ResearchGate

See discussions, stats, and author profiles for this publication at: https://www.researchgate.net/publication/263974436

Descubriendo reglas de asociación numéricas en series temporales

Conference Paper · November 2009

CITATIONS
0

READS
54

4 authors:

María Martínez Ballesteros
Universidad de Sevilla
31 PUBLICATIONS **381** CITATIONS
SEE PROFILE

Francisco Martínez-Álvarez
Universidad Pablo de Olavide
155 PUBLICATIONS **2,868** CITATIONS
SEE PROFILE

Alicia Troncoso
Universidad Pablo de Olavide
139 PUBLICATIONS **2,457** CITATIONS
SEE PROFILE

José C Riquelme
Universidad de Sevilla
278 PUBLICATIONS **3,915** CITATIONS
SEE PROFILE

Some of the authors of this publication are also working on these related projects:

PERSISTAH - Projetos de Escolas Resilientes aos Sismos no Território do Algarve e de Huelva View project

Earthquake Prediction View project

All content following this page was uploaded by Francisco Martínez-Álvarez on 02 April 2020.

The user has requested enhancement of the downloaded file.

Descubriendo Reglas de Asociación Numéricas entre Series Temporales

M. Martínez-Ballesteros[1], F. Martínez-Álvarez[2], A. Troncoso[2], and J. C. Riquelme[1]

[1]Departamento de Lenguajes y Sistemas Informáticos, Universidad de Sevilla, España
{mariamartinez,riquelme}@us.es
[2]Area de Lenguajes y Sistemas Informáticos, Universidad Pablo de Olavide de Sevilla, España
{fmaralv,ali}@upo.es

Resumen Este trabajo presenta el descubrimiento de reglas de asociación basadas en técnicas evolutivas para obtener relaciones entre series temporales correlacionadas. Para este propósito, se ha propuesto determinar los intervalos que forman las reglas sin discretizar los atributos y permitiendo el solapamiento de las regiones cubiertas por las reglas. Además, el algoritmo ha sido probado en series temporales climatológicas del mundo real tales como temperatura, viento y ozono y los resultados se presentan y comparan con los obtenidos por el muy conocido algoritmo Apriori.

Key words: Series temporales, predicción, reglas de asociación cuantitativas

1. Introducción

La predicción de la evolución temporal de variables –predicción de series temporales– es típicamente llevada a cabo mediante métodos estadísticos. A pesar de su buen funcionamiento e inherente simplicidad presentada por estos métodos en datos sintéticos, cuando se aplican a series temporales del mundo real los resultados no son tan satisfactorios como se esperaba debido a las características no lineales que tales datos manifiestan.

La existencia de otras series temporales correlacionadas con una bajo estudio, es un fenómeno usual. En el campo de las series temporales climatológicas, por ejemplo, es necesario evaluar series temporales como la temperatura, humedad o presión atmosférica para pronosticar si lloverá o no. Por tanto, el problema afrontado en este trabajo consiste en la predicción del comportamiento de una serie temporal mediante la obtención de reglas de asociación entre todas las series temporales correlacionadas existentes. Concretamente, la serie temporal que tiene como objetivo ser pronosticada es el ozono troposférico, el cual es un constituyente clasificado como contaminante cuando excede un cierto umbral. La variación de la concentración de este agente en el aire está bajo continuos análisis, desde que el famoso efecto nocivo que puede causar tanto en los seres humanos como en la naturaleza es conocido [5].

Alicia Troncoso and Marta Arias (Eds.): MINCODA 2009, pp. 16-24, 2009

El objetivo del proceso de extracción de reglas de asociación, consiste básicamente, en descubrir la presencia de conjunciones de pares (atributo(A) – valor (v)), que aparecen en un conjunto de datos con una cierta frecuencia para formular reglas que muestren la existente relación entre los atributos. Formalmente, una regla de asociación es una relación entre atributos en una base de datos en la forma $C_1 \Rightarrow C_2$ donde C_1 y C_2 son conjunciones de pares tales como $A = v$ si $A \in \mathbb{Z}$ o $A \in [v_1, v_2]$ si $A \in \mathbb{R}$. Generalmente, el antecedente C_1 está formado por una conjunción de múltiples pares mientras que el consecuente C_2 es normalmente un único par.

Existen numerosos algoritmos eficientes para encontrar estas reglas. Sin embargo, la mayoría de los investigadores se han centrado en bases de datos con atributos discretos mientras que en el mundo real las bases de datos constan básicamente de atributos continuos, como sucede en el análisis de series temporales. Además, la mayoría de las herramientas que trabajan en dominios continuos, simplemente se limitan a discretizar los atributos usando alguna estrategia concreta y, entonces, tratar estos atributos como si fueran discretos [6]. La principal motivación de esta investigación es desarrollar un algoritmo genético (AG) capaz de encontrar reglas de asociación en bases de datos con atributos continuos evitando la discretización como un paso previo en el proceso.

Una revisión de la reciente literatura publicada revela que el número de trabajos que proporcionan metaheurísticas y algoritmos de búsqueda relativos a reglas de asociación con atributos continuos es escaso. De esta manera, un clasificador fue presentado en [4] con el objetivo de extraer reglas de asociación cuantitativas sobre flujo de datos sin etiquetas. La principal novedad de este enfoque radica en su adaptabilidad a datos que se reciben on-line. Una metaheurística de optimización basada en técnicas de enjambre de partículas fue presentado en [1]. En este caso, la novedad fue la forma de obtener los valores que determinan los intervalos para las reglas de asociación. Además evalúan y prueban varios operadores nuevos en datos sintéticos. Un algoritmo multi-objetivo basado en frente de pareto fue presentado en [2]. La función fitness fue formada por cuatro objetivos diferentes: soporte, confianza, comprensibilidad de la regla (objetivos a maximizar) y la amplitud de los intervalos que forman las reglas (previsto para ser minimizado). El trabajo publicado en [9] presentó un nuevo enfoque basado en tres nuevos algoritmos: clustering valor-intervalo, clustering intervalo-intervalo y clustering matriz-intervalo. La aplicación de ellos fue encontrado especialmente provechoso en el procesamiento de información compleja. Finalmente, otro Algoritmo Genético fue usado en [8] para obtener reglas de asociación numérica. Sin embargo, el único objetivo a ser optimizado en la función fitness fue la confianza. Para satisfacer este propósito, los autores evitan la especificación del actual soporte mínimo, siendo esto la principal contribución de este trabajo.

El resto del trabajo se divide como sigue. La Sección 2 proporciona la metodología usada para obtener reglas de asociación numéricas. Los resultados obtenidos se muestran en la Sección 3. Finalmente, la Sección 4 describe las conclusiones obtenidas.

2. Descripción de la búsqueda de reglas

En un dominio continuo, es necesario agrupar ciertos conjuntos de valores que comparten las mismas características, y como consecuencia, se requiere ser capaz de expresar la pertenencia de los valores en cada grupo. En este trabajo se ha optado por usar no rangos fijos sino intervalos de confianza para representar la pertenencia de tales valores. La búsqueda de los intervalos más apropiados

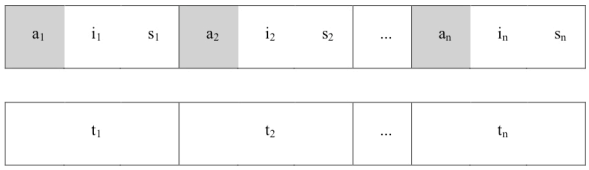

Figura 1. Representación de un individuo de la población.

se lleva a cabo mediante un AG. De este modo, los intervalos se ajustan para encontrar las reglas de asociación con altos valores tanto para el soporte como confianza, junto con otras medidas usadas para cuantificar la calidad de la regla.

En la población, cada individuo constituye una regla. Estas reglas son por tanto, sometidas a un proceso evolutivo en el cual tanto operadores de mutación y cruce son aplicados, y al final del proceso, el individuo que presenta mejor fitness es designado como la mejor regla. Además, la función fitness ha sido provista con un conjunto de parámetros para que el usuario pueda dirigir el proceso de búsqueda dependiendo de las reglas deseadas.

Cada proceso evolutivo proporciona una sola regla. La penalización de los ejemplos cubiertos permite que mediante un Iterative Rule Learning (IRL) [7] las siguientes ejecuciones del Algoritmo Evolutivo generen reglas que intenten cubrir ejemplos no cubiertos por las reglas previas.

Las siguientes secciones detallan el esquema general del algoritmo así como la función fitness, la representación de los individuos y los operadores genéticos.

2.1. Codificación de los individuos

Cada gen de un individuo representa el límite superior e inferior de los intervalos de cada atributo. Los individuos se representan mediante una codificación real ya que los valores de los atributos son continuos. Cada individuo se forma mediante un número de atributos, el cual tiene que ser menor que n, donde n es el número de atributos perteneciendo a la base de datos.

Se dispone de dos estructuras de datos para la representación de un individuo, tal y como se muestra en la Figura 1. Notar que todos los atributos incluidos en la base de datos se representan en la estructura de arriba. Los límites de los intervalos de cada atributo se almacenan en esta estructura, donde i_i es el límite inferior del intervalo y s_i el superior.

Sin embargo, no todos los atributos estarán presentes en las reglas que describen un individuo. Una segunda estructura indicando el tipo de cada atributo, mostrada en la parte inferior de la Figura 1, ha sido desarrollada con el objetivo de mejorar la eficiencia. Notar que t_i puede tener tres valores diferentes: 0 cuando el atributo no pertenece a ningún individuo, 1 cuando el atributo pertenece al antecedente y 2 cuando pertenece al consecuente. Por consiguiente, si se desea recuperar un atributo para una regla específica, puede ser realizado mediante la modificación del valor del tipo igual a 0 por un valor igual a 1 ó 2.

2.2. Generación de la población inicial

El número de atributos se genera aleatoriamente para cada individuo teniendo en cuenta la estructura deseada de las reglas, el número máximo y mínimo de antecedentes y consecuentes permitidos y el número máximo y mínimo de atributos que forman un individuo.

Es importante remarcar que la generación del límite de los intervalos no es arbitrario. Sino que se realiza de manera que al menos un ejemplo del conjunto de datos es cubierto y que el tamaño de los intervalos sea menor a una amplitud máxima dada.

2.3. Operadores genéticos

Los operadores genéticos usados en el algoritmo propuesto son: selección, cruce y mutación.

1. *Selección.* Se usa una estrategia elitista replicando el individuo con el mejor fitness y un método basado en selección por ruleta para los restantes individuos el cual selecciona a los mejores individuos de acuerdo a su fitness.
2. *Cruce.* Dos individuos padres, elegidos mediante selección por ruleta, se combinan para generar un nuevo individuo. Primero, todos los atributos asociados a cada padre se analizan para descubrir su tipo. Entonces, si el mismo atributo en ambos padres pertenece al mismo tipo de atributo, este tipo de atributo sería asignado al descendiente y el intervalo es obtenido generando un número aleatorio entre los límites de los intervalos de ambos padres. Es decir, generaremos un valor aleatorio entre el límite inferior de un padre y el límite inferior del otro padre y de forma análoga para el caso del límite superior. En otro caso, uno de los dos tipos sería elegido aleatoriamente entre ambos padres, sin modificar los intervalos de tal atributo.
3. *Mutación.* Consiste en variar uno de los genes de los individuos. La mutación puede ser enfocada en el tipo de atributo (de antecedente a consecuente, de consecuente a antecedente, de antecedente o consecuente a nulo) o en los intervalos, en el cual son posibles tres casos diferentes: mutación equiprobable del límite superior, del límite inferior o de ambos límites del intervalo. Para este propósito, se genera un valor aleatorio entre 0 y la amplitud máxima y se sumará o restará al límite del intervalo el cual es aleatoriamente seleccionado.

2.4. La función fitness

El fitness de cada individuo permite decidir cuales son los mejores candidatos para permanecer en subsiguientes generaciones. Para poder realizar esta decisión, se desea que el soporte sea alto ya que este hecho implica que más ejemplos de la base de datos sean cubiertos. Sin embargo, tener en cuenta sólo el soporte no es suficiente para calcular el fitness porque el algoritmo podría intentar alargar la amplitud de los intervalos hasta que todo el dominio de cada atributo fuera completado. Por esta razón, es necesario incluir una medida que limite el crecimiento de los intervalos durante el proceso evolutivo. La función fitness elegida para ser maximizada es:

$$f(i) = w_s \cdot sop + w_c \cdot conf - w_r \cdot recub + w_n \cdot nAtrib - w_a \cdot ampl \qquad (1)$$

donde *sop* es el soporte, $conf$ es la confianza, *recub* es el número de instancias cubiertas, *nAttrib* es el número de atributos que aparecen en la regla, *ampl* es la media del tamaño de los intervalos de los atributos que componen la regla y w_s, w_c, w_r, w_n y w_a son pesos para guiar la búsqueda dependiendo de las reglas requeridas.

El soporte recompensa las reglas con un alto valor de soporte, esto es, reglas cumplidas por muchas instancias y el peso w_s puede incrementarse o decrementarse para su efecto. La confianza junto con el soporte son las medidas más usadas para evaluar la calidad de las reglas de asociación. La confianza es el grado de fiabilidad de la regla. Altos valores de w_c pueden ser usados cuando se desean reglas sin error.

El número de instancias recubiertas se usa para indicar que un ejemplo ya ha sido cubierto por una regla previa. Por lo tanto, se prefieren reglas cubriendo diferentes regiones del espacio de búsqueda. La penalización de los ejemplos cubiertos por una regla se hace de la siguiente forma: cada vez que finaliza un proceso evolutivo y seleccionamos el mejor individuo como mejor regla, procesamos la base de datos, viendo qué ejemplos son cubiertos por dicha regla. Cada ejemplo de la base de datos lleva asociado un contador de forma que éste se incrementa cada vez que una regla cubre al mismo.

Las reglas con un alto número de atributos proporciona más información, pero también, en muchos casos, es difícil encontrar reglas en las cuales aparezca un alto número de atributos. El número de atributos de una regla puede ajustarse mediante el peso w_n.

Finalmente, la amplitud controla el tamaño de los intervalos de los atributos que componen las reglas y aquellos individuos con grandes intervalos son penalizados mediante el factor w_a, el cual permite que las reglas sean más o menos permisivas en cuanto a la amplitud de los intervalos.

3. Resultados

El algoritmo propuesto ha sido aplicado para descubrir reglas de asociación entre series temporales de temperatura, viento y ozono desde Junio del 2003 hasta Septiembre del 2003. Notar que para la tarea de predicción, la temperatura y

el viento se fuerzan para que estén en el antecedente y el ozono en el consecuente, obteniendo así una predicción aproximada del ozono en base a estas reglas.

Varios experimentos han sido llevados acabo para validar el comportamiento de los operadores propuestos. Los parámetros del algoritmo son inicialmente establecidos con valores por defecto aunque debería realizarse un análisis más exhaustivo para establecer un conjunto óptimo de valores. Los principales parámetros del AG son los siguientes: 100 para el tamaño de la población, 100 para el número de generaciones, 20 para el número de reglas a obtener y 0.8 para la probabilidad de mutación. Los pesos de la función fitness son: 2 para w_s, 0.5 para w_c, 1 para w_r, 0.2 para w_n y 1.2 para w_a.

La razón para asignar un valor alto en el peso w_s es para cubrir el máximo número de ejemplos por las reglas obtenidas. Sin embargo, el peso asociado a la confianza es más bajo ya que es imposible obtener reglas con una gran confianza debido a la existencia de bastante ruido en la base de datos. El peso asociado a las instancias cubiertas por otras reglas así como la amplitud de los intervalos son moderadamente altos para penalizar las reglas cuyos intervalos son demasiado grandes y están cubriendo ejemplos ya cubiertos mediante otras reglas (recordar que el objetivo es cubrir todo el conjunto de datos). El peso asociado al número de atributos ha sido establecido con un valor pequeño para permitir que las reglas consten de tantos atributos como sea necesario.

La Figura 2 muestra la evolución de la mejor regla y la media de la población durante todo el proceso de evolución para 10 ejecuciones. Se puede observar que el conjunto inicial de reglas mejora su calidad a lo largo de las generaciones.

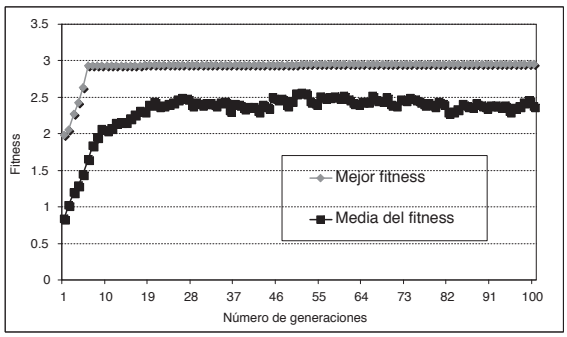

Figura 2. Evolución de la mejor regla y la media de la población.

La Tabla 1 muestra las cinco reglas seleccionadas entre las veinte encontradas por el AG. Puede notarse que cuatro de ellas tienen sólo dos atributos, los cuales son la temperatura (en el antecedente) y el ozono (en el consecuente). Este hecho revela que la temperatura proporciona más información acerca del ozono que el viento. La quinta regla seleccionada tiene dos atributos en el antecedente –la temperatura y el viento– y el ozono en el consecuente. Igualmente notable es la posibilidad de encontrar reglas que tengan solapamiento pero cubriendo todo el dominio del consecuente al cual la mayoría de los ejemplos pertenecen. Por otro

lado, la amplitud de los intervalos es similar para todas las reglas descubiertas mostrando la estabilidad del algoritmo propuesto.

Tabla 1. Descripción de las reglas encontradas por el AG propuesto.

Reglas	Descripción
R1	temperatura $\in$ [28.5,32.2] $\Longrightarrow$ ozono $\in$ [112.7,139.3]
R2	temperatura $\in$ [31.1,34.8] $\Longrightarrow$ ozono $\in$ [119.0,145.8]
R3	temperatura $\in$ [25.3,29.0] $\Longrightarrow$ ozono $\in$ [97.7,124.0]
R4	temperatura $\in$ [22.6,26.3] $\Longrightarrow$ ozono $\in$ [103.0,128.7]
R5	temperatura $\in$ [20.4,23.0] y viento $\in$ [13.0,15.5] $\Longrightarrow$ ozono $\in$ [91.5,115.5]

La Tabla 2 presenta tres medidas para cada regla mostrada en la Tabla 1. La columna *Confianza* indica el porcentaje de ejemplos cubiertos por la regla entre aquellos ejemplos cubiertos sólo por el antecedente. La segunda columna, *Cubiertos*, muestra el número de ejemplos cubiertos por cada regla el cual está directamente relacionado con el soporte. La columna *Amplitud* presenta la amplitud media de los intervalos para cada regla. Como puede observarse, la confianza en la mayoría de los casos, a pesar del pequeño peso asociado, es mayor del 50 % (y en algunos casos mayor del 70 %), lo cual significa que el error alcanzado por las reglas se puede considerar satisfactorio. El número de ejemplos cubiertos es mucho mayor en reglas con dos atributos (más de 100 ejemplos cubiertos en la mayor parte de los casos) que en aquellas con tres atributos. Además, la media de la amplitud de los intervalos es aproximadamente 15, el cual es un buen resultado teniendo en cuenta la dificultad que conlleva la predicción del ozono.

Tabla 2. Medidas para las reglas obtenidas usando el AG.

Reglas	Confianza (%)	Cubiertos	Amplitud
R1	50.8	159	15.1
R2	47.8	143	15.2
R3	54.8	135	15
R4	56.0	84	14.7
R5	72.7	8	9.7

El algoritmo Apriori[3] implementado en WEKA se ha aplicado para obtener reglas de asociación con el propósito de establecer una comparación entre los resultados del algoritmo propuesto y los del algoritmo Apriori. Antes de aplicar el algoritmo Apriori, la temperatura, viento y ozono del conjunto de datos han sido discretizados ya que este algoritmo sólo maneja atributos categóricos. Las reglas obtenidas mediante este algoritmo se muestran en la Tabla 3. Notar que todas las reglas generadas constan sólo de dos atributos. Puede observarse que hay reglas diferentes con el mismo intervalo de predicción para el ozono, por ejemplo, $R1$, R_3 y R_5, y R_2 y R_4. Finalmente, cabe destacar que estas reglas no cubren el intervalo del 90 al 100 en el cual el conjunto de datos tiene muchas instancias, mientras que el algoritmo propuesto sí lo hace.

Tabla 3. Descripción de las reglas encontradas por el algoritmo Apriori.

Reglas	Descripción
R1	temperatura $\in$ [24.49,27.42] $\Longrightarrow$ ozono $\in$ [100.76,121.54]
R2	temperatura $\in$ [30.35,33.28] $\Longrightarrow$ ozono $\in$ [121.54,142.32]
R3	temperatura $\in$ [27.42,30.35] $\Longrightarrow$ ozono $\in$ [100.76,121.54]
R4	viento $\in$ [11.36,14.2] $\Longrightarrow$ ozono $\in$ [121.54,142.32]
R5	viento $\in$ [11.36,14.2] $\Longrightarrow$ ozono $\in$ [100.76,121.54]

La Tabla 4 es equivalente a la Tabla 2 pero aplicando el algoritmo Apriori. Con respecto a la confianza, ninguna regla tiene valores mayores al 50 % lo cual implica que estas reglas proporcionan un error en la predicción mayor que el algoritmo propuesto en la mayoría de los casos. El número de instancias cubiertas por las reglas proporcionadas mediante el enfoque propuesto es mayor que en el algoritmo Apriori, obteniendo reglas con mejor soporte. Con referencia a la amplitud de la media de los intervalos, ambos algoritmos tienen similar comportamiento. Finalmente, no se ha encontrado ninguna regla con tres atributos usando el algoritmo Apriori.

Tabla 4. Medidas para las reglas obtenidas usando el algoritmo Apriori.

Reglas	Confianza (%)	Cubiertos	Amplitud
R1	42	71	11.8
R2	41	93	11.8
R3	39	88	11.8
R4	29	59	11.8
R5	27	55	11.8

4. Conclusiones

Se ha propuesto un nuevo Algoritmo Genético en este trabajo para descubrir reglas de asociación entre series temporales correlacionadas del mundo real. Este algoritmo ha determinado los intervalos que forman las reglas sin discretización de los atributos y permitiendo superposición de las regiones cubiertas por las reglas. Cuando se predice la serie temporal del ozono con este nuevo enfoque, se obtiene que el error es menor que el error proporcionado por el conocido algoritmo Apriori, ya que la confianza de las reglas generadas por el Algoritmo Genético propuesto es mayor que la confianza de las reglas obtenidas por el algoritmo Apriori.

Agradecimientos.

Los autores quieren agradecer la financiación recibida por parte del Ministerio de Ciencia y Tecnología, proyecto TIN2007-68084-C-00, así como por parte de la Junta de Andalucía, proyecto P07-TIC-02611.

Referencias

1. B. Alatas and E. Akin. Rough particle swarm optimization and its applications in data mining. *Soft Computing*, 12(12):1205–1218, 2008.
2. B. Alatas, E. Akin, and A. Karci. MODENAR: Multi-objective differential evolution algorithm for mining numeric association rules. *Applied Soft Computing*, 8(1):646–656, 2008.

3. S. Kotsiantis and D. Kanellopoulos. Association rules mining: A recent overview. *GESTS International Transactions on Computer Science and Engineering*, 32(1):71–82, 2006.
4. A. Orriols-Puig, J. Casillas, and E. Bernadó-Mansilla. First approach toward on-line evolution of association rules with learning classifier systems. In *Proceedings of the 2008 GECCO Genetic and Evolutionary Computation Conference*, pages 2031–2038, 2008.
5. Sujit K. Sahua, Stan Yipc, and David M. Hollandb. Improved space time forecasting of next day ozone concentrations in the eastern us. *Atmospheric Environment*, 43(3):494–501, 2009.
6. M. Vannucci and V. Colla. Meaningful discretization of continuous features for association rules mining by means of a som. In *Proceedings of the European Symposium on Artificial Neural Networks*, pages 489–494, 2004.
7. G. Venturini. SIA: a Supervised Inductive Algorithm with genetic search for learning attribute based concepts. In *Proceedings of the European Conference on Machine Learning*, pages 280–296, 1993.
8. X. Yan, C. Zhang, and S. Zhang. Genetic algorithm-based strategy for identifying association rules without specifying actual minimum support. *Expert Systems with Applications: An International Journal*, 36(2):3066–3076, 2009.
9. Y. Yin, Z. Zhong, and Y. Wang. Mining quantitative association rules by interval clustering. *Journal of Computational Information Systems*, 4(2):609–616, 2008.

View publication stats