente para que pueda incorporarse en un dispositivo móvil. Para conseguir este objetivo empleamos un algoritmo evolutivo que aprende y mantiene de forma incremental una población de reglas de asociación.**

***Keywords*—reglas de asociación; flujo de datos; algoritmo genético; aprendizaje automático; aprendizaje online**

I. Introducción

Vivimos en la era de los datos, donde todos nuestros movimientos y actividades son registrados (o podrían serlo) y, a veces, almacenados y procesados. Obviamente, una gran parte de la información generada carece de interés. Elegir qué información es relevante, sintetizarla y extraer conocimiento de ella es, cada vez, un aspecto más crítico en la sociedad actual. En ocasiones es posible utilizar esta información para obtener modelos (data mining) que simplifican la compleja realidad que contiene dicha información.

La necesidad de extraer información relevante de fuentes de datos ordenados cronológicamente en forma de un flujo continuo, veloz y que cambia con el tiempo, excediendo las capacidades habituales de almacenamiento y procesamiento son cada vez más comunes tanto en el entorno industrial como en el científico [1]. Para solucionar este tipo de problemas es posible gestionar flujos de datos, secuencias infinitas de registros estructurados que se reciben de forma continua [1]. La característica clave de estos sistemas es que los datos producidos por estos flujos no se almacenan de forma permanente, sino que se procesan sobre la marcha. Cada dato es analizado, procesado y olvidado, haciendo posible la gestión de grandes cantidades de datos en tiempo real, incluso con capacidades de almacenamiento y procesamiento reducidas.

Los principales problemas de aprendizaje estudiados en la minería de flujo de datos [1], [2] son: (1) clasificación [3], (2) clustering [4] y (3) patrones frecuentes. En los últimos años, la mayor parte de la literatura especializada en este área se ha centrado en clasificación (y *concept-drift*); a pesar de su falta de aplicabilidad en casos reales, lo que ha derivado en experimentaciones basadas únicamente en benchmarks y datos sintéticos. Sería más realista generar modelos descriptivos e interpretables que permitan monitorear sistemas.

En general, el aprendizaje no supervisado es más directamente aplicable a problemas reales de flujo de datos, por lo que el clustering incremental ha experimentado un desarrollo significativo. Sin embargo, el conocimiento que se descubre (segmentación) resulta a menudo insuficiente para ayudar en la toma de decisiones. Por lo tanto, el descubrimiento de patrones frecuentes y reglas de asociación se considera una muy buena manera de abordar muchos problemas de flujo de datos cuyo propósito consiste en supervisar o monitorear (no predecir) en tiempo real usando modelos independientes, significativos, legibles y simples. Más concretamente, el descubrimiento de asociaciones en flujos de datos mediante la producción de reglas de asociación en un proceso completamente on-line, es particularmente interesante debido a: (1) la demanda de interpretabilidad de los patrones descubiertos en los datos, (2) la necesidad de descubrir patrones a medida que suceden, y (3) los altos y continuos volúmenes de datos a procesar, que exigen algoritmos escalables. Un caso real que supone un buen ejemplo de la utilidad de este campo es la detección de amenazas potenciales para los sitios web y las infraestructuras de red [5]. Existen otras estrategias de detección de anomalías (estadísticas o basadas en densidad), pero típicamente se basan en datos etiquetados y, por lo tanto, no se adaptan a nuevos conceptos. Otro posible caso de uso es el analizado en este documento, en el que se trabaja sobre distintos tipos de información relacionada con el uso del teléfono móvil.

El objetivo de este trabajo es mostrar el potencial de la minería de reglas de asociación en flujo de datos al tratar con varios meses de datos reales de uso de teléfonos móviles (tasa de muestreo de un minuto) proporcionados por el MIT Media Lab. El objetivo final es mantener dinámicamente un conjunto de reglas de asociación que expliquen la actividad del usuario

Este trabajo ha sido financiado por los fondos MINECO/FEDER (TIN2017-89517-P), y por el Proyecto BigDaP-TOOLS - Ayudas Fundación BBVA a Equipos de Investigación Científica 2016. E. Ruiz disfruta de un contrato vinculado al proyecto TIN2014-57251-P del MINECO.

en cualquier momento de forma muy eficiente.

El resto de este documento está organizado de la siguiente manera: La sección II presenta las principales características de Fuzzy-CSar (el algoritmo utilizado en este estudio). La sección III proporciona una descripción del estudio realizado por los investigadores del MIT y explica el proceso de preparación que hemos aplicado a los datos. La sección IV presenta los resultados obtenidos. Finalmente, la Sección V resume y concluye el trabajo.

II. Fuzzy-CSar

Fuzzy-CSar [6] está diseñado para extraer reglas de asociación difusas de flujos de datos mediante la combinación de un algoritmo genético (GA) y mecanismos de aportación de crédito de forma on-line. Es uno de los pocos algoritmos capaces de generar directamente reglas de asociación (no solo *itemsets* frecuentes) con atributos tanto cuantitativos como cualitativos de forma puramente on-line, sin emplear ventana deslizante ni ninguna otra técnica para almacenar datos. Gracias a estas propiedades, el algoritmo encaja perfectamente con el propósito de este trabajo.

Fuzzy-CSar mantiene una población de individuos, donde cada uno está representado por una regla de asociación difusa y un grupo de parámetros que evalúan la calidad de la regla. La regla de asociación difusa consiste en un antecedente y un consecuente. Se permite que el antecedente tenga un número arbitrario de atributos mientras que el consecuente consiste en un solo atributo que no debe estar presente en el antecedente de la misma regla. Cada variable puede estar representada en la regla por una disyunción de términos lingüísticos (etiquetas) para facilitar una mayor generalización. Cada individuo tiene un total de ocho parámetros de calidad.

En cada iteración del proceso de aprendizaje de Fuzzy-CSar se recibe un nuevo ejemplo y el algoritmo lleva a cabo una serie de pasos para actualizar los parámetros de los individuos de la población y descubrir nuevas reglas relevantes. Para descubrir estas nuevas reglas prometedoras, se aplica un algoritmo genético estacionario basado en nichos [7]. Además, se aplican operadores de cruce, distintos tipos de mutación y *covering* con ciertas probabilidades. Podemos ver un esquema de la fase de aprendizaje de Fuzzy-CSar en el algoritmo 1. Complementariamente, explicamos brevemente algunos componentes de la fase de aprendizaje, una explicación más detallada se puede encontrar en [6].

Algoritmo 1: Esquema de la fase de aprendizaje de Fuzzy-CSar [6]

```
proceso TrainFuzzy-CSar( ejemploEntrenamiento e_t,
 Población [P] en el instante t )
Data: e_t tiene la forma {x_i}_{i=1}^{ℓ}
Result: Población [P] en el instante t + 1
begin
    e'_t ← granulación(e_t);
    genera [M] a partir de [P] usando e'_t;
    if |[M]| < θ_mna then
        genera θ_mna − |[M]| individuos que concuerdan usando e'_t y
          actualizando [P];
    end
    agrupa individuos en [M] por su antecedente formando distintos [A]_i;
    selecciona [A] según probabilidad;
    subsume individuos en [A];
    actualiza individuos en [M] ; // Por lo tanto, todos los
      [A]_i se actualizan.
    if el tiempo medio en [A] desde la última vez de GA > θ_GA then
        se lleva a cabo un evento genético en [A] considerando e'_t y
          actualizando [P];
    end
end
```

A. Parámetros de Calidad: Soporte y Confianza

En Fuzzy-CSar tenemos ocho parámetros de calidad para cada individuo de la población. Vamos a explicar cómo se calculan dos de ellos, los dos más utilizados y los más significativos para este trabajo: soporte y confianza (información sobre el cálculo de otros parámetros en [6]).

Si definimos formalmente el campo de minería de reglas de asociación como sigue [8]: Siendo $I = i_1, i_2, ..., i_l$ un conjunto de características binarias (ítems) de l elementos. Siendo $Tr = tr_1, tr_2, ..., tr_N$ un conjunto de N transacciones donde cada transacción tr_j contiene un vector binario que indica en cada posición si un ítem en particular está presente o no. Entonces un ítem X ($X \subset I$) va a tener asociado un soporte que es una medida de su importancia en T y se calcula como $supp(X) = |X(T)|/|T|$, donde $X(T)$ es el conjunto de variables en el antecedente de la regla. Si el soporte de un determinado *itemset* (conjunto de ítems) supera un umbral definido por el usuario (*minsupp*) este *itemset* se considera como un *conjunto frecuente de ítems*. Si X e Y son ambos conjuntos frecuentes de ítems y $X \cap Y = \emptyset$, podemos definir una regla de asociación como una implicación del tipo $X \to Y$. El soporte y la confianza de una regla de asociación son las medidas cualitativas tradicionalmente más usadas:

$$supp(X \to Y) = \frac{supp(X \cup Y)}{|T|}, conf(X \to Y) = \frac{supp(X \cup Y)}{supp(X)}. \quad (1)$$

Donde el soporte indica la frecuencia con la que se cumplen los patrones y la confianza evalúa la fuerza de la implicación indicada en la regla de asociación.

Sea $I = \{i_1, i_2, \ldots i_\ell\}$ un conjunto de ℓ características, $A \subset I$, $C \subset I$ y $A \cap C = \emptyset$. Una regla de asociación difusa es una implicación de la forma $X \to Y$ en la cual:

$$X = \bigwedge_{i_i \in A} \mu_{\widetilde{A}}(i_i) \quad \text{e} \quad Y = \bigwedge_{i_j \in C} \mu_{\widetilde{C}}(i_j), \quad (2)$$

donde $\mu_{\widetilde{C}}(i_j)$ es el grado de pertenencia de la variable en el consecuente y $\mu_{\widetilde{A}}(i_i)$ es el grado de pertenencia de las variables del antecedente. En esta situación, el soporte se extiende usando el producto T-norm y la confianza se extiende usando la *implicación de Dienes* [9]:

$$supp(X \to Y) = \frac{1}{|T|} \sum \mu_{\widetilde{A}}(X) \cdot \mu_{\widetilde{C}}(Y) \quad (3)$$

$$conf(X \to Y) = \frac{\sum(\mu_{\widetilde{A}}(X) \cdot \max\{1 - \mu_{\widetilde{A}}(X), \mu_{\widetilde{C}}(Y)\})}{\sum \mu_{\widetilde{A}}(X)}. \quad (4)$$

donde $\mu_{\widetilde{A}}(X)$ es el grado de pertenencia de la parte del antecedente de la regla y $\mu_{\widetilde{C}}(Y)$ es el grado de pertenencia de la parte del consecuente de la regla.

B. Operador de Covering

Este operador genera nuevas reglas de asociación difusas cuando hay menos de θ_{mna} (siendo θ_{mna} un parámetro de configuración) individuos en el *match set [M]*, individuos de la población actual que concuerdan con el ejemplo recibido e.

El operador de covering construye un nuevo individuo que concuerda con e en el máximo grado posible: para cada variable de entrada de e, e_i, el operador decide aleatoriamente si e_i va a formar parte del antecedente de la regla. Después, elige la variable que estará en el consecuente de entre aquellas que no han sido seleccionadas para formar parte del antecedente.

C. Subsunción de Reglas

Para cada regla de $[A]$ se comprueba su posible subsunción con cada una de las otras reglas del conjunto. Una regla r_i es una candidata para subsumir r_j si: (1) r_i es más general que r_j, y (2) ambas reglas tienen confianzas similares y r_i tiene la suficiente experiencia. Una regla r_i es considerada más general que r_j si todas las variables de r_i están también definidas en r_j y, para cada una de estas variables, r_i tiene, al menos, los mismos términos lingüísticos que r_j.

D. Descubrimiento de Nuevas Reglas

Fuzzy-CSar usa un algoritmo genético incremental estacionario basado en nichos para descubrir nuevas reglas. Este algoritmo genético que se aplica al *association set* seleccionado, cuenta con tres tipos diferentes de mutación: (1) mutación del antecedente; (2) mutación del consecuente, y (3) mutación de términos lingüísticos.

En Fuzzy-CSar, como es común entre los miembros de la familia Michigan-style LCS [10], el coste del algoritmo se incrementa linealmente con el tamaño máximo de la población de reglas, el máximo número de variables por regla, y semilogarítmicamente con el coste de ordenar el *match set*. Es importante resaltar que Fuzzy-CSar no depende directamente del número de transacciones, lo que lo hace muy adecuado para trabajar con grandes bases de datos. En el estudio aquí presentado, el algoritmo se aplica en un problema real (el cual se explica a continuación). La eficiencia es un requisito muy importante en minería de flujo de datos. En el problema aquí abordado, hemos estimado que el tiempo medio que Fuzzy-CSar tarda en procesar cada dato es de unos 15 milisegundos, siendo así factible procesar más de 60 muestras por segundo.

III. Estudio 'Friends and Family'

A. Datos Originales

El *estudio Friends and Family*[1] es una investigación llevada a cabo por el MIT Media Lab, durante los años 2010 y 2011 [11]. Este estudio transforma una comunidad residencial cercana a una conocida universidad norteamericana en un *laboratorio viviente* durante 15 meses. Durante estos meses, los investigadores del Media Lab presentaron su sistema de registro de interacciones sociales y comportamiento basado en teléfonos móviles. Durante cerca de un año, toda actividad, comunicación y detalle social de las vidas de un gran número de miembros de la mencionada comunidad fue registrado mientras ellos realizaban sus tareas cotidianas con normalidad. Un total de 130 sujetos formaron parte del estudio. Durante el período del estudio, se recogió una gran cantidad de datos que resultó en un conjunto de datos muy completo y longitudinal, bautizado como *Friends and Family dataset*. Dicho conjunto de datos incluye una gran colección de señales basadas en la actividad del teléfono móvil incluyendo comunicación (llamadas y mensajes), aplicaciones instaladas, ejecución de aplicaciones, acelerómetro, dispositivos bluetooth próximos...

El estudio se dividió en dos fases. Una fase piloto de 6 meses de duración que comenzó en marzo de 2010, y una segunda fase iniciada en septiembre de 2010. Hasta 130 sujetos participaron en esta segunda fase.

Se proporcionaron *smartphones* a los participantes del estudio con la condición de que estos debían ser sus teléfonos principales durante su participación en el estudio. Estos dispositivos harían el papel de sensores sociales *in-situ* para registrar las características de la actividad de los sujetos.

Parte de la colección de datos obtenida del "Estudio Friends and Family" fue publicada y ha servido como punto de partida para nuestro estudio. El tamaño de esta parte publicada de la colección supera los 7GB. Además, esta gran cantidad de datos está distribuida en distintos archivos cuyo origen, formato y estructura varían.

Los datos recogidos de los teléfonos móviles son el núcleo principal de la colección de datos construida a partir del estudio. La tabla I enumera algunos tipos de información incluidos en esta colección y utilizados en nuestro estudio, junto con sus frecuencias de muestreo originales.

Tabla I: Principales tipos de información incluidos en los datos originales y usados en nuestro estudio

Información	Frecuencia muestreo
Dispositivos bluetooth próximos	cada 5 minutos
Registro de llamadas	cuando una llamada es enviada/recibida
Registro de SMS	cuando un SMS es enviado/recibido
Aplicaciones en el dispositivo	cada 10 minutos
Aplicaciones en ejecución	cada 30 segundos

Dado el origen y las peculiaridades de la información recogida, el estudio se llevó a cabo bajo estrictos protocolos que aseguran la privacidad de todos los participantes.

B. Preparación del Flujo de Datos

La estructura original de los datos no era la más adecuada para el proceso de extracción de conocimiento ni para obtener resultados de calidad. Por lo tanto, los datos tuvieron que ser tratados antes de aplicar el algoritmo. Los datos se encontraban distribuidos en diferentes archivos con diferente formato, estructura y frecuencia de muestreo dependiendo del tipo de información y del método de recolección utilizado. Cada tipo de información había sido registrado con sus propias peculiaridades, los datos de todos los participantes estaban mezclados y no todos los sujetos estuvieron implicados al mismo nivel en el estudio. Fue necesario aplicar un proceso de preparación sobre los datos en bruto para obtener un conjunto de datos completo, unificado y específico para cada sujeto.

[1] http://realitycommons.media.mit.edu/friendsdataset.html

En primer lugar, se realizó un estudio exploratorio de los datos para conocer y comprender mejor el problema. No toda la información original resulta útil para nuestros objetivos. Es necesario decidir qué información es relevante y cuál no.

Un algoritmo evolutivo completamente en línea presenta algunas peculiaridades dado que cada dato va a ser procesado una sola vez y el algoritmo no va a tratar con el conjunto de datos completo en ningún momento. Esto puede hacer que cierta información aparezca solo por un momento para acabar desapareciendo para el algoritmo a pesar de que puedan proporcionar información relevante. Tratando de minimizar este riesgo, se generaron variables del tipo "cuántas llamadas se han registrado en los últimos X minutos". Finalmente, se eligió una frecuencia de muestreo unificada de un minuto para los conjuntos de datos finales tratando de no perder demasiado detalle pero, al mismo tiempo, intentando evitar que el nivel de granularidad de los datos se vuelva innecesariamente bajo.

Para clarificar y resumir, describimos cada paso de la preparación de datos.

1) **Filtrado de datos por fecha**: se seleccionan los datos a partir del 01/10/2010, excluyendo los de la fase piloto.
2) **Agrupamiento de datos por participante**: se agrupa todos los datos de cada participante.
3) **Duplicados y orden cronológico**: eliminamos registros duplicados y ordenamos cronológicamente.
4) **Integración de datos**: los datos libres de duplicados y ordenados cronológicamente recogidos durante la segunda fase del estudio para cada participante se integran en conjuntos de datos individuales.

La figura 1 representa la evolución en el tiempo para los principales atributos después del proceso de preparación de los datos. Los datos presentados en dicha figura corresponden a un sujeto elegido como ejemplo para el análisis de resultados (Sección IV). El gráfico de la figura 1 tiene una resolución diaria, por lo que debemos tener en cuenta que los valores mostrados para las variables acumulativas (llamadas y contadores de mensajes) en los gráficos para un día dado son diez veces el número real de llamadas/mensajes. Esta figura nos ayuda a entender lo difícil que sería extraer información útil y descubrir asociaciones interesantes sin la asistencia proporcionada por el algoritmo de minería de flujo de datos.

IV. Experimentación y Resultados

A. Diseño de Experimentos

Los atributos de entrada utilizados en la experimentación realizada para obtener los resultados presentados en este trabajo son los siguientes: día de la semana; minuto del día [0, 1439]; porcentaje de batería; tres contadores de SMS en los últimos 10 minutos (entrantes, salientes y global); cuatro contadores de llamadas durante los últimos 10 minutos (entrantes, salientes, perdidas y global); tres variables referentes a acelerómetro; alguna aplicación desinstalada y alguna aplicación en ejecución.

La tabla II muestra los valores asignados a los principales parámetros de configuración de Fuzzy-CSar para los experimentos (cuyos resultados se discuten a continuación).

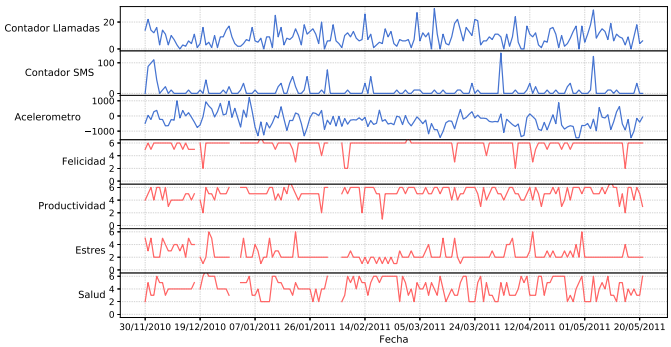


Figura 1: Representación gráfica de la evolución de algunos atributos que forman el conjunto de datos de uno de los participantes (Identificador del participante: sp10-01-24)

Tabla II: Principales parámetros de configuración de Fuzzy-CSar junto con los valores asignados en los experimentos

Parámetro	**Valor**
Tamaño máximo población	10.000
Comprobar subsunción	Sí
Tipo de *association sets*	Antecedente
Número máximo de conjuntos difusos	Dependiente del rango de la variable
Forzar etiquetas adyacentes	Sí
Número máximo de término lingüísticos	60% de los conjuntos difusos
Mutación	Sí
Comprobar subsunción en el GA	Sí

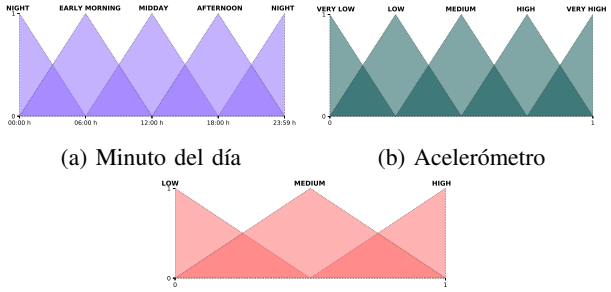
B. Análisis de Resultados

Como se ha explicado, Fuzzy-CSar mantiene continuamente una población de reglas. Es complicado encontrar una manera de trazar la evolución completa de toda esta población. Para poder representar y analizar esta evolución, nos centramos en ciertas reglas completas, consecuentes y antecedentes.

La figura 2 puede ayudar a entender mejor la descripción de algunas reglas de asociación que se muestran en esta sección. En ella es posible observar un esquema de la nomenclatura empleada para referirnos a los conjuntos difusos usados por Fuzzy-CSar para atributos numéricos.

Según estas nomenclaturas, la tabla III muestra tres reglas obtenidas por Fuzzy-CSar. Estos ejemplos ayudan a entender y visualizar mejor la estructura de las reglas de asociación obtenidas por Fuzzy-CSar. En la figura 3 podemos ver la evolución del número de copias almacenadas en la población (numerosidad), lo que representa la importancia relativa de la regla en cada momento, para cada una de las reglas representadas en la tabla III. Podemos observar cómo esta evolución es completamente diferente para cada regla. R_1 (rojo) aparece en un cierto momento, luego la numerosidad de la regla aumenta, para posteriormente disminuir hasta que la regla desaparece. Pero después de eso, la regla aparece de nuevo y repite el mismo proceso. Sin embargo, la regla R_2 (azul) aparece cerca del mes de abril para continuar aumentando su numerosidad hasta el final del experimento. Finalmente, la regla R_3 (verde) existe en la población desde el principio y su numerosidad es siempre creciente. Si analizamos la asociación representada por R_3 esta evolución parece muy lógica ya que normalmente

la gente duerme por la noche y no hace ninguna llamada por lo que esta condición es casi siempre cierta.



(a) Minuto del día (b) Acelerómetro

(c) Llamadas y SMS

Figura 2: Nomenclatura empleada para los conjuntos difusos en las variables referentes a: (a) minutos del día, (b) acelerómetro, y (c) contadores de llamadas y mensajes

Tabla III: Ejemplos de reglas generadas por Fuzzy-CSar durante los experimentos (figura 3)

ID	Rule	Soporte*	Confianza*
R_1	Si hora es MIDDAY o AFTERNOON entonces acelerómetro es MEDIUM, HIGH, VERY HIGH	0.124	0.802
R_2	Si número de SMS salientes es LOW entonces acelerómetro es HIGH, VERY HIGH	0.606	0.709
R_3	Si hora es NIGHT o EARLY MORNING y número de SMS entrantes es LOW entonces número de llamadas salientes es LOW	0.496	0.997

* El soporte y confianza se refieren al momento con máximo número de copias

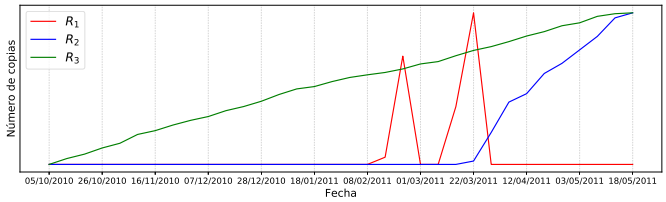


Figura 3: Evolución del número de copias en la población (numerosidad) de: R_1 (máximo 21 copias), R_2 (máximo 90 copias) y R_3 (máximo 537 copias) (tabla III)

Generalizando un poco nuestro análisis podemos centrarnos en un consecuente determinado en lugar de en una regla determinada. Siguiendo esta idea presentamos una serie de gráficos en los que analizamos la evolución en numerosidad de un conjunto específico de reglas que contienen la misma variable en su consecuente o incluso la misma etiqueta. Como prueba de concepto, a continuación se analiza la evolución de la cantidad de actividad física practicada por el participante.

La variable *AccelAccum*, incluida en el conjunto de datos usado, presenta valores relacionados con la información de acelerometría recogida de los *smartphones* de los participantes. Los investigadores del MIT relacionan dicha información de acelerometría con la actividad física, como también lo hacen otros investigadores [12]. Siguiendo esta interpretación,

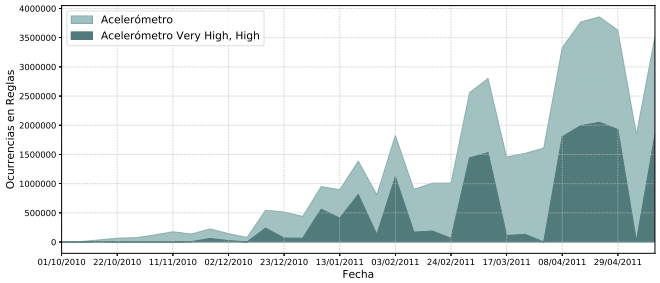


Figura 4: Comparativa entre la evolución en el nº total de reglas con *AccelAccum* en el consecuente y el número de ellas con la etiqueta VERY HIGH, HIGH ($supp \geq 0.1$ y $conf \geq 0.7$). Datos del participante *sp10-01-24*

entendemos que un mayor número de reglas cuyos consecuentes apuntan a valores *HIGH, VERY-HIGH* de la variable *AccelAccum* puede ser interpretado como un indicador de un aumento de la actividad física practicada por el sujeto. La figura 4 representa la evolución en el número de reglas, que superan los umbrales de soporte y confianza, cuyos consecuentes contienen la variable *AccelAccum* con cualquier etiqueta (área color claro) y las reglas cuyos consecuentes contienen valores altos de la variable *AccelAccum* (área color oscuro). Teniendo esto en cuenta, el gráfico representado en la figura 4 muestra cómo el participante *sp10-01-24* aumenta la cantidad de actividad física durante ciertos períodos de tiempo.

La figura 5 nos ayuda a entender la distribución de los valores de *AccelAccum* durante un cierto periodo de tiempo. Como se observa, no hay concentraciones especiales de valores altos o muy altos. El algoritmo comienza a encontrar relaciones que explican los altos valores de *AccelAccum* y, como consecuencia, aumenta el número de reglas con este consecuente específico y suficiente confianza.

Dado que se ha utilizado un algoritmo de reglas de asociación, y no uno de patrones frecuentes, podemos utilizar las relaciones establecidas por las reglas de asociación entre antecedentes y consecuentes como parte de nuestro análisis.

Así, profundizamos en los resultados analizando la composición del antecedente de las reglas con altos valores de *AccelAccum* en el consecuente. La figura 6 representa la evolución de estas reglas de calidad distinguiendo entre variables en el antecedente. En esta figura se muestra cómo la mayoría de los antecedentes se relacionan con la no utilización de varias funciones del teléfono móvil, por ejemplo, llamadas, mensajes, aplicaciones.... Los antecedentes agrupados bajo la etiqueta "Otro" se relacionan principalmente con la hora y el día de la semana. Este hecho refuerza la idea de que los valores altos del acelerómetro están relacionados con la actividad física y no con otros tipos de uso del *smartphone*.

V. Conclusiones

En este trabajo hemos mostrado una aplicación real de un algoritmo evolutivo (Fuzzy-CSar) para minería de reglas de asociación en flujo de datos, cuyo objetivo es descubrir de forma on-line y en tiempo real las relaciones entre los atributos

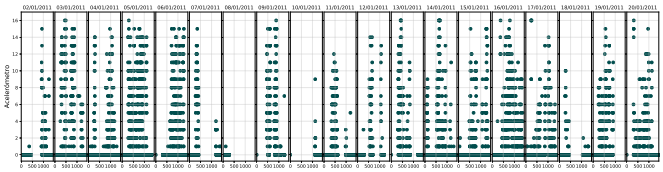


Figura 5: Distribución de los valores de la variable *AccelAccum* a lo largo de los minutos de cada día del 2 al 21 de enero de 2011. Datos del participante *sp10-01-24*

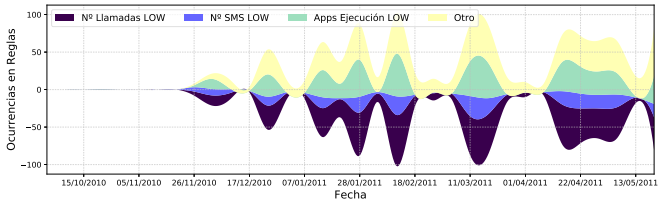


Figura 6: Comparación entre la evolución del nº total de reglas ($supp \geq 0.1$ y $conf \geq 0.7$) cuyo consecuente contiene *AccelAccum* con etiqueta VERY HIGH, HIGH y el nº de ellas con antecedentes referidos a la inactividad del teléfono. Datos del participante *sp10-01-24*

de un flujo de datos. Esta aplicación simula un sistema de monitorización en el que los datos de entrada consisten en muestras reales basadas en información sobre llamadas, SMS, acelerómetros, aplicaciones, etc. recogida desde *smartphones*. Estos datos constituyen un claro ejemplo de problema real en el que la información se genera como un flujo continuo e infinito y puede ser explotada de forma muy productiva sin necesidad de ser almacenada en grandes conjuntos de datos que requieran altas capacidades de procesamiento.

En este caso, el algoritmo evolutivo descubre reglas en tiempo real para explicar lo que está sucediendo en todo momento. Los resultados muestran la evolución que está experimentando la población de reglas de asociación a medida que aumenta la cantidad de datos procesada. Estos son solo algunos ejemplos que muestran cómo el algoritmo Fuzzy-CSar y las reglas de asociación descubiertas por él, hacen posible descubrir nuevas relaciones que no habrían sido descubiertas directamente a partir de datos en bruto. Además, esto se consigue de una manera muy eficiente, ya que Fuzzy-CSar tarda solo 15 ms en procesar cada dato, es decir, en el caso específico de datos con esta estructura es capaz de lidiar con una frecuencia de entrada de unos 67 Hz. En un entorno donde los datos llegan en tiempo real en forma de flujo infinito, como en este trabajo, es posible generar y actualizar un modelo en tiempo real, permitiendo un proceso de monitoreo que puede ayudar en un sistema de toma de decisiones. Otro posible caso de uso podría ser integrar este algoritmo en una aplicación móvil que utilice directamente el conocimiento descubierto por el algoritmo para tomar decisiones (por ejemplo, recomendaciones musicales o sugerencia de aplicaciones) basadas en reglas específicas. Dado que el algoritmo no asume ninguna estructura de problema a priori, es capaz de adaptarse a las características de los datos de cada sujeto. Cabe destacar la importancia de utilizar un algoritmo con capacidad para adaptarse a los cambios conceptuales, ya que a menudo estos cambios proporcionan la información más relevante.

En conclusión, en este trabajo se muestra el potencial de la minería de reglas de asociación en flujo de datos para la monitorización de problemas reales. El trabajo realizado y los resultados obtenidos revelaron que el desarrollo de un sistema capaz de monitorizar el uso que una persona está haciendo de su teléfono a través de un algoritmo de minería de asociaciones en flujos de datos (utilizando técnicas on-line e incrementales), descubriendo información útil, es una opción factible. En este momento, se están considerando varias líneas para continuar con este trabajo, entre las que se incluyen las siguientes: (1) continuar con las mejoras de Fuzzy-CSar y pulir las adaptaciones del algoritmo a los datos de actividad de teléfonos móviles, (2) estudiar más profundamente los resultados obtenidos en este conjunto de datos, (3) aprovechar la buena eficiencia de Fuzzy-CSar para la integración en dispositivos móviles, y (4) desarrollar aplicaciones para obtener más datos de este tipo utilizando luego la información relevante descubierta a través de las asociaciones aprendidas.

References

[1] J. Gama, *Knowledge discovery from data streams*. Chapman & Hall/CRC, 2010.

[2] M. Sayed-Mouchaweh and E. Lughofer, *Learning in non-stationary environments : methods and applications*. Springer, 2012.

[3] A. Orriols-Puig and J. Casillas, "Fuzzy knowledge representation study for incremental learning in data streams and classification problems," *Soft Computing*, vol. 15, no. 12, pp. 2389–2414, dec 2011.

[4] S. Guha, A. Meyerson, N. Mishra, R. Motwani, and L. O'Callaghan, "Clustering data streams: theory and practice," *IEEE Transactions on Knowledge and Data Engineering*, vol. 15, no. 3, pp. 515–528, may 2003.

[5] G. Corral, A. Garcia-Piquer, A. Orriols-Puig, A. Fornells, and E. Golobardes, "Analysis of vulnerability assessment results based on CAOS," *Applied Soft Computing*, vol. 11, no. 7, pp. 4321–4331, oct 2011.

[6] A. Sancho-Asensio, A. Orriols-Puig, and J. Casillas, "Evolving association streams," *Information Sciences*, vol. 334-335, pp. 250–272, mar 2016.

[7] S. W. Wilson, "Classifier Fitness Based on Accuracy," *Evolutionary Computation*, vol. 3, no. 2, pp. 149–175, jun 1995.

[8] R. Agrawal, T. Imieliński, A. Swami, R. Agrawal, T. Imieliński, and A. Swami, "Mining association rules between sets of items in large databases," in *Proceedings of the 1993 ACM SIGMOD international conference on Management of data - SIGMOD '93*, vol. 22, no. 2. New York, New York, USA: ACM Press, 1993, pp. 207–216.

[9] D. Dubois, E. Hüllermeier, and H. Prade, "A systematic approach to the assessment of fuzzy association rules," *Data Mining and Knowledge Discovery*, vol. 13, no. 2, pp. 167–192, sep 2006.

[10] A. Orriols-Puig, J. Casillas, and F. J. Martínez-López, "Unsupervised Learning of Fuzzy Association Rules for Consumer Behavior Modeling," *Mathware & Soft Computing*, vol. 16, pp. 29–43, 2009.

[11] N. Aharony, W. Pan, C. Ip, I. Khayal, and A. Pentland, "Social fMRI: Investigating and shaping social mechanisms in the real world," *Pervasive and Mobile Computing*, vol. 7, no. 6, pp. 643–659, dec 2011.

[12] J. Bort-Roig, N. D. Gilson, A. Puig-Ribera, R. S. Contreras, and S. G. Trost, "Measuring and Influencing Physical Activity with Smartphone Technology: A Systematic Review," *Sports Medicine*, vol. 44, no. 5, pp. 671–686, may 2014.

Un Sistema de Recomendación de Asignaturas Multi-Criterio con Optimización Genética

Aurora Esteban, Amelia Zafra, Cristóbal Romero
Departamento de Informática y Análisis Numérico
Universidad de Córdoba
Córdoba, España

***Resumen*—Este artículo propone un Sistema de recomendación (SR) híbrido multi-criterio para resolver el problema de recomendación de asignaturas en los estudios universitarios. Por un lado, utiliza Filtrado Colaborativo (FC) con la información del estudiante y por otro lado, Filtrado Basado en Contenido (FBC) con la información de las asignaturas. Para determinar los factores que resultan más relevantes en la recomendación, así como para optimizar la configuración del SR híbrido en lo referente a las medidas de distancia a considerar y el tamaño del vecindario, se ha diseñado un Algoritmo Genético (AG) que optimiza de forma automática el ajuste de todos los parámetros. Un estudio experimental con 2500 valoraciones reales de los estudiantes del Grado en Ingeniería Informática de la Universidad de Córdoba, demuestra los excelentes resultados obtenidos por el SR propuesto y la importancia de utilizar un modelo híbrido donde los diferentes criterios utilizados sean ponderados de acuerdo a su relevancia.**

***Index Terms*—Sistema de recomendación, Filtrado colaborativo, Filtrado basado en contenido, Algoritmo genético**

I. Introducción

En la era de la información que recientemente se vive, los Sistemas de Recomendación (SR) se han consolidado en diversos campos, entre los que se encuentran los entornos educativos. Específicamente, la recomendación de asignaturas o cursos se ha afianzado como una interesante y creciente línea de investigación dentro de la minería de datos aplicada a la educación [1]. En el caso concreto de la recomendación de asignaturas en los estudios universitarios, su importancia viene dada por el hecho de que los estudios universitarios tienen un número variable de asignaturas optativas entre las que los estudiantes deben elegir para completar los créditos de la titulación que están cursando. Normalmente, los estudiantes no cuentan con información suficiente para hacer esta elección, y recurren a compañeros que la hayan cursado para conocer su opinión. En este contexto, los SR se presentan como una herramienta esencial, capaz de ofrecerles asignaturas relevantes en base a sus preferencias individuales, sus intereses, sus necesidades o su rendimiento [2].

Aunque existen algunos estudios que trabajan con enfoques de SR híbridos [3], [4] o con múltiples criterios [5], [6], no incluyen todos los criterios aquí considerados, ni se centran en estudiar la influencia que los diferentes factores tienen en el proceso de recomendación.

En este trabajo se presenta un SR híbrido multi-criterio, que combina una técnica de Filtrado Colaborativo (FC) utilizando información del estudiante (considerando como criterios, sus valoraciones, sus calificaciones y la especialidad cursada dentro de la carrera), con una técnica de Filtrado basado en Contenido (FBC) utilizando información de las asignaturas (considerando como criterios, sus competencias, profesores, contenidos teóricos o prácticos y área de conocimiento). Para determinar la importancia de cada criterio se propone un Algoritmo Genético (AG) que optimice los pesos asignados a cada uno de estos criterios en el SR. Así mismo, otros parámetros del SR como las métricas de similitud utilizadas, o el número de vecinos también serán optimizadas simultáneamente. En la metodología propuesta, el SR es configurado con los parámetros optimizados por el AG, para finalmente, realizar las recomendaciones finales a los estudiantes.

Con el fin de garantizar una evaluación rigurosa del SR, se va a utilizar un conjunto de datos proveniente del Grado en Ingeniería Informática de la Universidad de Córdoba (España), que incluye valoraciones y calificaciones de estudiantes reales. Así mismo, en el proceso de evaluación se utilizará validación cruzada manteniendo el equilibrio entre las particiones de datos.

El resto del trabajo se organiza como sigue. En la sección II se hace un repaso de los trabajos previos. En la sección III se especifica la metodología propuesta, describiendo tanto la información, como el SR híbrido diseñado. En la sección IV se explica el AG utilizado para la optimización. En la sección V se describe el estudio experimental realizado, tanto el estudio de la relevancia de los criterios, como la comparativa con otros modelos. Finalmente, la sección VI presenta las conclusiones obtenidas y el trabajo futuro.

II. Trabajos relacionados

En los últimos años los SR multi-criterio han sido ampliamente aplicados a la recomendación de asignaturas o cursos. S. Spiegel [7] explora una de las primeras aplicaciones de factorización de matriz multi-criterio para la predicción de valoraciones de asignaturas. Más adelante, Vialardi et al. [5] propusieron técnicas multi-criterio para la predicción de calificaciones de estudiantes abordadas

como un problema de clasificación, y Parameswaran et al. [8] exploraron la aplicación de restricciones a la recomendación multi-criterio. Así mismo, se ha explorado la aplicación de otras técnicas, como propuestas basadas en ontologías [9], [10], redes neuronales [11], y algoritmos bio-inspirados como colonias de hormigas [12] o sistemas inmunes artificiales [3]. La mayoría de estas propuestas sólo están basados en las valoraciones de los estudiantes. Desde otra perspectiva, la importancia del momento en el que se cursan las asignaturas ha sido estudiada usando Cadenas de Markov en base a las calificaciones [13], así como aplicando multi-criterio [14]. Recientemente, se ha estudiado la relevancia de las competencias que las asignaturas ofrecen [6] o la aplicación de análisis semántico [15].

Se puede resumir que la mayoría de los enfoques se centran principalmente en el rendimiento de los estudiantes, y no usan criterios adicionales. Además, los sistemas que utilizan más criterios, no hacen ningún estudio para determinar la influencia de cada uno de ellos en la calidad de las recomendaciones, mostrando así los beneficios que aporta nuestra propuesta.

III. Propuesta de Sistema de Recomendación Híbrido

En esta sección se describe la metodología propuesta, especificando desde la recolección y preparación de los datos, hasta el desarrollo de la propuesta planteada.

A. Descripción y preparación de los datos

Este trabajo se ha desarrollado utilizando los datos recogidos del Grado en Ingeniería Informática de la Universidad de Córdoba (España). Estos datos son relativos tanto a los estudiantes, como a las asignaturas.

1) Información de estudiante: la información de los estudiantes (figura 1) se ha obtenido por medio de encuestas realizadas durante tres cursos académicos (de 2016 a 2018) a los alumnos de los últimos cursos, recogiendo un total de 2500 valoraciones de 63 asignaturas incluidas en el plan de estudios. La información que se ha considerado es:

- Una valoración de la satisfacción general del estudiante en cada asignatura, representada como un valor numérico entero entre 1 y 5. El valor máximo correspondería a un 5, y el mínimo a un 1.
- La calificación obtenida por el estudiante para cada asignatura, representada con un valor numérico real, en el rango $[0, 10]$.
- La especialidad seleccionada por el estudiante. Concretamente, el grado en Ingeniería Informática que se estudia, ofrece tres especialidades: computación, computadores e ingeniería del software, representados con un identificador numérico de 1 a 3.

2) Información de la asignatura: la información de las 63 asignaturas consideradas en el plan de estudios (figura 2), se ha obtenido de la página oficial del grado dentro de la Universidad[1]. La información que se ha considerado es:

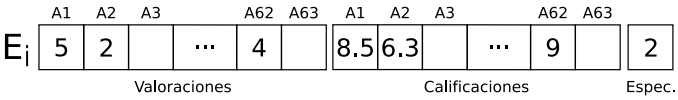


Figura 1. Información del estudiante

- Los profesores que imparten docencia en cada asignatura, representados con un vector binario, indicando con el valor 1 si el profesor ha impartido la asignatura y con el valor 0 si no la ha impartido.
- Las competencias que se deben adquirir al realizar la asignatura, representadas con un vector binario, indicando con el valor 1 que la asignatura proporciona la competencia, y con el valor 0 que no la proporciona.
- El área de conocimiento al que la asignatura pertenece, representada con un identificador numérico de 1 a 8, ya que en el grado estudiado se contemplan ocho áreas de conocimiento diferentes.
- Los contenidos teóricos y prácticos de la asignatura, representados con un vector de frecuencias de las palabras claves. Las palabras claves son obtenidas a partir del procesamiento automático de textos aplicado a la guía docente de cada asignatura.

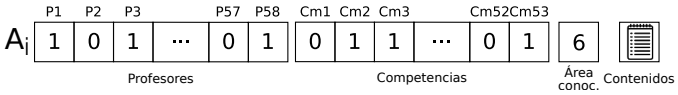


Figura 2. Información de la asignatura

B. Sistema de Recomendación

El SR híbrido multi-criterio propuesto en este trabajo combina las valoraciones obtenidas por un sistema basado en FC (especificado en la sección III-B1 y las obtenidas por un sistema de FBC (especificado en la sección III-B2). De este modo, la estimación de preferencia p de un estudiante i sobre una asignatura j, se calcula como la combinación lineal de las preferencias dadas por el FC ($FC_{i,j}$) y el FBC ($FBC_{i,j}$):

$$p_{i,j} = \alpha \cdot FC_{i,j} + \beta \cdot FBC_{i,j} \quad (1)$$
$$Con\ \alpha + \beta = 1$$

Tanto en el sistema FC, como en el FBC se representan las valoraciones estimadas de cada asignatura en un rango de $[1, 5]$, por lo que la estimación final también estará en dicho rango.

Se puede apreciar en la ecuación 1, que los parámetros que determinan la relevancia que se le da a la valoración obtenida por cada uno de los sistemas considerados son los pesos α y β, que deben ser configurados.

[1] http://www.uco.es/eps/node/619

En los siguientes apartados se especifican las características del sistema de FC y del sistema de FBC que se han diseñado.

1) Filtrado Colaborativo utilizando la información del estudiante: en el sistema de FC desarrollado, la valoración estimada de una asignatura para un estudiante se obtiene a partir de las valoraciones que ésta ha recibido por parte del resto de estudiantes con un perfil similar.

El aspecto multi-criterio es introducido en el cálculo de la similitud entre estudiantes: para cada par de estudiantes i y j, la similitud, s, agrega tres medidas de similitud que están en el rango $[0, 1]$: por un lado, calcula la similitud según sus valoraciones $V_{i,j}$ y sus calificaciones $C_{i,j}$, por otro lado, comprueba si la especialidad que están cursando ambos estudiantes coincide ($E_{i,j}$). Finalmente, los tres criterios se agregan en una combinación lineal:

$$s_{i,j} = \alpha \cdot V_{i,j} + \beta \cdot C_{i,j} + \gamma \cdot E_{i,j} \quad (2)$$
$$Con\ \alpha + \beta + \gamma = 1$$

Por tanto, será necesario configurar las métricas utilizadas para valoraciones y calificaciones, el vecindario, y los pesos α, β y γ de los criterios considerados.

2) Filtrado basado en Contenido utilizando la información de la asignatura: en el sistema de FBC desarrollado, la valoración estimada de una asignatura para un estudiante se obtiene a partir de la valoración que él mismo ha hecho de las asignaturas más similares a ésta.

El aspecto multi-criterio es introducido en el cálculo de la similitud entre asignaturas: para cada par de asignaturas i y j, la similitud (s) es una combinación de cuatro medidas que están en el rango $[0, 1]$: por un lado se calcula cuántos profesores y competencias tienen en común, $P_{i,j}$ y $Cm_{i,j}$, respectivamente. Por otro lado, se comprueba si el área de conocimiento coincide $A_{i,j}$, y finalmente, se obtiene la similitud según los contenidos aplicando minería de textos $Cn_{i,j}$, siguiendo el procedimiento especificado más adelante. Finalmente, se calcula la similitud final mediante una combinación lineal:

$$s_{i,j} = \alpha \cdot P_{i,j} + \beta \cdot Cm_{i,j} + \gamma \cdot A_{i,j} + \delta \cdot Cn_{i,j} \quad (3)$$
$$Con\ \alpha + \beta + \gamma + \delta = 1$$

Por tanto, para aplicar FBC será necesario determinar las métricas usadas para $P_{i,j}$ y $Cm_{i,j}$ y la relevancia de cada criterio por medio de los pesos α, β, γ y δ.

La similitud según los contenidos de la asignatura es obtenida a partir de la información descrita en su guía docente mediante minería de textos. El proceso compara los contenidos de dos asignaturas, para finalmente dar un valor de similitud entre 0 y 1:

1) Indexación del apartado *Contenidos* de las guías docentes de las asignaturas: este proceso se lleva a cabo con un analizador sintáctico y con un conjunto de *stop-words* específicas al dominio. Como resultado se obtiene para cada documento un conjunto de *tokens* junto a sus frecuencias de aparición.
2) Para cada par de asignaturas i y j, se crea un conjunto B que es la unión de sus *tokens*, y para cada asignatura, se crea un vector $\vec{i}$ y $\vec{j}$ que tendrá una posición por cada *token* en B indicando la frecuencia de aparición del mismo. Finalmente, se aplica la norma l_1 sobre cada vector para obtener las frecuencias relativas.
3) Finalmente, a los 2 vectores de frecuencias se les aplica la similitud del coseno, obteniendo el valor $Cn_{i,j}$ que se integra en la ecuación parametrizable descrita en 3:

$$cos(\theta) = \frac{\vec{i} \cdot \vec{j}}{\|\vec{i}\| \cdot \|\vec{j}\|} = \frac{\sum_{k=1}^{n} i_k j_k}{\sqrt{\sum_{k=1}^{n} i_k^2}\sqrt{\sum_{k=1}^{n} j_k^2}} \quad (4)$$

IV. Optimización del Sistema de Recomendación mediante Algoritmos Genéticos

El SR que aquí se presenta, descrito en la sección III-B, cuenta con múltiples criterios que deben ser ponderados con un peso que indique la relevancia que va a tener cada uno de ellos, así como con la configuración de diferentes métricas de similitud para cada uno de ellos, y del tamaño del vecindario a utilizar. Por ello, se ha decidido diseñar un AG que automáticamente descubra una configuración óptima de todos los parámetros del SR y permita obtener unas recomendaciones lo más satisfactorias posibles para los estudiantes.

A continuación, se especifica las características del AG que se ha diseñado y que sigue un esquema generacional con elitismo.

A. Representación de los individuos

Los individuos de la población se codifican mediante un cromosoma entero compuesto de 14 genes:

- Los dos primeros genes son los pesos que se van a utilizar para combinar las recomendaciones obtenidas por el FC utilizando la información del estudiante y por el FBC utilizando la información de la asignatura. Concretamente, serían los valores α y β definidos en la ecuación 1.
- Los tres siguientes genes se corresponden con los pesos que se asignan a los tres criterios que se han considerado en el FC utilizando la información del estudiante. Concretamente, serían los valores de α, β y γ definidos en la ecuación 2.
- Los cuatro siguientes genes codifican los pesos que son asignados a los cuatro criterios utilizados en el FBC que emplea la información de la asignatura. Concretamente, serían los valores de α, β, γ y δ definidos en la ecuación 3.
- El siguiente gen es el número de vecinos tenidos en cuenta para realizar las estimaciones.
- Los dos siguientes genes son las métricas utilizadas para calcular la similitud según las valoraciones y las

calificaciones respectivamente. Sus valores son tratados como categóricos, y se corresponderán con las métricas contempladas para obtener estas similitudes.
- Los dos últimos genes se corresponden con las métricas utilizadas para calcular la similitud según los profesores y las competencias. Al igual que el grupo anterior, tomarán valores categóricos que se corresponderán con las métricas contempladas para obtener estas similitudes.

B. Operadores genéticos

1) El operador de cruce: es un cruce uniforme que selecciona gen a gen de qué padre toma la información que pasará al hijo. La particularidad de este operador es que el conjunto de genes: (1,2), (3,4,5) y (6,7,8,9) son tratados en bloque, de forma que cada bloque completo es seleccionado del mismo padre para que lo herede uno de los hijos.

2) El operador de mutación: es una mutación uniforme, donde para cada individuo que va a ser mutado, se selecciona cada gen con una cierta probabilidad y se cambia su valor por otro valor aleatorio dentro del rango de valores definidos para ese gen.

3) Optimizador local: este operador, similar a los empleados en los algoritmos meméticos [16], consiste en realizar una pequeña modificación en los genes que representan las medidas de similitud y el número de vecinos de un mismo individuo. Si el cambio realizado hace que el individuo tenga una mejor función de aptitud, se continúa realizando cambios, hasta un máximo de cinco variaciones. Finalmente, si el individuo obtenido mejora al individuo inicial, lo sustituye en la nueva población, en caso contrario, el nuevo individuo obtenido será eliminado. Debido, al coste computacional de este operador, solamente es aplicado al mejor individuo de la población.

C. La función de aptitud

La función de aptitud o *fitness* que se emplea es la raíz del error cuadrático medio (RMSE) entre las valoraciones reales y las estimadas. Para optimizar los tiempos de cómputo, el fitness se obtendrá utilizando el 80% de los datos como valores conocidos y el restante 20% como datos a estimar para obtener el valor de RMSE.

V. Estudio experimental

Esta sección describe la experimentación realizada. El SR se ha desarrollado dentro del *framework* Apache Mahout[2] y el AG utilizando la *librería* JCLEC[3]. La máquina sobre la que se han ejecutado las pruebas es un ordenador personal con SO Ubuntu 16.04 64-bits, un procesador Intel Core i5-3317U y 12 GB de RAM.

[2]https://mahout.apache.org/
[3]http://jclec.sourceforge.net/

A. Configuración del Algoritmo Genético

Los principales parámetros utilizados en el AG, tras un estudio de diferentes configuraciones, son: tamaño de la población: 200, número de generaciones: 100, y probabilidad de mutación y cruce: 0.5 y 0.9 respectivamente.

Con respecto al rango de valores de cada gen: los 9 primeros genes que representan los pesos de cada criterio considerado, se han definido en el rango $[0, 10]$. Este rango podría variarse, ya que estos valores son normalizados antes de ser utilizados en las ecuaciones 1, 2 y 3. El gen 10, que representa el número de vecinos, se ha definido en el rango $[1, 50]$. Los genes 11 y 12 que representan las medidas de similitud de las valoraciones y las calificaciones, se definen en el rango $[0, 4]$ representando las 5 métricas consideradas, la distancia euclidiana, la distancia Manhattan, el coeficiente de correlación de Pearson, el coeficiente de correlación de Spearman y la similitud del coseno. Finalmente, los genes 13 y 14 que representan las medidas de similitud de los profesores y las competencias, se definen en el rango $[0, 1]$ representando las 2 métricas consideradas, el índice Jaccard y la función de verosimilitud logarítmica.

B. Influencia de los diferentes criterios en el Sistema de Recomendación

En esta sección, se evalúa la relevancia de cada uno de los criterios considerados en el SR híbrido presentado, para determinar los factores que pueden resultar más significativos para mejorar las recomendaciones.

En la tabla I se muestra la mejor configuración de pesos encontrada por el AG. En el caso de la configuración del FC que emplea información del estudiante, se puede apreciar que el factor más relevante es la calificación obtenida por los estudiantes (peso de 0.53), seguido por las valoraciones dadas a las asignaturas (peso de 0.25) y la especialidad cursada (peso de 0.22). Estos resultados muestran la relevancia de las calificaciones en la búsqueda

Cuadro I
Mejor configuración del SR obtenida por el AG

SR Híbrido	
Peso del FC	0.58
Peso del FBC	0.42
FC (Basado en información del estudiante)	
Valoraciones (métrica sim.)	Distancia Manhattan
Calificaciones (métrica sim.)	Distancia Manhattan
Valoraciones (peso)	0.25
Calificaciones (peso)	0.53
Especialidad (peso)	0.22
Tamaño vecindario	12
FBC (Basado en información de la asignatura)	
Profesores (métrica sim.)	Verosimilitud logarítmica
Competencias (métrica sim.)	Índice de Jaccard
Profesores (peso)	0.54
Competencias (peso)	0.00
Área conoc. (peso)	0.00
Contenido (peso)	0.46

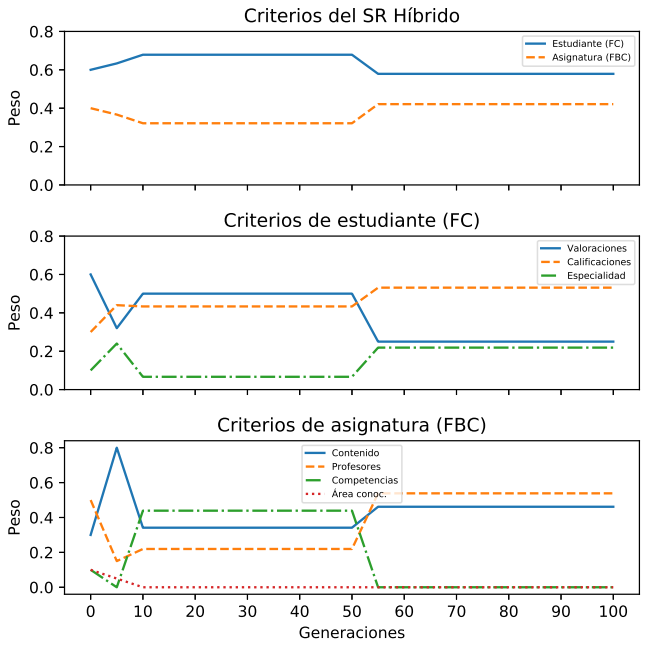


Figura 3. Evolución de los pesos de los criterios en el AG

de estudiantes similares. Así, una asignatura será recomendada a un cierto estudiante si, principalmente, ha sido valorada positivamente por otros estudiantes con unas calificaciones similares a la suya.

Analizando los criterios más relevantes relativos a la información de la asignatura (FBC), se obtiene como factores muy relevantes los profesores que imparten las asignaturas (0.54) y el contenido teórico y práctico de las mismas (0.46); los otros dos criterios (competencias que cubren las asignaturas y el área de conocimiento a la que pertenecen), aparecen como irrelevantes (0.00). Estos resultados, ponen de manifiesto la importancia de los profesores, siendo un factor tan importante como el contenido propio de la asignatura para determinar cómo de similares se perciben dos asignaturas. De esta manera, una asignatura será recomendada a un cierto estudiante si tiene un contenido similar y/o ha sido impartida por el mismo profesorado que otra asignatura que le haya interesado en el pasado.

Finalmente, los pesos asignadas por el AG a las estimaciones realizadas por los sistemas FC y FBC son ponderados con 0.58 y 0.42 respectivamente, mostrando la importancia de considerar tanto información del estudiante, como de la asignatura en la recomendación final.

Para estudiar cómo los pesos han ido evolucionando en las sucesivas generaciones del AG, se analiza la evolución del mejor individuo de la población (Figura 3). Se aprecia que los pesos asignados a las recomendaciones realizadas por el FC y por el FBC, comienzan bastante desequilibradas, dando más importancia al FC. Sin embargo, acaban convergiendo en una importancia similar. En el caso de los pesos asignados a los diferentes criterios considerados en la información del estudiante, se puede apreciar como en las soluciones iniciales el factor de las valoraciones es el más relevante. No obstante, conforme el AG va convergiendo y optimizando el ajuste de parámetros, las calificaciones toman más importancia, pasando valoraciones y especialidad a tener una relevancia similar y menor a las calificaciones. Finalmente, si se estudia la evolución de los pesos asignados por el AG a los diferentes criterios relativos a la información de las asignaturas, se puede apreciar que desde el principio, el área de conocimiento aparece como poco significativa, y es algo que se mantiene hasta el final, mientras que el contenido y los profesores va tomando más relevancia cada vez.

Se puede apreciar que a partir de la generación 60 el peso asignado a los diferentes criterios se mantiene, no siendo mejorados por ninguna otra solución de la población durante el resto de las generaciones. El motivo de que el AG continúe realizando generaciones y optimizando soluciones se debe a que continúa obteniendo mejores soluciones optimizando el resto de parámetros considerados en el AG y configurables en el SR híbrido diseñado, que son las métricas de similitud utilizadas y el tamaño del vecindario.

En la tabla I se muestra las métricas de similaridad y el tamaño de vecindario seleccionadas por el AG. En la figura 4, se muestra la evolución del mejor individuo de la población a lo largo de las generaciones para estos valores. En este caso, es interesante resaltar como estos valores van cambiando, a partir de la generación 60, para optimizar las valoraciones proporcionadas por el SR y en consecuencia reducir el RMSE. Se puede apreciar que existe cierta tendencia a que los criterios de valoraciones y calificaciones utilicen la misma métrica. Además, si esta métrica es la medida de similitud del coseno, se necesita

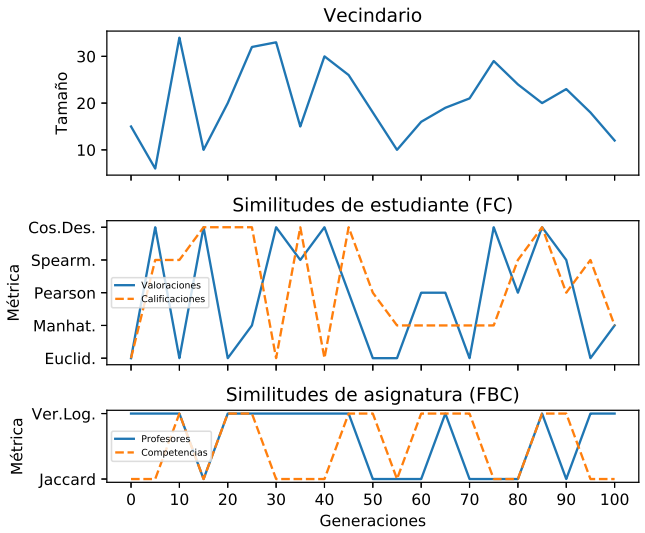


Figura 4. Evolución de las métricas y el vecindario en el AG

un mayor número de vecinos, mientras que para el resto de casos tiende a mantenerse entre 10 y 20.

C. Comparativa con otros modelos

Para concluir el estudio experimental, se compara el rendimiento del SR híbrido frente al uso de los modelos: FC multi-criterio y FBC multi-criterio con la misma configuración de pesos establecida por el AG, pero sin combinar sus resultados. También se incluye en la comparativa el uso de FC mono-criterio empleando solamente el criterio de las valoraciones (el criterio más ampliamente utilizado en los SR) y FBC mono-criterio empleando solamente la información del criterio del profesorado (el más representativo, de acuerdo a los pesos asignados).

Para llevar a cabo este estudio, se ha ejecutado una evaluación del SR utilizando validación cruzada con 10 *fold* con muestreo estratificado. Las métricas estudiadas son el RMSE, relacionado con cuánto difiere una estimación de la valoración real; la ganancia nDCG (*normalized Discounted Cumulative Gain*), relacionada con la capacidad del SR de ofrecer recomendaciones relevantes; el alcance, que determina el porcentaje de usuarios para los que se pueden obtener recomendaciones; y el tiempo de ejecución que necesita el SR para entrenar el modelo y ofrecer una recomendación.

Los resultados finales pueden verse en la tabla II. El enfoque híbrido obtiene mejores resultados en todas las métricas estudiadas. Aunque al usar más información, tarda algo más que los otros modelos en ejecutarse. De este modo, queda demostrada la importancia de utilizar múltiples criterios en el SR, asignándoles además pesos concretos, así como la de realizar una hibridación de diferentes técnicas.

Cuadro II
Comparativa entre modelos de SR

Método	**RMSE**	**nDCG**	**Alcance(%)**	**Tiempo(s)**
SR Híbrido	1.056	0.811	100.00	8.23
FC[1]	1.167	0.806	96.48	7.53
FBC[1]	1.201	0.214	99.36	6.86
FC[2]	1.233	0.798	96.48	5.98
FBC[2]	2.530	0.284	99.36	5.77

[1] multi-criterio, [2] mono-criterio

VI. Conclusiones

En este trabajo se ha desarrollado un SR aplicado a la recomendación de asignaturas híbrido y multi-criterio. Así mismo, se ha implementado un AG que realiza un ajuste de todos los parámetros utilizados en el SR propuesto para determinar las configuraciones óptimas para el SR. Las pruebas realizadas muestran que considerar varios criterios proporciona mejores resultados, pero que la relevancia de cada uno de ellos debe ser estudiada, ya que no todos los factores resultan igual de relevantes. Además, la consideración de un sistema híbrido, que combina tanto filtrado colaborativo, como basado en contenido, también optimiza los resultados alcanzados.

En un futuro se pretende ampliar las pruebas a otras titulaciones, de forma que se pueda comprobar si se obtienen los mismos resultados, y bajo qué circunstancias se pueden generalizar las conclusiones.

VII. Agradecimiento

Este trabajo está financiado por el proyecto de investigación TIN2017-83445-P del Ministerio de Economía y Competitividad y el Fondo Europeo de Desarrollo Regional.

Referencias

[1] C. Romero and S. Ventura. Data mining in education. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 3(1):12–27, 2013.

[2] T.N. Huynh-Lv, N. Huu-Hoa, and T.N. Nguyen. Methods for building course recommendation systems. In *Proceedings of the 8th International Conference on Knowledge and Systems Engineering*, pages 163–168, 2016.

[3] P.-C. Chang, C.-H. Lin, and M.-H. Chen. A hybrid course recommendation system by integrating collaborative filtering and artificial immune systems. *Algorithms*, 9(3):1–18, 2016.

[4] C. Kim, N. Choi, Y. Heo, and J. Sin. On the development of a course recommender system: A hybrid filtering approach. *Entrue Journal of Information Technology*, 14(2):71–82, 2015.

[5] C. Vialardi, J. Chue, J.P. Peche, G. Alvarado, B. Vinatea, J. Estrella, and A. Ortigosa. A data mining approach to guide students through the enrollment process based on academic performance. *User Modeling and User-Adapted Interaction*, 21(1-2):217–248, 2011.

[6] J.W. Han, J.C Jo, H.S. Ji, and H.S. Lim. A collaborative recommender system for learning courses considering the relevance of a learner's learning skills. *Networks Software Tools and Applications*, 19(4):2273–2284, 2016.

[7] S. Spiegel. *A Hybrid Approach to Recommender Systems based on Matrix Factorization*. Thesis, Tech. University Berlin, 2009.

[8] A. Parameswaran, P. Venetis, and H. Garcia-Molina. Recommendation systems with complex constraints: A course recommendation perspective. *ACM Transactions on Information Systems*, 29(4):1–33, 2011.

[9] C.Y. Huang, R.C. Chen, and L.S. Chen. Course-recommender system based on ontology. In *Proceedings of 12th International Conference on Machine Learning and Cybernetics*, pages 1168–1173, 2013.

[10] L. Zhuhadar, O. Nasraoui, R. Wyatt, and E. Romero. Multi-model ontology-based hybrid recommender system in e-learning domain. In *Proceedings of the International Joint Conferences on Web Intelligence and Intelligent Agent Technology*, pages 91–95, 2009.

[11] A.A. Kardan, H. Sadeghi, S.S. Ghidary, and M.R.F. Sani. Prediction of student course selection in online higher education institutes using neural network. 65:1–11, 2013.

[12] J. Sobecki and J.M. Tomczak. Student courses recommendation using ant colony optimization. In *Proceedings of the 2sd Intelligent Information and Database Systems*, volume 5991 of *Lecture Notes in Artificial Intelligence*, pages 124–133, 2010.

[13] E. Khorasani, Z. Zhenge, and J. Champaign. A markov chain collaborative filtering model for course enrollment recommendations. In *Proceedings of the 4th International Conference on Big Data*, pages 3484–3490. IEEE, 2016.

[14] R. Wang. *Sequence-based Approaches to Course Recommender Systems*. Thesis, University of Alberta, 2017.

[15] H. Ma, X. Wang, J. Hou, and Y. Lu. Course recommendation based on semantic similarity analysis. In *Proceedings of 3rd Ieee International Conference on Control Science and Systems Engineering*, pages 638–641, 2017.

[16] E. Özcan and C. Başaran. A case study of memetic algorithms for constraint optimization. *Soft Computing*, 13(8-9):871–882, 2009.

Reconocimiento de genes en secuencias de ADN por medio de imágenes.

Luis A. Santamaría C. *Fac. Cs Computación. BUAP* Puebla, México ALULOC_KAPA@hotmail.com

Sarahí Zuñiga H. *Fac. Cs Computación. BUAP* Puebla, México sarahi.zuhe@gmail.com

Ivo H. Pineda T. *Fac. Cs Computación. BUAP* Puebla, México

María J. Somodevilla *Fac. Cs Computación. BUAP* Puebla, México mariasg@cs.buap.mx

Mario Rossainz L. *Fac. Cs Computación. BUAP* Puebla, México

***Resumen*—En los últimos años, el campo del aprendizaje automático ha progresado enormemente al abordar problemas difíciles de clasificación. El problema planteado en este artículo es reconocer secuencias de ADN, reconocer los límites entre exones e intrones utilizando una representación gráfica de secuencias de ADN y métodos recientes de aprendizaje profundo. El objetivo de este trabajo es clasificar secuencias de ADN utilizando una red neuronal convolucional (Convolutional Neural Network CNN). El conjunto de secuencias de ADN utilizado es la base de datos Molecular (Splice-junction Gene Sequences) Data Set que cuenta con 3190 secuencias, disponible en la página de la UCI, con tres clases de secuencias: límite exón-intrón, límite intrón-exón y ninguna. Se utilizó el conjunto de secuencias de ADN utilizado para el reconocimiento fueron 1847 secuencias de una base de datos con 4 tipos de virus de hepatitis C (tipo 1, 2, 3 y 6) tomada del repositorio disponible en la página de ViPR. Para la utilización de las secuencias de ADN se diseñó un método de representación donde cada base nitrogenada es representada en escala de grises para formar una imagen. Las imágenes generadas se utilizaron para entrenar la red neuronal convolucional. Los resultados muestran que una CNN puede hacer la clasificación de secuencias de ADN con un porcentaje de precisión de entrenamiento del 82 %, una precisión de validación del 75 % y una precisión de evaluación del 80.8 %. Se llega a la conclusión de que es posible clasificar las imágenes de secuencias de ADN de la base de datos empleada.**

***Palabras clave*—Reconocimiento de genes, Aprendizaje profundo, Redes neuronales convolucionales, Codificación de secuencias de ADN .**

I. Introducción

Los métodos de aprendizaje automático permiten identificar características que propician la clasificación, análisis y reconocimiento de patrones. En el área de la biología, el uso de métodos de aprendizaje automático, facilitan el reconocimiento de secuencias de ADN. Este trabajo reconoce los genes de ADN previamente procesados para ser representados por una imagen. Este artículo es divido en secciones, la primera sección es el estado del arte que es el conocimiento previo necesario para el reconocimiento de genes. La segunda sección detalla la metodología que fue utilizada para el análisis de las secuencias de ADN. La tercera y cuarta sección muestran los resultados y las conclusiones obtenidas.

II. Estado del Arte

Los mecanismos o procesos de predicción de genes son aquellos que, dentro del área de la biología computacional, se utilizan para la identificación algorítmica de trozos de secuencias, usualmente ADN genómico [8], y que son biológicamente funcionales. Esto, especialmente incluye los genes codificantes de proteínas y secuencias reguladoras. La identificación degenes es uno de los primeros y más importantes pasos para entender el genoma de una especie una vez ha sido secuenciado [11].

El ácido desoxirribonucleico (ADN) está compuesto por cuatro moléculas llamadas nucleótidos o bases nitrogenadas: adenina, timina, guanina y citosina [9]. Una molécula completa de ADN o, dicho de otro modo, una secuencia de ADN está compuesta por un alfabeto que contiene las letras de las cuatro bases nitrogenadas.

$$\begin{aligned} &\Sigma\{ATGC\} \\ \phi_i = (V_1, V_2, V_3, \ldots, V_n) \\ &V_i \in \Sigma \end{aligned} \tag{1}$$

Donde una cadena ϕ es una secuencia de ADN formada elementos del alfabeto Σ y puede definir las características de un organismo vivo, conteniendo toda la información genética en unidades de herencia llamadas genes. Los mecanismos o procesos de predicción de genes son aquellos que, dentro del área de la biología computacional, se utilizan para la identificación algorítmica de trozos de secuencias, usualmente ADN genómico [1], y que son biológicamente funcionales. Esto, especialmente incluye los genes codificantes de proteínas y secuencias reguladoras. La identificación de genes es uno de los primeros y más importantes pasos para entender el genoma de una especie una vez ha sido secuenciado [2].

Las uniones de empalme son puntos en una secuencia de ADN en la que se elimina ADN ‘’inútil” durante el proceso de creación de proteínas en organismos superiores. El problema planteado en este conjunto de datos es reconocer, dada una secuencia de ADN, los límites entre los exones (las partes de la secuencia de ADN retenidas después del corte y empalme) y los intrones (las partes de la secuencia de ADN que se cortan). Este problema consiste en dos subtareas: reconocimiento de límites de exón / intrón (denominados sitios EI) y reconocimiento de límites de intrón / exón (sitios IE). (En la comunidad biológica, los límites de IE se refieren a los ‘’aceptantes” mientras que los límites de EI se conocen

como ‘’donantes”) [7]. Ambas tareas son complicadas ya que no existe una secuencia estándar para reconocer intrones y exones, razón por la cual es interesante diseñar herramientas que nos ayuden a identificarlos y clasificarlos.

El número de proyectos de investigación sobre genomas actualmente vigentes aumenta a un ritmo acelerado, y proporcionar un catálogo de genes para estos nuevos genomas es un desafío clave. La obtención de un conjunto de genes bien caracterizados, es un requisito básico en los pasos iniciales de cualquier proceso de creación de un genoma. Los métodos de búsqueda de genes computacionales se pueden categorizar libremente como basados en la alineación y en la composición de secuencias o una combinación de ambos. Los métodos basados en la alineación de secuencias se pueden usar cuando se intenta predecir un gen que codifica una proteína para la cual existe un homólogo estrechamente relacionado, este es el enfoque en GeneWise [5] y PROCRUSTES [4].

Los algoritmos basados en composición de secuencias (también conocidos como métodos de búsqueda de genes) contienen un modelo probabilístico de estructura génica basado en señales biológicas (sitios de empalme y sitios de inicio / detención de traducción) y propiedades de composición de secuencias funcionales (exones como secuencias codificantes e intrones como secuencias intermedias entre exones e intrones). A diferencia de los métodos basados en la alineación, estos algoritmos se basan sólo en las propiedades intrínsecas de los genes para construir estructuras genéticas predichas. Genscan [10] y Geneid [3] son los dos ejemplos de este enfoque y pueden encontrar genes conocidos y genes nuevos siempre que los genes se ajusten al modelo probabilístico subyacente. Una cadena de ADN es una molécula caracterizada por cuatro bases nitrogenadas Adenina, Timina, Guanina y Citosina [12]. Para mejorar la representación de una cadena de ADN se utilizan secuencias que pueden ser transformadas a representación con valores numéricos o alfabéticos: A (adenina), T (timina), G (guanina) y C (citosina). [10]. Sin embargo, la representación de grandes cantidades de información como secuencias de ADN no hacen sencillo su análisis matemático, esto crea la necesidad de encontrar nuevas formas de representar la información.

En 1988 Lapedes [6] y su equipo de trabajo entrenaron una red neuronal para reconocer genes en secuencias de ADN, lograron una precisión del 91.2 % en las uniones de corte y empalme de intrón / exón y del 92.8 % en las uniones de empalme de exón / intrón. Lo que dio origen a plantear el uso de redes neuronales convolucionales para resolver este mismo problema de clasificación. Este trabajo consistió en buscar una nueva forma de representar secuencias de ADN para su análisis, como ya se ha hecho referencia, existen actualmente diferentes métodos para reconocer genes, pero estas representaciones complican su análisis. La propuesta que presentamos es generar imágenes a partir secuencias de ADN y someterlas a análisis por técnicas de aprendizaje profundo, en específico a redes neuronales convolucionales; utilizadas para la clasificación de imágenes. Actualmente no se ha encontrado un modelo matemático que resuelva el proceso de clasificación por redes neuronales, pero sus resultados llegan a ser tan altos que superan el 99 % en algunos casos [1].

II-A. Red Neuronal Convolucional (CNN)

En los últimos años, el campo del aprendizaje automático ha progresado enormemente al abordar problemas de clasificación, identificación y reconocimiento de patrones. En particular, se ha encontrado que un tipo de modelo llamado red neuronal convolucional CNN (Convolutional Neural Network) por sus siglas en inglés, que logra un rendimiento razonable en tareas de reconocimiento visual de hardware, igualando o superando el rendimiento humano en algunos dominios [11]. Una CNN es un algoritmo para el aprendizaje automático en el que un modelo aprende a realizar tareas de clasificación directamente a partir de imágenes, videos o sonidos. Las CNNs son especialmente útiles para localizar patrones en imágenes con el objetivo de reconocer objetos, caras y escenas. Aprenden directamente a partir de los datos de imágenes, utilizando patrones para clasificar las imágenes y eliminar la necesidad de una extracción manual de características.

Inception-v3 está diseñado para el desafío de Reconocimiento Visual, ésta es una tarea estándar en visión artificial, donde los modelos intentan clasificar imágenes completas en 1000 clases de ImageNet. TensorFlow es una herramienta para el aprendizaje automático. Si bien contiene una amplia gama de funcionalidades, TensorFlow está diseñado principalmente para modelos de redes neuronales profundas. Los modelos modernos de reconocimiento de imágenes tienen millones de parámetros; entrenarlos desde cero requiere una gran cantidad de datos de entrenamiento etiquetados y una gran cantidad de potencia de cálculo (cientos de horas de GPU o más). El aprendizaje de transferencia es una técnica que ataja mucho de esto tomando una pieza de un modelo que ya ha sido entrenado en una tarea relacionada y reutilizándola en un nuevo modelo, en la figura 1 se muestra un ejemplo de una CNN, los filtros se aplican a cada imagen de entrenamiento con diferentes resoluciones, y la salida de cada imagen convolucionada se usa como entrada para la capa siguiente [2]. Aunque no es igual de preciso en comparación a la capacitación del modelo completo, es sorprendentemente eficaz para muchas aplicaciones, funciona con cantidades moderadas de datos de capacitación (miles, no millones de imágenes etiquetadas) y se puede ejecutar en tan solo treinta minutos en una computadora portátil sin una GPU [11].

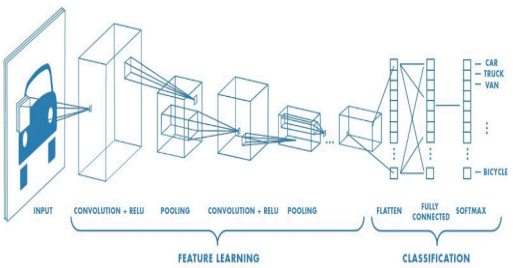


Figura 1. Ejemplo de una Red Neuronal Convolucional

III. Metodología

En esta sección se describe detalladamente como se generaron imágenes a partir de secuencias de ADN y su posterior uso en el entrenamiento de una red neuronal convolucional para clasificación de tres clases de secuencias. En esta etapa del proyecto consistió en convertir las secuencias de ADN a representaciones gráficas para entrenar una CNN. Un aspecto importante que se ha considerado este trabajo es que las CNN son utilizadas para el reconocimiento de patrones y clasificación de imágenes. Las secuencias de ADN de manera general son representadas por letras: A usada para la adenina, G para la guanina, C para la citosina y T para la timina, sin embargo, una CNN no está establecida para procesar información bajo este formato, por esta razón se diseñó una representación gráfica de las secuencias. El primer paso fue asignar un color en escala de grises a cada una de las letras como se muestra en el Cuadro I. Las escalas de grises va de 0 que representa negro, a 1 que represa el blanco, de tal manera que los colores intermedios resultantes son tonalidades de gris para mostrar un mejor contraste. Lo segundo fue hacer

Cuadro I
Representación por color de las bases nitrogenadas.

Base Nitrogenada	Valor de gris
A	0
C	0.3
G	0.7
T	1

que las secuencias pudieran ser representadas por una imagen específica a cada una. Para lograr esto se utilizó una matriz de dimensión 60 X 60, donde el valor 60 coincide con el número de bases nitrogenadas de todas las secuencias de la base de datos. Cada secuencia fue colocada en la primera fila y copiada en el resto de las filas hasta tener 60 en total, así el resultado final es una imagen con barras en la escala de grises como la que se muestra en la Figura 3, cada una de las imágenes obtenidas es específica para cada instancia de la base de datos como se observa en la Figura 2. En total se obtuvieron 3190 imágenes.

```
1  CCAGCTGCATCACAGGAGGCCAGCGAGCAGGTCTGTTCCAAGGGCCTTCGAGCCAGTCTG
2  CCAGCTGCATCACAGGAGGCCAGCGAGCAGGTCTGTTCCAAGGGCCTTCGAGCCAGTCTG
3  CCAGCTGCATCACAGGAGGCCAGCGAGCAGGTCTGTTCCAAGGGCCTTCGAGCCAGTCTG
.
.
.
60 CCAGCTGCATCACAGGAGGCCAGCGAGCAGGTCTGTTCCAAGGGCCTTCGAGCCAGTCTG
```

Figura 2. Secuencia ADN a codificar

III-A. Utilización de CNN a secuencias de ADN

En esta subsección se describe cómo se entrenó una CNN con las imágenes representativas de cada secuencia. Se utilizó

Figura 3. Imagen asociada

una CNN InceptionV3 a la que se le aplicó la transferencia de aprendizaje profundo para categorizar el reconocimiento de tres clases se secuencias de ADN: reconocimiento de límites de exón / intrón (denominados sitios EI), reconocimiento de límites de intrón / exón (sitios IE) y reconocimiento de ninguno de los dos anteriores (N).
Una vez que se logró representar a las secuencias de ADN como imágenes se utilizó una CNN y con la librería de software TensorFlow se construyó un modelo de clasificación basado en una red neuronal convolucional pre-entrenada. Se utilizaron CNNs InceptionV3 a las que se les aplicó la transferencia de aprendizaje profundo para categorizar el reconocimiento de una base de datos con cuatro clases de secuencias de ADN: virus de Hepatitis C tipo 1, 2, 3 y 6 y el reconocimiento de otra base de datos con tres clases de límites de exón / intrón (denominados sitios EI), reconocimiento de límites de intrón / exón (sitios IE) y reconocimiento de ninguno de los dos anteriores (N). Para ajustar el modelo a nuestro problema se entrenaron las últimas capas de las redes con instancias obtenidas de las bases de datos, ambas redes fueron entrenadas en 4000 pasos.

Primero se entrenó la CNN para hacer la clasificación de los 4 tipos de virus de Hepatitis, posteriormente se entrenó una CNN con solamente 2 clases: EI e IE y por último se entrenó otra CNN con todas las clases de la base de datos: EI, IE y N para comparar los resultados de las ultimas dos neuronas.

IV. Resultados

Los resultados de clasificación para la CNN entrenada con la base de datos de los cuatro tipos de virus de Hepatitis C muestran una precisión de evaluación 95 % con 145 imágenes probadas y al terminar el paso (k) 4000 la precisión de entrenamiento fue del 94.5 % y la precisión de validación del 95 % como se observa en la figura 4. El comportamiento decreciente de la entropia durante el entrenamiento, se aprecia en la figura 5 .
Al usar una CNN con las clases EI e IE se obtiene una precisión de evaluación del 80.8 % con 177 imágenes de prueba y al terminar el paso (k) 4000 la precisión de entrenamiento es del 82 % y la precisión de validación del 75 %. En la Figura 6 se muestra como la exactitud de entrenamiento (naranja) y validación (azul) va cambiando en cada paso y en la Figura 7 se muestra como la entropía disminuye con el incremento de los pasos durante el entrenamiento. Por otro lado, los resultados de la segunda CNN donde se utilizaron las tres clases de la base de datos muestran una precisión de evaluación

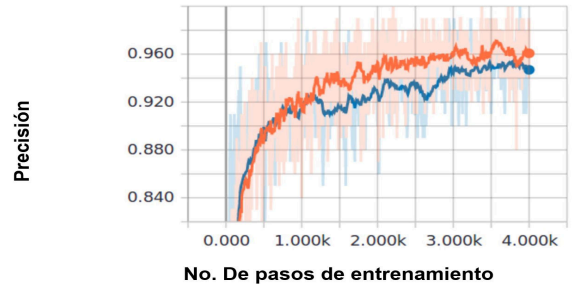


Figura 4. CNN con las clases de virus de Hepatitis C tipo 1, 2, 3 y 6. Naranja: precisión de entrenamiento. Azul: precisión de validación después de 4000 pasos (k).

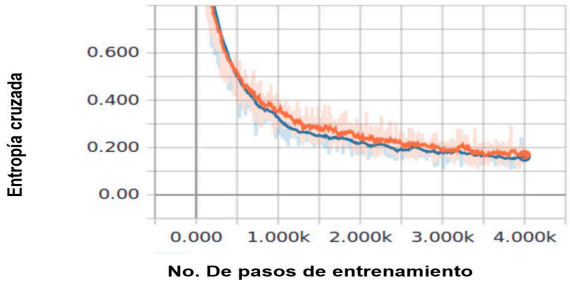


Figura 5. Entropía cruzada de la CNN con las clases de virus de Hepatitis C tipo 1, 2, 3 y 6 después de 4000 pasos (k). Trazo superior: entrenamiento. Trazo inferio: validación.

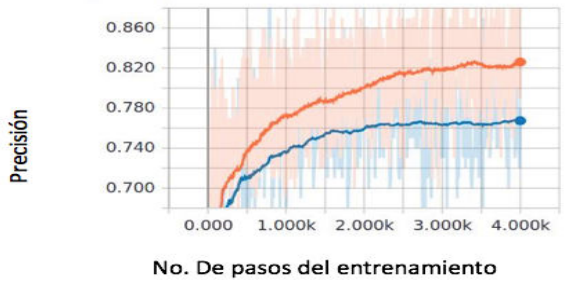


Figura 6. CNN con las clases IE y EI. Naranja: precisión de entrenamiento. Azul: precisión de validación después de 4000 pasos (k).

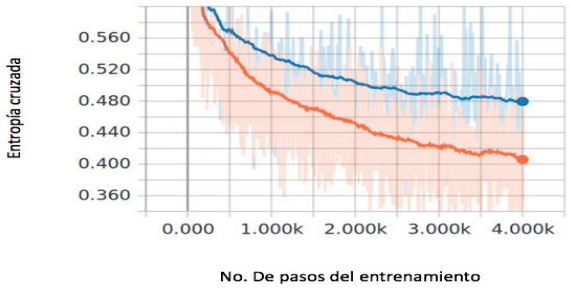


Figura 7. Entropía cruzada de la CNN con las clases IE y EI después de 4000 pasos (k). Naranja: entrenamiento. Azul: validación.

de 57.5 % con 301 imágenes y al terminar el paso 4000 la precisión de entrenamiento 69 % y la precisión de validación con un 56 % como se aprecia en la Figura 8. En la Figura 9 se muestra los cambios de la entropía en cada etapa del entrenamiento.

V. Conclusiones

Los resultados obtenidos de la CNN entrenada con la base de datos de virus de Hepatitis C sugieren que la metodología de aprendizaje automático empleada en este trabajo es adecuada para la clasificación de las imágenes generadas a partir de las secuencias de ADN, mostrando importantes y altos porcentajes de precisión de evaluación, precisión de entrenamiento y la precisión de validación. Estos resultados nos llevaron a realizar los siguientes experimentos para el reconocimento de exones e introns en la siguiente base de datos. Para este caso las CNN muestran que los porcentajes de precisión de validación son menores en comparación a los de una red neuronal tomando como referencia el trabajo de Lapedes [6]. La importancia del trabajo es que se presentan resultados favorables para seguir explorando el uso de las redes neuronales convolucionales utilizando la representación de las secuencias de ADN como imágenes, un método de codificación sencillo y práctico.

En este trabajo se ha logrado realizar clasificación de secuencias de ADN usando una CNN y los resultados demuestran que las CNN son capaces de realizar esta clasificación hasta con un 80.8 % de precisión de evaluación para el experimento con las clases IE e EI y el 57.5 % para el experimento con las clases IE, EI y N. Resultados similares se pueden observar en la precisión de entrenamiento y validación de las Figuras 6 y 8. En el caso de los cuatro tipos de hepatitis se logran resultados de hasta 94.5 % de precisión de evaluación .

La diferencia entre los resultados obtenidos para los experimentos con dos y tres clases se puede justificar que al incrementar el número de clases se incrementa la entropía Figuras 7 y 9. La entropía cruzada es una métrica que puede utilizarse para reflejar la precisión de los pronósticos probabilísticos y está estrechamente vinculada con la estimación por máxima verosimilitud. La entropía cruzada es una función que permite evaluar el resultado de la clasificación en vez de utilizar la métrica del error cuadrático medio, el valor de la entropia cruzada permite evaluar el progreso del proceso de aprendizaje de la información [1].

Por otro lado, se habla de que la transferencia de aprendizaje es buena cuando se disponen de pocas imágenes para entrenar la red y que permite llegar a resultados aceptables en la mayoría de los casos, sin embargo, todavía es posible mejorar aún más la precisión de validación y entrenamiento y disminuir la entropía si se entrena una red neuronal desde cero, es decir se debe contar con una base de datos de millones de instancias y un equipo de cómputo con GPU para entrenar esta red pero seguramente ofrecerá mejores resultados que la CNN preentrenada que utilizamos para este trabajo.

En conclusión, se puede afirmar que una red neuronal convolucional del modelo InceptionV3 es capaz de clasificar secuencias de ADN si la secuencia es procesada y transformada a una imagen, sin embargo, los porcentajes de exactitud se pueden mejorar si se entrena una CNN con una base de

Figura 8. CNN con las clases IE, EI y N. Naranja: precisión de entrenamiento. Azul: precisión de validación después de 4000 pasos (k).

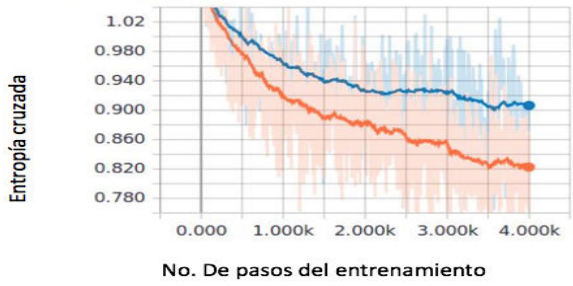


Figura 9. Entropía cruzada de la CNN con las clases IE, EI y N después de 4000 pasos (k). Naranja: entrenamiento. Azul: validación.

secuencias más grande.

Reconocimientos

Los autores agradecen al Consejo Nacional de Ciencia y Tecnología (CONACyT) de México , a la Benemérita Universidad Autónoma de Puebla la cual a través de la Facultad de Ciencias de la Computación han brindado el apoyo necesario para la realización y presentación del presente trabajo.

Referencias

[1] How to retrain an image classifier for new categories. https://www.tensorflow.org/tutorials/image_retraining. Accessed: 2018-05-28.

[2] Mathworks (2018). deep learning. https://la.mathworks.com/solutions/deep-learning/convolutional-neural-network.html. Accessed: 2018-05-27.

[3] Jonas S Almeida and Susana Vinga. Universal sequence map (usm) of arbitrary discrete sequences. *BMC bioinformatics*, 3(1):6, 2002.

[4] Chris Burge and Samuel Karlin. Prediction of complete gene structures in human genomic dna1. *Journal of molecular biology*, 268(1):78–94, 1997.

[5] Mikhail S Gelfand, Andrey A Mironov, and Pavel A Pevzner. Gene recognition via spliced sequence alignment. *Proceedings of the National Academy of Sciences*, 93(17):9061–9066, 1996.

[6] A Lapedes, Christopher Barnes, Christian Burks, R Farber, and K Sirotkin. Application of neural networks and other machine learning algorithms to dna sequence analysis. Technical report, Los Alamos National Lab., NM (USA), 1988.

[7] Michiel O Noordewier, Geoffrey G Towell, and Jude W Shavlik. Training knowledge-based neural networks to recognize genes in dna sequences. In *Advances in neural information processing systems*, pages 530–536, 1991.

[8] Christos A Ouzounis. Rise and demise of bioinformatics? promise and progress. *PLoS computational biology*, 8(4):e1002487, 2012.

[9] Arturo Panduro. *BiologÃa molecular en la clÃnica*. McGraw-Hill Interamericana, 2009.

[10] Genís Parra, Enrique Blanco, and Roderic Guigó. Geneid in drosophila. *Genome research*, 10(4):511–515, 2000.

[11] SL Salzberg, DB Searls, and S Kasif. Computational gene prediction using neural networks and similarity search. *Computational Methods in Molecular Biology*, 32:109, 1998.

[12] Zhu-Jin Zhang. Dv-curve: a novel intuitive tool for visualizing and analyzing dna sequences. *Bioinformatics*, 25(9):1112–1117, 2009.