Minería de Reglas de Asociación usando Co-Evolución

Juan Pablo Salamanca Ramírez

Departamento de Ingenieria de Sistemas e Industrial, Universidad Nacional de Colombia, Bogotá, Carrera 45 No 26-85, Colombia,
jpsalamarcara@unal.edu.co

Abstract. Since everyday humanity generates enormous amounts of information, we need to find techniques to process and extract useful knowledge, if there is one there. The task of finding association rules automatically on large data sets, is one of many problems that academia and industry are facing today. Herein, an alternative way to perform association rule mining is studied by applying evolutionary computing technique SPEA2

Keywords: evolutionary computation, evolutionary algorithm, real optimization, large scale

1 Introducción

Ya que día a día la humanidad genera cantidades enormes de información, se hace necesario encontrar técnicas para procesarla y extraer conocimiento útil, si es que allí lo hay. La tarea de hallar reglas de asociación automáticamente en grandes conjuntos de datos, es uno de los tantos problemas a los que se enfrentan la academia y la industria hoy en día. En este documento, se estudia una manera alternativa de realizar minería de reglas de asociación, mediante la aplicación de la técnica de computación evolutiva SPEA2 [4].

2 Estado del Arte

La optimización multi-objetivo mediante el uso co-evolución, toma como base en el concepto de dominancia entre individuos de una misma población y divide el problema general en pequeños problemas manejables [4]. Para el problema de hallar reglas de asociación en un conjunto de datos, se han propuesto diferentes estrategias usando operaciones sobre gráfos. A partir de esto, se pueden analizar dos familias de algoritmos para hallar reglas de asociación, los que utilizan una estrategia de recorrer el gráfo en profundidad y los que recorren el gráfo en amplitud [6]. Entre los del primer grupo, encontramos al algoritmo A Priori [2] y sus diferentes variantes, este se diferencia del algoritmo Partition [5], en que en vez de armar intersecciones del conjunto de datos, contabiliza para calcular el soporte y la confianza de sus reglas. En el segundo grupo se encuentra el

algoritmo FP-Growth, cuyo soporte se basa en conteo, y el algoritmo Eclat, que al igual que Partition, usa intersecciones para calcular el soporte de sus reglas de asociación. Finalmente, desde la computación evolutiva, se ha propuesto tratar el problema de hallar reglas de asociación como un problema de optimización multi-objetivo [7], en donde la dominancia esta dada por las reglas de asociación que tienen mayor confianza, comprensibilidad y correlación de sus componentes. Una de las mejores técnicas optimización que usan co-evolución, es el algoritmo SPEA2[4].

3 Propuesta

Teniendo en cuenta las principales conclusiones de J. Gómez [1] en relación a la preservación de la diversidad como estrategia evolutiva y al análisis de los diferentes esquemas de selección realizado por T. Blicke & L. Thiele[3]; se plantea encontrar un método para seleccionar individuos de una población, en el cual se favorezca la exploración y explotación como principios básicos para disminuir la perdida de diversidad y la probabilidad de estancamiento en un máximo (o mínimo) local. El resultado es un híbrido entre la selección por ruleta y la selección por ranking.

3.1 Estrategia de Selección: Ruleta-Ranking

El procedimiento de selección propuesto, llamado inicialmente Ruleta-Ranking, se describe paso a paso a continuación:

1. Se normalizan las medidas de aptitud de cada individuo de la población usando escala decimal.
2. Se normalizan los datos del paso anterior en el intervalo $[0, 1]$
3. Se ordenan de forma descendente los individuos de acuerdo su medida de aptitud normalizada, de manera que el mejor individuo sea el primero de la lista y el peor el último.
4. Generar un número aleatorio usando una distribución normal con $\mu = 0$ y $\sigma = \frac{1}{3}$, y aplicarle la función valor absoluto.
5. Seleccionar el individuo con la medida de aptitud normalizada inmediatamente mayor al número aleatorio generado en el paso anterior.
6. Retirar al individuo seleccionado de la lista y ajustar las probabilidades de los individuos restantes.

En esta estrategia, los mejores individuos tienen mayor probabilidad de ser seleccionados inicialmente, sin embargo, con su salida de la lista de espera, se redistribuye la probabilidad en los individuos restantes, ofreciéndole prioridad a los nuevos mejores. Esta técnica permite exploración ya que por la forma de la distribución normal, existe la probabilidad de seleccionar individuos no tan buenos. Y permite explotación, porque se seleccionan los mejores en mayor proporción. Para controlar la complejidad del algoritmo, se mantiene la sumatoria total de las medidas de aptitud normalizadas de los individuos en espera y cada

vez que alguno es elegido, su medida de aptitud es restada del total y se aplica otra normalización respecto a este nuevo total solamente cuando se lanza nuevamente la ruleta.

Comentario. Al cambiar la forma de la distribución del número generado en el paso 4, por una con $\mu = 1$, se obtiene que la ejecución del algoritmo genético minimiza la función de aptitud.

3.2 Estrategia de Generación y Reemplazo

Para la generación de nuevos individuos se propone como estrategia el matrimonio, generando de a dos hijos por pareja. Sin embargo, el criterio de selección de pareja se dividió en dos: por similitud y por mejor medida de aptitud, teniendo en cuenta que el mejor individuo tiene $n-1$ oportunidades de elegir, el segundo mejor individuo tiene $n-2$ y así sucesivamente hasta que el penúltimo individuo solo pudiese elegir al peor (n es el tamaño total de los individuos elegibles).

Como estrategia de reemplazo se propuso el reemplazo de estado estable o generacional, en combinación con alguna de las estrategias de generación anteriormente descritas.

3.3 Función de Aptitud

La función de aptitud esta determinada por la noción de fortaleza propia del algoritmo SPEA2 [4]. En la cual se asume la aptitud de un individuo, como la sumatoria de las fortalezas de los individuos que lo dominan. A partir de esto, se entiende que un individuo es mejor entre menos individuos lo dominen.

Funciones Objetivo Se tomaron las funciones objetivo propuestas por J. Hu [7]: la función de comprensibilidad, la función de correlación y la confianza. Adicionalmente, se decidió añadir el soporte para cada uno de los individuos de la población.

K-Medoids para SPEA2 Una de las particularidades de SPEA2, es que preserva un archivo de los mejores individuos hallados hasta el momento t, pero este archivo tiene un tamaño fijo, entonces cuando hacen falta individuos se añaden los mejores de la población en el instante t que no estén presentes en el archivo, hasta completar el número de individuos requerido. Sin embargo, cuando hay una cantidad mayor de individuos que la permitida, el problema se vuelve un poco delicado, no se puede decidir que individuos son mejores, entonces, se usa un algoritmo de agrupamiento para extraer los individuos más representativos del archivo, de manera que no se vaya a perder diversidad. Para tratar esta situación se propone usar el Simple-K-Medoids propuesto por Park [8].

3.4 Inicialización

La inicialización se hace tomando como entrada un conjunto de datos categóricos, a partir del cual, por cada uno de sus registros transaccionales, se hace un proceso con las variables presentes, este proceso consiste en mapear los datos categóricos a un código numérico, de manera que por cada variable exista un entero único que la identifica, el proceso también se encarga de llevar su frecuencia de aparición, para así tener un conjunto de probabilidades que consultar a la hora de generar los candidatos iniciales a regla de asociación.

Formato de las Reglas de Asociación Las reglas de asociación generalmente son representadas por un antecedente X y un consecuente Y. En el caso de la minería de datos, X y Y, hacen referencia a dos subconjuntos de items (o variables) presentes en el conjunto de datos, los cuales manifiestan una relación de causalidad (X implica Y). Para la representación de los individuos en un espacio de búsqueda, se adoptó la representación hecha en el trabajo de J. Hu [7], en donde cada individuo es un arreglo de números y cada número del arreglo representa la presencia o no presencia de la variable categórica j-ésima del conjunto de datos, ya sea en el antecedente o en el consecuente.

3.5 Operadores

En cuanto a los operadores se tienen tan solo dos, mutación y cruce.

Mutación Para la mutación simplemente se toma un individuo, y de acuerdo a una probabilidad fijada como parámetro, se cambian valores del arreglo de números, respetando el dominio de: presencia en antecedente, presencia en consecuente y no presencia.

Cruce El operador de cruce también es sencillo, en este simplemente se genera un pivote aleatorio sobre los arreglos de los progenitores, y a partir de allí se crea el genotipo de los nuevos individuos.

4 Experimentos y Resultados

El algoritmo es ejecutado usando como datos de entrada el conjunto Mushroom [9]. Finalmente al usar una estrategia de reemplazo generacional, se encuentra muy poca variedad de reglas, a diferencia de cuando se usa reemplazo de estado estable, en el cuál la variedad y cantidad es mucho mayor. Por otra parte, en varias ejecuciones, con un número de iteraciones al rededor de 10, se encuentran reglas con soportes muy buenos (superiores al 1 por ciento) y niveles de confianza del 100 por ciento, sin embargo, la función de comprensibilidad favorece la presencia de reglas con antecedentes vacíos.

5 Trabajo Futuro

Se espera que a partir de este estudio, se puedan adelantar en un futuro, investigaciones sobre operadores de cruce y mutación, esquemas de adaptabilidad y estrategias evolutivas más elaboradas, con el objetivo de realizar nuevos aportes a una técnica tan eficaz como SPEA2.

References

1. J. Gómez: Self Adaptation of Operator Rates in Evolutionary Algorithms Universidad Nacional de Colombia and The University of Memphis, (2004)
2. Agrawal, R., Mannila, H., Srikant, R., Toivonen, H., & Verkamo, A. I. (1996). Fast Discovery of Association Rules. Advances in knowledge discovery and data mining, 12(1), 307-328.
3. Tobias Blickle & Lothar Thiele: A Comparison of Selection Schemes used in Genetic Algorithms. Computer Engineering and Communication Networks Lab. Swiss Federal Institute of Technology. (1995)
4. Eckart Zitzler, Marco Laumanns, and Lothar Thiele, SPEA2: Improving the Strength Pareto Evolutionary Algorithm Computer Engineering and Networks Laboratory (TIK), Department of Electrical Engineering, Swiss Federal Institute of Technology (ETH) Zurich (2001)
5. Ashok Savasere, Edward Omiecinski & Shamkant Navathe, An efficient algorithm for Mining Association Rules in Large Databases, College of Computing, Giorgia Institute of Technology
6. Ma, B. L. W. H. Y. (1998, August). Integrating classification and association rule mining. In Proceedings of the fourth international conference on knowledge discovery and data mining.
7. Hu, J., & Yang-Li, X. (2007, December). Association rules mining using multiobjective coevolutionary algorithm. In Computational Intelligence and Security Workshops, 2007. CISW 2007. International Conference on (pp. 405-408). IEEE.
8. Park, H. S., & Jun, C. H. (2009). A simple and fast algorithm for K-medoids clustering. Expert Systems with Applications, 36(2), 3336-3341.
9. Blake, C. L., and Merz, C. J. 1998. UCI Repository of Machine Learning Databases. http://www.ics.uci.edu/mlearn/MLRepository.html.