Nereis. Revista Iberoamericana Interdisciplinar de Métodos, Modelización y Simulación	7	59-66	Universidad Católica de Valencia San Vicente Mártir	Valencia (España)	ISSN 1888-8550

Big-Datasets Manager: una herramienta libre para la manipulación de ficheros de datos con número elevado de instancias y atributos

Fecha de recepción y aceptación: 15 de octubre de 2014, 10 de noviembre de 2014

Elizabeth Martínez-Pérez[1], Juan Alberto Castillo-Garit[2,3*] y Yasser Bruno Ruiz-Blanco[1]

[1] CAMD-BIR Unit, Facultad de Química-Farmacia. Universidad Central "Marta Abreu" de Las Villas.
[2] Applied Chemistry Research Center. Universidad Central "Marta Abreu" de Las Villas.
[*] Correspondencia: Santa Clara, 54830 Villa Clara, Cuba. *E-mail*: juancg@uclv.edu.cu
[3] Institut Universitari de Ciència Molecular, Edifici d'Instituts de Paterna. Universitat de València.

ABSTRACT

A program called Big-Datasets Manager designed to manage large text documents with information structured in instances and attributes is introduced. The program has a friendly graphical interface, is free and multiplatform. Big-Datasets Manager allows editing text datasets for data mining analyses, common tasks are related to selecting and ordering attributes, filtering and replace data, concatenate files, etc. The program is aimed to specialists in medicinal chemistry, economic, social science specialists or other specialists who employ large files of data structured in rows (instances) and columns (attributes).

KEYWORDS: *CSV file, big-data file, data mining, file manager.*

RESUMEN

Se introduce un programa denominado Big-Datasets Manager destinado a la gestión de documentos con grandes volúmenes de información estructurada en instancias y atributos. El programa posee una interfaz gráfica amigable, es libre y multiplataforma. El software permite dar solución a problemáticas usuales en la edición de documentos que se han de emplear en análisis de minería de datos cuya base implica seleccionar y ordenar atributos, filtrar y remplazar datos y concatenar documentos, entre otros. El programa está dirigido a especialistas de la química medicinal, económicos, especialistas de las ciencias sociales y otros especialistas que empleen grandes ficheros de datos estructurados en filas y columnas.

PALABRAS CLAVE: *fichero CSV, archivo con gran volumen de datos, minería de datos, gestor de archivos.*

INTRODUCCIÓN

La gestión de grandes volúmenes de datos en investigaciones relacionadas con bioinformática, quimioinformática, medicina o economía, entre otras, es un problema creciente, debido al aumento continuo de la información publicada en bases de datos relacionadas con las distintas temáticas. El uso eficiente de técnicas de minería de datos es un factor determinante en la inferencia de conocimiento y realización de predicciones a partir de tales volúmenes de datos.

Los principales softwares para minería de datos y reconocimiento de patrones, como Weka (1), SPSS (2) y STATISTICA (3) emplean conjuntos de datos estructurados en instancias (filas) y atributos (columnas) como fuente de información. La edición y cribado de tal estructura de datos se convierte en un problema al emplear volúmenes muy elevados de información que pueden

conformar ficheros compuestos por decenas de miles de atributos y/o cantidades aún mayores de instancias, de tal forma que modificaciones como unir dos o más conjuntos de datos, adicionar instancias o atributos, seleccionar un subconjunto de estas o cambiar el formato del fichero de datos, entre otras, se vuelven operaciones casi irrealizables con editores gráficos de texto como NotePad++ y TextPad en Windows, y gedit en Linux, o con gestores de datos como Excel.

En quimioinformática son varios los programas de cálculo de descriptores moleculares empleados para obtener un vector de atributos para una molécula dada (por ejemplo: DRAGON, TOMOCOMD-CARDD, PADEL, CDK descriptor calculator, ADRIANA CODE, CODESSA-PRO y CERIUS) (4-15). De modo similar, para aplicaciones bioinformáticas relacionadas con proteínas, recientemente en nuestro laboratorio se implementó un programa de nombre ProtDCal capaz de generar del orden de decenas de miles de descriptores por secuencia o estructura de proteínas (16, 17), lo cual se une a los varios miles de descriptores que es posible obtener, para proteínas, con servidores web como PROFEAT (18, 19), PROTEIN RECON (20) y PseAAC (21). En la práctica el número máximo de descriptores posibles por generar, para cada instancia de un conjunto de moléculas orgánicas o proteínas, puede dar lugar a ficheros con elevado requerimiento de memoria, lo que limita su visualización y gestión con editores de texto.

La solución común para tales problemas implica la implementación de algoritmos específicos en lenguajes de programación como PERL, Python y Shell Script, los cuales poseen baja capacidad de reutilización, debido a que se implementan para dar solución a problemas puntuales y con códigos personalizados cuyo uso por otros investigadores requiere el dominio del lenguaje empleado a fin de ajustar el código a otras condiciones de uso.

De modo general, la utilización de cualquier información requiere poseer un sistema de gestión de datos que garantice calidad y eficiencia en las operaciones sobre los datos que se han de gestionar (22). En este sentido se introduce en el presente trabajo una herramienta computacional denominada *Big-Datasets Manager* (*BDM*) destinada a facilitar la edición de grandes y/o múltiples ficheros de texto estructurados en filas y columnas.

IMPLEMENTACIÓN

El programa BDM ofrece un entorno gráfico amigable, es libre, fue implementado en lenguaje Java (multi-plataforma) (23, 24) y permite gestionar eficientemente los procesos de entrada y salida de datos, manteniendo un bajo consumo de memoria (25). Se puede descargar gratis el programa y su manual de usuario (http://sourceforge.net/projects/bigdatamanager/files/Big-Datasets_manager.rar/download).

El software está diseñado como una aplicación de escritorio y procesa documentos de texto atributos donde los separadores de atributos pueden ser los caracteres: coma, punto, punto y coma, tabulación o espacio(s).

Los requerimientos mínimos de hardware y software son: tener instalado un *JRE* (*Java Runtime Environment*) igual o superior a la versión 7 (26, 27) y 512 Mb de memoria RAM.

USOS Y DESCRIPCIÓN

BDM permite realizar tres grupos de operaciones: *i*) adicionar instancias y/o atributos; *ii*) manipular atributos; *iii*) remplazar texto. La aplicación posee una ventana principal (figura 1) que permite acceder a las demás ventanas de la aplicación. En esta primera interfaz se muestran las operaciones que se han de realizar junto con el estado de su procesamiento (en cola, procesando, terminado o terminado con error).

Una vez seleccionados los ficheros que se han de analizar el programa permite escoger entre diferentes opciones de procesamiento (figura 2).

A continuación se describen a modo de ejemplo algunas de las principales funciones de BDM. Para más detalles consultar el manual de usuario.

Figura 1. Ventana principal.

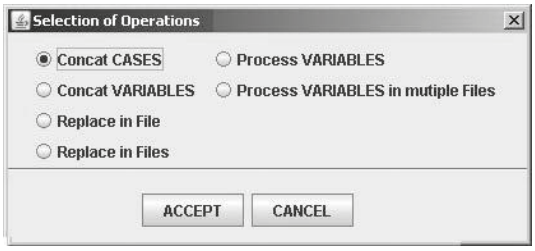


Figura 2. Selección de operaciones.

1. Adicionar instancias y/o atributos

Con esta función se solucionan dos problemáticas básicas donde el objetivo final es unificar en un único fichero de datos toda la información, de instancias y atributos, que se encuentre dispersa en múltiples ficheros: *i*) se posee un conjunto de datos distribuido en varios documentos donde cada uno de ellos contiene los mismos atributos pero diferentes instancias; *ii*) se posee un conjunto de datos distribuido en varios documentos donde cada uno de ellos contiene las mismas instancias pero diferentes atributos. Ejemplos de solución a estos problemas se encuentran en el material suplementario (MS) en la sección 1.

Cuando se desea concatenar casos el usuario debe supervisar que cada fichero que se vaya a utilizar contenga las variables en el mismo orden; esta verificación (y modificación en caso de ser necesario) puede realizarse con el propio BDM según se explica en el epígrafe 2. En el caso de la opción de concatenar variables, la verificación de la coincidencia en el orden de los casos debe ser realizada por el usuario. Para acceder a estas opciones en la ventana de selección de operaciones (figura 2) seleccionamos "CONCAT CASES" (CONCAT VARIABLES), donde se visualizará la ventana correspondiente a la figura 3.

En el momento de agregar documentos es necesario escoger el separador de atributos que utiliza este (figura 4).

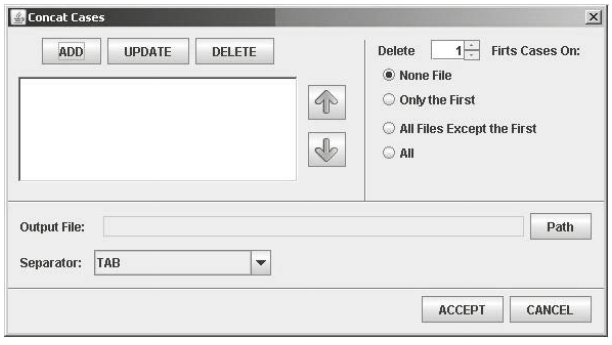


Figura 3. Concatenar documentos.

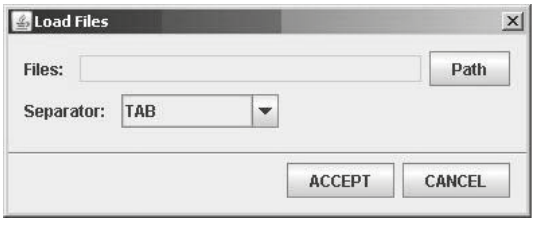


Figura 4. Cargar documentos.

El BDM interpreta cada línea de igual manera, por lo que si el documento en su primera línea tiene los nombres de las variables, para no repetir esta línea o cuando queremos prescindir de las primeras instancias, es posible emplear la opción "DELETE X FIRST CASES (VARIABLES) IN".

2. Manipulación de atributos

La manipulación de atributos permite: cambiar los separadores, seleccionar y/u ordenar variables y seleccionar variables entre múltiples ficheros. A las operaciones que se realizan sobre un solo documento se accede en la opción PROCESS VARIABLES, la ventana de la figura 2, y en PROCESS VARIABLES IN MULTIPLE FILES se procesan con múltiples documentos. En los párrafos siguientes se explicarán algunos problemas que pueden ser resueltos con estas operaciones.

Se poseen diversos ficheros de datos cuyos separadores entre los campos de cada línea no coinciden con los requeridos para su interpretación por un determinado programa de minería de datos, se hace necesario *cambiar el separador* –por ejemplo, el software Weka puede cargar directamente ficheros estructurados en filas y columnas con los campos separados por coma (CSV)–, pero no reconoce los mismos datos si estuviesen separados por otros caracteres; en cuyo caso debe indicársele al cargar cada fichero el separador empleado (ejemplo 3, sección 2 del MS).

Se necesita conformar un nuevo documento con una *selección de los atributos* originales (por ejemplo, se desea eliminar de un documento un conjunto de variables que no son relevantes para el proceso); más detalles sobre este ejemplo pueden ser consultados en el MS (sección 2, ejemplo 2).

Se necesita *modificar el orden en que aparecen las variables* en un fichero (por ejemplo, el Weka reconoce la última como el atributo de clase, por lo que necesitamos mover esta columna para el final). En el ejemplo 1 de la sección 2 del MS se ejemplifica con un caso real cómo se soluciona este problema.

Los tres problemas planteados anteriormente pueden ser resueltos en la ventana de la figura 5. Se puede consultar el manual de usuario para mayor detalle de las operaciones.

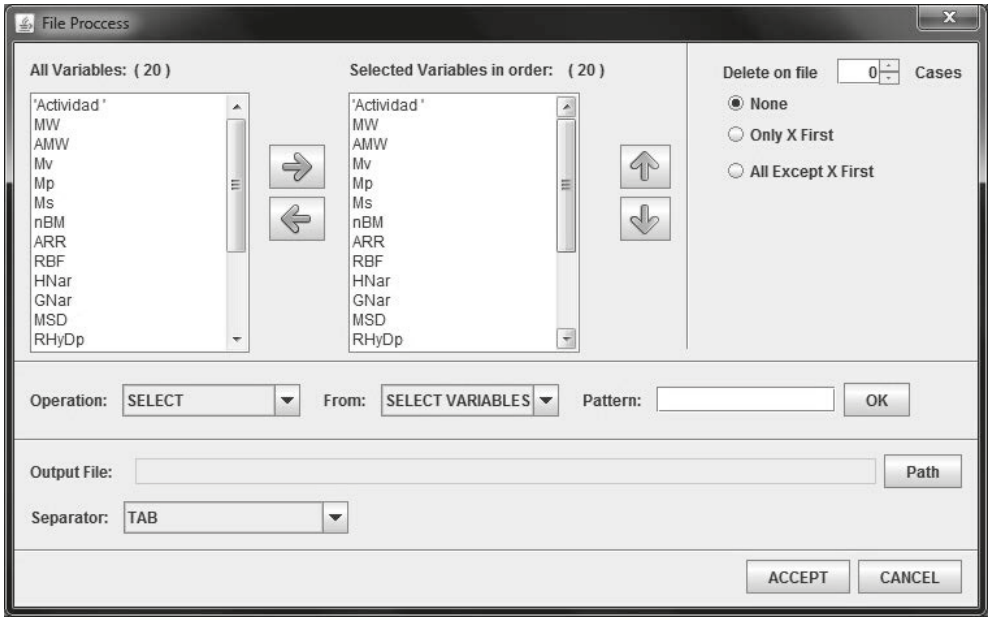


Figura 5. Procesar un documento.

Cuando aplicamos varias técnicas de selección de variables necesitamos obtener un nuevo documento donde se encuentren los atributos de mayor interés. Para ello el programa posee una opción de selección de variables a partir de múltiples documentos (figura 6) respecto a los siguientes criterios:

- Todos los atributos (opción "ALL") (figura 7a, equivalente a la expresión: $D_1 \cup D_2 \cup D_3$ = U en teoría de conjuntos).
- Los atributos comunes en todos los documentos (opción "EQUAL") (figura 7b, equivalente a: $D_1 \cap D_2 \cap D_3$).

- Atributos que aparecen en un único documento (opción "NO EQUAL") (figura 7c, equivalente a la expresión: $U/((D_1 \cap D_2) \cup (D_1 \cap D_3) \cup (D_2 \cap D_3))$.
- Atributos que no están representados en todos los documentos (opción "ALL EXCEPT THE EQUAL") (figura 7d, equivalente a la expresión: $U/(D_1 \cap D_2 \cap D_3)$.
- Atributos que coinciden en al menos equis documentos. (opción "AT LEAST IN X") (figura 7e, equivalente a la expresión: $(D_1 \cap D_2) \cup (D_1 \cap D_3) \cup (D_2 \cap D_3)$.

Ejemplos sobre estas operaciones se encuentran en el MS en la sección 3.

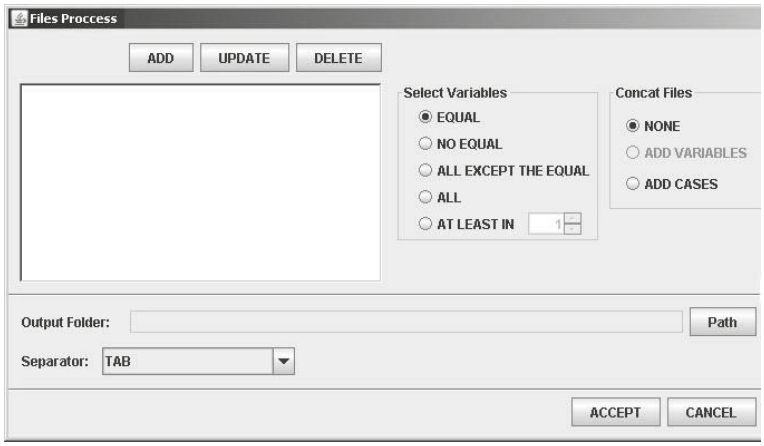
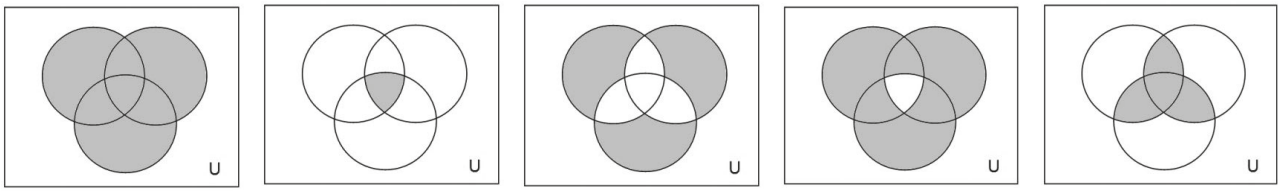


Figura 6. Procesar múltiples documentos.

A continuación se muestra la figura 7, la cual representa a través de diagramas de Venn (28) las diferentes opciones de selección de atributos. Cada circunferencia (D_i) representa los atributos de un documento y las áreas sombreadas corresponden a las diferentes combinaciones de atributos que pueden ser seleccionados.



a b c d e

Figura 7. Ejemplo de procesar 3 documentos: *a*) todos los atributos; *b*) atributos que se repiten en todos los documentos; *c*) atributos que no se repiten entre los documentos; *d*) todos los atributos excepto los que se repiten en todos los documentos; *e*) atributos que se repiten en al menos dos documentos.

3. Remplazo de texto

Esta opción permite reemplazar cualquier carácter o secuencia de caracteres en una serie de documentos (figura 8). En problemas de minería de datos al trabajar con información experimental suelen faltar mediciones para diversas instancias, lo que se conoce como "valores perdidos"; programas como Weka emplean el carácter '?' para interpretar tales "valores", mientras que otros como STATISTICA utilizan el espacio en blanco, lo cual incide en la necesidad de modificar con facilidad tales caracteres en los ficheros de datos aun cuando su tamaño sea elevado. El BDM permite la búsqueda y el reemplazo mediante patrones definidos por una secuencia específica de caracteres (expresión regular) que se comentan en el epígrafe 3.1. La aplicación igualmente permite elegir el reemplazo de todas las apariciones o de un número específico de ellas. En el MS en la sección 4 se encuentran ejemplos útiles de remplazo de texto.

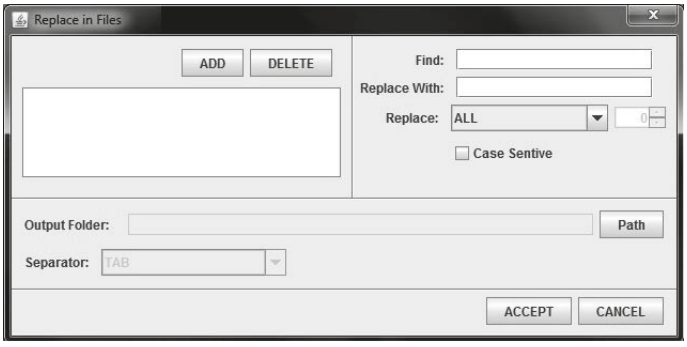


Figura 8. Remplazar texto.

3.1. Expresiones regulares

El BDM empela las expresiones regulares o *regex* (29, 30) correspondientes al lenguaje Java. Estos patrones permiten realizar búsquedas complejas de secuencias de caracteres, incluyendo caracteres especiales y considerando su ubicación en el fichero como parámetro en la búsqueda.

En Java los caracteres más utilizados en la construcción de *regex* son (31, 32):

Tabla 1. Caracteres en las *regex*

Caracteres	Significado
.	Cualquier carácter
\.	Carácter punto
a	Carácter a
[abc]	Cualquiera de los caracteres a, b o c
[^abc]	Cualquier carácter excepto a, b y c
\s	Cualquier carácter de espacio en blanco
\d	Cualquier dígito del 0 al 9. Equivalente a [0-9]
\w	Cualquier carácter alfanumérico. Equivalente a [a-zA-Z0-9]

En el manual de usuario se muestra una mayor cantidad de los caracteres posibles que se pueden utilizar para la construcción de las expresiones regulares. Las expresiones regulares se construyen utilizando caracteres y expresiones lógicas que se muestran en la tabla siguiente (31, 32):

Tabla 2. Expresiones lógicas

Expresión lógica	Significado
ab	El carácter a y seguidamente el b
a|b	El carácter a o el b
(a)	Capturar un grupo. Puedes referirte al grupo capturado *i* con la expresión *i*

Además, en las expresiones regulares existen unos caracteres especiales llamados cuantificadores que detrás de otro carácter especifican la frecuencia con la que el anterior puede ocurrir. En la tabla siguiente se expresan (31, 32):

Tabla 3. Cuantificadores

Uso de cuantificador	Significado
x?	x aparece una o ninguna vez
x*	x aparece cero o más veces
x+	x aparece una o más veces
x{n}	x aparece exactamente n veces
x{n,}	x aparece como mínimo n veces
x{n,m}	x aparece como mínimo n veces y m veces como máximo

CONCLUSIONES

El programa presentado en este trabajo permite realizar disimiles funciones de edición con relevancia en problemas de minería de datos en documentos de texto de gran tamaño que se encuentren estructurados en instancias (filas) y atributos (columnas). Ejemplo de estas son: seleccionar y ordenar atributos, homogenizar documentos, utilizar las expresiones regulares para seleccionar atributos y remplazo de texto. El programa es libre, multiplataforma y ofrece una interfaz web amigable. *Big-Datasets Manager* está dirigido a especialistas de la química medicinal, económicos, especialistas de las ciencias sociales u otros especialistas que empleen grandes ficheros de datos estructurados en filas y columnas.

Anexo material suplementario y programa: el material suplementario asociado con este artículo puede ser encontrado en la versión *on-line*. El programa y su manual de usuario pueden descargarse gratuitamente.

REFERENCIAS BIBLIOGRÁFICAS

1. M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann y I. H. Witten. *The WEKA Data Mining Software: An Update. SIGKDD Explorations*, 2009, 11(1).
2. J. P. Marques-de-Sà. Applied Statistics, using SPSS, STATISTICA, MATLAB and R: Springer, 2007.
3. STATISTICA version. 6.0 Statsoft I. Tulsa, 2004.
4. L. Hall, G. Kellogg y D. Haney. *MOLCONN-Z*. Hall Associates Consulting: Quincy, MA., 1991.
5. G. Cruciani, M. Pastor y W. Guba. VolSurf: a new tool for the pharmacokinetic optimization of lead compounds. *Eur J Pharm Sci*, 2000, 11, Supplement 2(0) S29-S39.
6. A. Mauri, V. Consonni, M. Pavan y R. Todeschini. *DRAGON software: an easy approach to molecular descriptor calculations*. Match. 2006; 56(2) 237-248.
7. Z.R. Li, L. Y. Han, Y. Xue, C. W. Yap, H. Li, L. Jiang *et al*. MODEL-molecular descriptor lab: A web-based server for computing structural and physicochemical features of compounds, *Biotechnology and Bioengineering*, 2007, 97(2) 389-396.
8. H. Hong, Q. Xie, W. Ge, F. Qian, H. Fang, L. Shi *et al*. Mold2, Molecular Descriptors from 2D Structures for Chemoinformatics and Toxicoinformatics, *J Chem Inf Comput Sci*, 2008, 48 1.337-1.344.

9. H. Georg. *BlueDesc-Molecular Descriptor Calculator*. Tübingen, Germany: University of Tübingen, 2008.
10. R. Guha. *The CDK Descriptor Calculator*. 1.3.9 ed. Indiana, 2013.
11. C. W. Yap. PaDEL-Descriptor: An Open Source Software to Calculate Molecular Descriptors and Fingerprints, *J Comput Chem*, 2011, 32 1.466-1.474.
12. J. A. Castillo-Garit, Y. Marrero-Ponce y F. Torrens. Atom-based 3D-chiral quadratic indices. Part 2: Prediction of the corticosteroid-binding globulinbinding affinity of the 31 benchmark steroids data set, *Bioorg Med Chem*, 2006, 14 2.398-2.3408.
13. Y. Marrero-Ponce, J. A. Castillo-Garit, E. Olazabal, H. S. Serrano, A. Morales, N. Castañedo *et al*. Atom, atom-type and total molecular linear indices as a promising approach for bioorganic and medicinal chemistry: theoretical and experimental assessment of a novel method for virtual screening and rational design of new lead anthelmintic, *Bioorg Med Chem*, 2005, 13, 1.005-1.020.
14. Y. Marrero-Ponce, A. Huesca-Guillén y F. Ibarra-Velarde. Quadratic indices of the molecular pseudograph's atom adjacency matrix and their stochastic forms: a novel approach for virtual screening and in silico discovery of new lead paramphistomicide drugs-like compounds, *J Mol Struct(THEOCHEM)*, 2005, 717(1-3) 67-79.
15. Y. Marrero-Ponce, F. Torrens, R. García-Domenech, S. E. Ortega-Broche y V. Romero Zaldivar. Novel 2D TOMOCOMD-CARDD molecular descriptors: atom-based stochastic and non-stochastic bilinear indices and their QSPR applications, *J Math Chem*, 2008, 44 650-673.
16. Y. B. Ruiz-Blanco, Y. Marrero-Ponce, W. Paz, Y. García y J. Salgado. Global Stability of Protein Folding from an Empirical Free Energy Function, *Journal of Theoretical Biology*, 2013, 321 44-53.
17. Y. B. Ruiz-Blanco, Y. Marrero-Ponce, P. J. Prieto, J. Salgado, Y. García y C. M. Sotomayor-Torres. A Hooke's law-based approach to protein folding rate, *Journal of Theoretical Biology*, 2015 1/7/, 364(0), 407-417.
18. H. B. Rao, F. Zhu, G. B. Yang, Z. R. Li y Y. Z. Chen. Update of PROFEAT: a web server for computing structural and physicochemical features of proteins and peptides from amino acid sequence, *Nucleic Acids Research*, 2011, 39 (Web Server): W385-W90.
19. Z. R. Li, H. H. Lin, L. Y. Han, L. Jiang, X. Chen y Y. Z. Chen. PROFEAT: a web server for computing structural and physicochemical features of proteins and peptides from amino acid sequence, *Nucleic Acids Research*, 2006, 34: W32-W7.
20. N. Sukumar y C. M. Breneman. *QTAIM in Drug Discovery and Protein Modeling*. In: C. F. Matta, R. J. Boyd (editors), *The Quantum Theory of Atoms in Molecules: From Solid State to DNA and Drug Design: Wiley-VCH*, 2007 471-498.
21. H. B. Shen y K. C. Chou. PseAAC: a flexible web server for generating various kinds of protein pseudo amino acid composition, *Anal Biochem*, 2008, 373 386-388.
22. G. Ponjuán Dante. Gestión de información en las organizaciones: principios, conceptos y aplicaciones. Santiago de Chile: CECA PI, 1988.
23. J. García de Jalón, J. Ignacio Rodríguez, I. Mingo, A. Imaz, A. Brazález, A. Larzabal *et al*. *Aprenda Java como si estuviera en primero: Escuela Superior de Ingenieros Industriales*. Universidad de Navarra, 1999.
24. L. Lemay y C. L. Perkins. *Teach Yourself Java in 21 Days*. 1st ed: Sams.net; 1996.
25. Procedimientos iniciales con Java: Borland Software Corporation, 2003.
26. Oracle-Java. Disponible en: http://java-oracle.com/downloads/.
27. Download Net. Disponible en: http://download.cnet.com/Java-Runtime-Environment-JRE/3001-2378_4-10009607.html.
28. J. Venn. On the Diagrammatic and Mechanical Representation of Propositions and Reasonings, *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, 1880, 9, 1-18.
29. R. Mitkov. *The Oxford Handbook of Computational Linguistics*. New York: Oxford University Press Inc., 2003.
30. B. Kernighan. *A Regular Expressions Matcher*. Beautiful Code: O'Reilly Media, 2007.
31. J. Zukowski. *Programación Java2*. Anaya Multimedia S.A., 2003.
32. B. Eckel. *Thinking in Java* (2nd ed). Pearson Education, Inc., 2006.