ARTÍCULO DE INVESTIGACIÓN

Aprendizaje de clases de equivalencia de redes bayesianas basado en búsqueda competitiva de hormigas artificiales

Learning of bayesian networks equivalence classes based on competitive search of artificial ants

E. Fabián Cardozo[*], *Henry Arguello Fuentes*[**]

Resumen

Este artículo propone un algoritmo de aprendizaje de clases de equivalencia de redes bayesianas basado en un algoritmo de búsqueda Greedy y modelos de búsqueda inspirados en hormigas competitivas. Específicamente para el algoritmo propuesto, se obtuvo una mejor aproximación entre la red predicha y la red bayesiana teórica de ejemplo ASIA, con respecto a algoritmos anteriores, para conjuntos de datos con 20 y 500 muestras. En promedio el algoritmo desarrollado obtuvo una aproximación con respecto a la distancia estructural de hamming de 10.7% y 5.3% menor comparada con la obtenida por los algoritmos Greedy y de colonia de hormigas (ACO-E) respectivamente para 20 muestras, y de hasta el 6.8% menor con respecto al algoritmo ACO-E para 500 muestras. Además, para 500 muestras el número de llamadas a la función de puntaje realizadas por el algoritmo propuesto fue menor que las realizadas por el algoritmo ACO-E en el 90% de las combinaciones, concluyendo que hubo una reducción de la complejidad computacional. Finalmente se presentan los resultados de la aplicación del algoritmo propuesto a un microarreglo obtenido por muestras de pacientes con Leucemia Mieloide Aguda (LMA) con 6 nuevas interacciones con dependencias estadísticas como potenciales interacciones biológicas con alta probabilidad.

Palabras clave: redes bayesianas, aprendizaje estructural, clases de equivalencia, colonia de hormigas, búsqueda heurística.

Abstract

This article proposes an algorithm for learning equivalence classes of Bayesian networks based on a Greed search algorithm and search patterns inspired by competitive ants. Specifically, for the proposed algorithm, we obtained a better approximation between the predicted network and the theoretical network ASIA with respect to previous algorithms for data sets with 20 and 500 samples. On average, the algorithm developed an approximation with respect to Structural Hamming Distance of 10.7% and 5.3% lower than Greedy algorithms and ACO-E respectively to 20 samples, and up to 6.8% lower tan ACO-E algorithm for 500 samples. Furthermore, for 500 samples the number of calls to the scoring function performed by the algorithm proposed was smaller than in the ACO-E algorithm in 90% of the combinations, concluding that there was a reduction in the computational complexity. Finally, we present the results of applying the proposed algorithm to a microarray samples obtained from patients with acute myeloid leukemia (AML) with 6 new interactions with statistical dependencies as potential biological interactions with high probability.

Key words: Bayesian networks, learning sructural equivalence classes, ant colony, heuristic search.

Recibido: mayo 15 de 2013 **Aprobado:** octubre 1 de 2014

Introducción

El surgimiento de tecnologías para obtener muestras de elementos de una célula a nivel molecular, ha traído consigo interés en inferir conocimiento estructural acerca de sus interacciones (Pe'er, 2005). El enfoque de biología de sistemas supone que las muestras están relacionadas con una estructura, donde los elementos muestreados no están aislados sino que interactúan para formar una red con propiedades y funciones propias (Pe'er, 2005). Dentro de diferentes modelos matemáticos que interpretan de manera estructural

* MSc. Ingeniería de Sistemas e Informática, Universidad Industrial de Santander, Bucaramanga - Colombia. e-mail: fabiancardozo@gmail.com

** PhD, Profesor Asociado, Universidad Industrial de Santander. e-mail: henarfu@uis.edu.co

un fenómeno, las redes bayesianas han sido utilizadas para representar el sistema de interacciones moleculares. En estas aproximaciones se asume que el nivel de expresión de los genes que conforman la red y sus relaciones tiene un grado de incertidumbre, por lo tanto pueden ser representados como variables aleatorias y probabilidades condicionales entre ellas. Dada esta incertidumbre, un estado propio de la red se puede describir por una distribución de probabilidad conjunta de las variables (Jensen, 2007). Así, el análisis e interpretación de las muestras se define como el problema de encontrar una red bayesiana que mejor se ajuste a ellas. Conocida una posible red, es posible hacer aproximaciones de diferentes escenarios de las moléculas (Needham *et al*., 2009). El proceso de encontrar una red bayesiana consiste en obtener los parámetros de dicha red (Heckerman, 1995), y su estructura. Esto último, sin embargo, es un problema NP-Duro (Chickering, 1996) y se formaliza de la siguiente manera: dado E_B, el conjunto de todas las estructuras posibles de redes bayesianas con n nodos, Sf(G):EB→ ℝ la función de puntaje que mide cada estructura $\{G\}$ en E_B; y M un algoritmo de búsqueda, el objetivo del aprendizaje de redes bayesianas consiste en encontrar la mejor estructura G^* de una red bayesiana tal que:

$$G^* = \max_{G_i \in E_B} S_f(G) \qquad (1)$$

Este enfoque es llamado puntaje búsqueda, donde los algoritmos diseñados tienen en común operadores para moverse dentro del espacio de búsqueda, y su diferencia radica en como se utilizan dichos operadores. Entre ellos se han presentado en la literatura, el algoritmo K2 (Cooper, 1992), algoritmos evolutivos (Wong, 2004), colonias de hormigas (de Campos, 2002), enjambre de partículas (Du *et al.,* 2009), sistemas inmunes artificiales (Castro, 2005) y uno de los más utilizados, el Hill Climbing (Pe'er, 2005). Por otro lado, $\mathbf{E}_B$ se puede dividir en clases de equivalencia en las cuales estructuras diferentes dentro de la misma clase pueden describir la misma distribución de probabilidad (Chickering, 2002). La principal ventaja de las clases de equivalencia reside en que se evita el movimiento dentro de una misma clase (Chickering, 2002). Sin embargo, incluso la búsqueda de clases de equivalencias no es trivial, sino que es de orden exponencial (Acid, 2003, Gillispie, 2006). Así, dada la complejidad de hallar clases de equivalencia, diseñar algoritmos más exactos para su inferencia es todavía un problema abierto (Daly, 2011). Entre los trabajos más recientes y con mejor desempeño se destacan extensiones del algoritmo greedy (Zhang, 2013), algoritmos evolutivos (Larrañaga, 2013) y aquellos que utilizan optimización basada en colonia de hormigas. En la última categoría se encuentra el algoritmo llamado ACO-E propuesto (Daly, 2009) y el propuesto en (Pinto *et al*., 2009) llamado MMACO.

En este artículo se propone un nuevo algoritmo para el aprendizaje de clases de equivalencia de redes bayesianas extendiendo el trabajo previo en la categoría de colonia de hormigas, combinando el enfoque competitivo propuesto en (Pinto *et al*., 2009) y la extensión del algoritmo greedy usando operadores para moverse entre clases de equivalencia (Chickering, 2002). El enfoque competitivo de hormigas difiere del modo clásico en el sentido que un conjunto de hormigas construye un solo camino o solución en lugar de que cada hormiga construya su propio camino como en (Daly, 2009). De esta forma, se reduce la complejidad computacional y por tanto se obtienen mejores resultados que el algoritmo de búsqueda Greedy clásico (Chickering, 2002) en términos del número de evaluaciones de la función de puntaje $\boldsymbol{S}_f$ y del número de interacciones erróneas en la estructura aproximada. Finalmente, este artículo describe la aplicación del algoritmo propuesto en la inferencia de interacciones de naturaleza estocástica entre moléculas que responden en relación a cambios de la molécula EVI-1 y su aporte al posible descubrimiento de interacciones reales entre ellas. La razón por la cual se escogió este gen, es su asociación a una de las formas más severas de un tipo de patología tumoral denominado Leucemia Mieloide Aguda (LMA) (Wieser. 2007).

Materiales y métodos

Modelado bayesiano para el aprendizaje de redes bayesianas

Una red bayesiana es una forma de representar el conocimiento de relaciones entre elementos a través de un digrafo acíclico y un conjunto de probabilidades condicionales que cuantifican dichas relaciones, de tal forma que codifican una distribución de probabilidad conjunta de los elementos. Como un ejemplo, una red bayesiana puede representar la regulación en la expresión de un gen por otros dos genes como se muestra en la figura 1(a). En este caso, cada gen X_i tiene dos posibles estados: estado 1 si el gen está activo o estado 2 si no lo está. La regulación en la red genética es cuantificada por los parámetros de la red bayesiana. En la figura 1(b) el parámetro θ_{311} denota la probabilidad de que X_3 esté activo cuando X_1 y X_2 están activos. Esto es, cuando ambos genes X_1 y X_2 están expresados pueden actuar sobre el gen X_3 activando su expresión. Las demás relaciones siguen el mismo razonamiento, permitiendo cuantificar las relaciones entre los elementos. Formalmente una Red Bayesiana es una dupla (G,Θ) que representa la distribución de probabilidad conjunta:

$$p(X_1, X_2, \ldots, X_n) = \prod_{i=1}^{n} p(X_i | \Xi_i, G, \Theta) \qquad (2)$$

donde $G = (X, A)$ es la estructura de la red representada por un digrafo acíclico tal que sus nodos representan el conjunto de variables aleatorias $X = \{X_1, X_2, ..., X_n\}$

con $X_i \in \{1,..., r_i\}$y $A \subset X^2$ define el conjunto de arcos dirigidos entre los nodos.

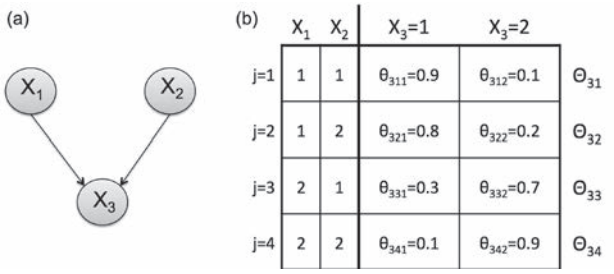

Figura 1. (a) Ejemplo de la estructura de una red bayesiana ***G*** de tres variables donde la variable X_3 tiene dos padres. Una relación o arco es el par (X_1, X_3). (b) Ejemplo de los parámetros para X_3 denotados como Θ_3. Cada elemento de la tabla representa una probabilidad condicional.

Además $\Xi_i = \{X_k | (X_k, X_i) \in A\}$ describe el conjunto de padres de X_i, y Θ es el conjunto de relaciones entre las variables en $\boldsymbol{X}$ definido como:

$$\Theta = \bigcup_{i=1}^{n} \Theta_i, \Theta_i = \bigcup_{j=1}^{q_i} \{\Theta_{ij}\}, \Theta_{ij} = \bigcup_{k=1}^{r_i} \{\theta_{ijk}\} \quad (3)$$

donde θ_{ijk} representa la probabilidad, $p(Xi = v_k | \Xi_i = w_j)$, w_j es la $j^{ésima}$ combinación del conjunto $\Omega = \{(v_1,...,v_\beta,...,v_{|\Xi_i|}) | \forall\beta, X_\beta = v_\beta, X_\beta \in \Xi_i\}$ y $q_i = |\Omega|$. En la figura 1(b), Θ_3 es el conjunto de todos los valores dentro de la tabla, Θ_{3j} es el conjunto de valores de la fila j y θ_{311} es igual a $\boldsymbol{p}(\boldsymbol{X}_3 = 1 | \Xi_3 = w_1$. Entonces, el problema a resolver es inferir la dupla ($\boldsymbol{G}$, Θ) que se ajuste a un conjunto de datos. Estudios se han realizado para obtener el conjunto Θ cuando la estructura $\boldsymbol{G}$ es conocida (Heckerman, 1995; Jensen, 2007). Sin embargo, obtener la estructura es un problema NP-Duro. Formalmente, el modelado bayesiano para aproximar una estructura G se basa en que, dado un conjunto de datos con $\mathbf{D} = [\mathbf{d}_1, \mathbf{d}_2,..., \mathbf{d}_m]$ con $\mathbf{d}_h$ como el estado de las variables en $\boldsymbol{X}$ para un caso **h** y dada la hipótesis que dichos datos fueron generados a partir de la distribución de probabilidad conjunta derivada de la red bayesiana con estructura *G*, la aproximación bayesiana describe cómo se actualiza la probabilidad de que esa red genera dichos datos (Cooper, 1992; Heckerman, 1995; Jensen 2007). Así, la probabilidad de obtener *G* a partir de **D** está dado por:

$$p(G|\mathbf{D}) = (p(\mathbf{D}|G)/\ p(\mathbf{D}))\ p/G) \quad (4)$$

donde $p(G)$ es la probabilidad a priori de la estructura hipotética, $p(\mathbf{D})$ es una constante de normalización y $p(\mathbf{D}|G)$ es la probabilidad marginal de **D** dada la red con estructura *G*. Asumiendo que $p(G)$ es uniforme para todas las posibles estructuras, el problema de encontrar la función de puntaje S_f, se reduce a calcular la probabilidad que una red con estructura *G* pueda generar un conjunto de datos **D**, p(**D**|G).

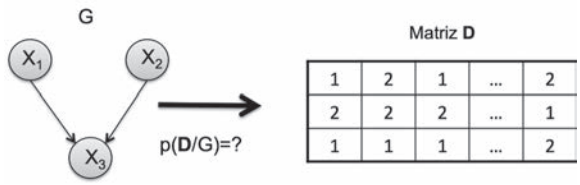

Figura 2. El problema de aprendizaje estructural, para el caso de 3 variables se reduce a encontrar la probabilidad $p(\mathbf{D}|G)$ de que la matriz **D** de tres filas sea generada por la red con estructura *G*.

La figura (2) muestra la matriz **D** asociada a la estructura de la figura (1(a)). Cada fila *i* de **D** está asociada a la variable X_i y cada columna *h* de **D** es un caso *h* en que todas las variables han sido generadas. La pregunta a resolver es, ¿cuál es la probabilidad que los valores en la matriz **D** sean generados por la red con estructura *G*? Esta probabilidad define la función de puntaje que puede ser obtenida siguiendo el razonamiento realizado en (Cooper, 1992; Heckerman, 1995; Pe'er, 2005) como es descrito en detalle en el apéndice A. En resumen, a partir de las siguientes suposiciones: (1) El muestreo de las variables de **X** en **D** tiene distribución multinomial para cualquier estructura. (2) Cada caso $\mathbf{d}_n$ es independiente de los demás y para cada variable X_i solo se puede tener un conjunto finito de estados. (3) Existe un valor para todas las variables en todos los casos en **D**. (4) Los parámetros asociados con cada variable en la estructura son independientes. (5) Los parámetros asociados con cada instancia de los valores de los padres de cada variable son independientes. (6) La función de densidad para los parámetros sigue una distribución de Dirichlet. De esta manera, la ecuación resultante es:

$$S_f(G) = \sum_{i=1}^{n} S(X_i|\Xi_i) \quad (5)$$

donde $s(X_i|\Xi_i)$ se define como:

$$S(X_i|\Xi_i) = \log\left[\prod_{j=1}^{q_i} \frac{\Gamma(N'_{ij})}{\Gamma(N_{ij}+N'_{ij})} \prod_{k=1}^{r_i} \frac{\Gamma(N_{ijk}+N'_{ijk})}{\Gamma(N'_{ijk})}\right] \quad (6)$$

La ecuación (5) es llamada ecuación bayesiana de Dirichlet (BD), donde $\Gamma(\cdot)$ es la función Gamma, N_{ijk} es el número de veces que la variable X_i tiene el valor v_k y los padres Ξ_i tienen la combinación de estado w_j. El valor de N_{ijk} tiene el mismo significado de N_{ijk} pero con la diferencia que es un valor asumido antes de tener el conjunto de muestras **D**. Este parámetro $\boldsymbol{N}'_{ijk}$ se ajusta a $N'_{ijk} = \eta \cdot 1/r_i q_i$ asumiendo que dos estructuras equivalentes pueden tener el mismo puntaje y que en η posibles muestras iniciales la probabilidad de que $X_i = w_k$ es $1/r_i$ y que $\Xi_i = v_j$ es $1/q_i$ en cada muestra (Daly, 2009).

Clases de equivalencia de redes bayesianas

Formalmente, como es probado en (Chickering, 2002; Jensen, 2007), se dice que dos variables X_α, $X_\beta \in X$ de una red bayesiana están direccionalmente separadas si existe una variable intermedia Xγ (diferente de X_α y X_β) y se cumple una de las siguientes condiciones:

(1) existe una configuración de la forma $X_\alpha \rightarrow X_\gamma \rightarrow X_\beta$ o de la forma $X_\alpha \leftarrow X_\gamma \leftarrow X_\beta$ y se conoce el estado de la variable X_γ, (2) existe una configuración de la forma $X_\alpha \leftarrow X_\gamma \rightarrow X_\beta$ y se conoce el estado de la variable X_γ o (3) existe una configuración de la forma $X_\alpha \rightarrow X_\gamma \leftarrow X_\beta$ y no se conoce el estado de la variable X_γ. Si X_α está d-separada de X_β dado el conocimiento del estado de la variable X_γ se dice que X_α y X_β son condicionalmente independientes dado X_γ y se denota ($X_\alpha \perp X_\beta | X_\gamma$). De tal forma X_α, X_β y X_γ, satisfacen la siguiente relación,

$$p(X_\alpha \left| X_\gamma, X_\beta \right.) = p(X_\alpha \left| X_\gamma \right.) \quad (7)$$

Dados estos conceptos, dos estructuras G y G′ son equivalentes si y sólo si, para toda red bayesiana B = (G, Θ) existe una red Bayesiana B′ = (G′, Θ′) tal que **B** y **B**′describen el mismo conjunto de independencias y tienen la misma distribución de probabilidad (Chickering 2002, Jensen 2007). De forma similar, dos estructuras **G** y **G**′son equivalentes si y sólo si tienen el mismo esqueleto y las mismas estructuras-v (Chickering, 2002).

Representación de Clases de Equivalencia

Dado el espacio no vacío $\mathbf{E}_B$ de estructuras de redes bayesianas, la relación entre redes equivalentes en dicho espacio, denotada como ~, cumple las propiedades de reflexividad, simetría y transitividad, formando todas ellas una clase de equivalencia. Así, para representar de una sola forma, una clase de equivalencia, se define a un digrafo acíclico parcial P con aristas dirigidas y no dirigidas, que describa todos los grafos de dicha clase, así:

$$[P] = \{G \in E_B | \, G \sim P\} \quad (8)$$

Si la arista $X_\alpha \rightarrow X_\beta$ está presente en todo $\boldsymbol{G} \in [\boldsymbol{P}]$, se dice que $X_\alpha \rightarrow X_\beta$ es forzada y si una arista $X_\alpha \rightarrow X_\beta$ no es forzada, se dice que es reversible. Un digrafo acíclico parcial completo P^c es aquel en el que toda arista forzada es una arista dirigida, y toda arista reversible es una arista no dirigida. Dado entonces un espacio de estructuras de redes bayesianas $\mathbf{E}_B$, existe un único digrafo acíclico parcial completo por cada clase de equivalencia en dicho espacio.

Movimientos entre clases de equivalencia

El conjunto de todas las clases de equivalencia de redes bayesianas de cierto número de nodos es denominado $\mathbf{E}_E$, donde existen un conjunto de operadores para hacer posible el movimiento entre dichas clases o estados en un solo paso por medio de la transformación de un digrafo acíclico parcial completo a otro. Este documento utiliza un conjunto de seis operadores con una prueba de validación para cada uno, con la ventaja de que se puede calcular el puntaje del nuevo grafo generado solo con el puntaje local $\boldsymbol{s}(X_i | \Xi_i$ de acuerdo a la tabla 1 y la ecuación (6). Los teoremas y pruebas concernientes a la validación de los operadores se encuentran detallados en (Chickering 2002) y tienen en cuenta que: (1) Ξ_α es el conjunto de Padres de X_α, es decir, el conjunto de nodos que tienen arcos dirigidos hacia X_α. (2) Ψ_α es el conjunto de vecindad de X_α, es decir, el conjunto de nodos que tienen arcos no dirigidos hacia X_α. (3) $\Psi_{\alpha,\beta}$ es la intersección entre Ψ_α y Ψ_β. (4) $\Omega_{\alpha,\beta}$ es el conjunto de padres de X_α que son vecinos de X_β. Además, dentro de la función local de puntaje en la última columna, se utiliza la notación $\Psi^{\pm\beta}_{\underline{\alpha}}$ y $\Psi^{\underline{\pm\beta}}_{\alpha}$, que define el conjunto $\Psi_\alpha \cup \{X_\beta\}$ y $\Psi_\alpha \setminus \{X_\beta\}$ correspondientemente. Finalmente, teniendo un operador y dado un digrafo acíclico parcial $\boldsymbol{P}$, un movimiento / es una instancia de un operador aplicado a un conjunto de nodos de $\boldsymbol{P}$. $\boldsymbol{M}$ es el conjunto de todos los movimientos / válidos a partir de $\boldsymbol{P}$. Dichos operadores hacen más eficientes los algoritmos moviéndose entre subconjuntos dentro de $\mathbf{E}_B$. Cada operador tiene como entrada dos o tres nodos de un grafo acíclico parcial completo y produce un grafo acíclico parcial de acuerdo a la operación.

Colonia de hormigas en el aprendizaje

El método de optimización basado en colonia de hormigas es un algoritmo heurístico multiagente propuesto por Marco Dorigo (2005). Estos algoritmos son inspirados en el comportamiento colaborativo de las hormigas para encontrar la ruta más corta desde su nicho hasta la comida, y son utilizados con el fin de resolver problemas combinatorios de búsqueda y optimización con alta complejidad (o llamados NP-Duros).

Representación del problema

El primer elemento del problema de aprendizaje es el conjunto de componentes $\boldsymbol{C} = \{c_1, c_2, ..., c_n\}$ que se pueden combinar para obtener una solución. En este problema corresponde al conjunto de movimientos /que pueden ser realizados en un estado representado por el digrafo acíclico parcial $\boldsymbol{P}$. Un camino (o solución) $\boldsymbol{s}$ es una secuencia de movimientos $\langle \boldsymbol{l}_i, ..., \boldsymbol{l}_j \rangle$ de tal forma que si se aplica cada uno de ellos (en el exacto orden) en un digrafo acíclico parcial $\boldsymbol{P}$, dará como resultado un digrafo acíclico parcial completo que representa el estado actual. Un conjunto de *m* hormigas son los agentes que construyen una solución. Así, cada hormiga compite para incluir un nuevo movimiento al camino, hasta que todas construyen una solución **S**. Existe una feromona τ_i asociada a cada movimiento $\boldsymbol{l}_i \in \boldsymbol{C}$ en una solución **S**. El comportamiento de cada τ_i es representado por un modelo que define su incremento de acuerdo a si el mejor camino **S** contiene el movimiento $\boldsymbol{l}_i$ y disminuye exponencialmente de acuerdo a un factor ρ de evaporación. Esta actualización se aplica tanto para la mejor solución de una iteración, como al mejor camino de todas las iteraciones. En el algoritmo todas las hormigas construyen una única solución o camino escogiendo cada una, un sólo movimiento, y

Tabla 1. Conjunto de operadores que permiten pasar de una clase de equivalencia a otra (Chickering 2002).

Operador	Efecto	Prueba de Validación	Cambio en el Puntaje
$X_\alpha - X_\beta$	Añadir un arco no dirigido entre X_α y X_β.	(1) Todo camino no dirigido de X_α a X_β contiene un nodo en $\Psi_{\alpha,\beta}$. (2) $\Xi_\alpha = \Xi_\beta$.	$s(X_\beta \mid \Psi^{+\alpha}_{\alpha,\beta} \cup \Xi_\beta) - s(X_\beta \mid \psi_{\alpha,\beta} \cup \Xi_\beta)$
$X_\alpha \dashv X_\beta$	Eliminar un arco no dirigido entre X_α y X_β.	$\Psi_{\alpha,\beta}$ es un clique.	$s(X_\beta \mid \Psi_{\alpha,\beta} \cup \Xi_\beta) - s(X_\beta \mid \Psi^{+\alpha}_{\alpha,\beta} \cup \Xi_\beta)$
$X_\alpha \rightarrow X_\beta$	Añadir un arco dirigido entre X_α y X_β.	(1) Todo camino semi-dirigido de X_α a X_β contiene un nodo en $\Omega_{\alpha,\beta}$. (2) $\Omega_{\alpha,\beta}$ es un clique. (3) $\Xi_\alpha = \Xi_\beta$.	$s(X_\beta \mid \Omega_{\alpha,\beta} \cup \Xi^{+\alpha}_\beta) - s(X_\beta \mid \Omega_{\alpha,\beta} \cup \Xi_\beta)$
$X_\alpha \not\rightarrow X_\beta$	Eliminar un arco dirigido entre X_α y X_β.	Ψ_β es un clique.	$s(X_\beta \mid \psi_\beta \cup \Xi^{-\alpha}_\beta) - s(X_\beta \mid \psi_\beta \cup \Xi_\beta)$
$X_\alpha \leftarrow X_\beta$	Invertir el arco dirigido de X_α a X_β.	(1) Todo camino semi-dirigido de X_α a X_β que no incluya el arco $X_\alpha \rightarrow X_\beta$ contiene un nodo en $\Omega_{\alpha,\beta} \cup \Psi_\beta$. (2) $\Omega_{\alpha,\beta}$ es un clique.	$\mathbf{s}(X_\beta \mid \Xi^{-\alpha}_\beta) + \mathbf{s}(X_\alpha \mid \Xi^{+\beta}_\alpha \cup \Omega_{\beta,\alpha})$ $\mathbf{s}(X_\beta \mid \Xi_\beta) - \mathbf{s}(X_\alpha \mid \Xi_\alpha \cup \Omega_{\beta,\alpha})$
$X_\alpha \rightarrow X_\gamma$ $X_\gamma \leftarrow X_\beta$	Convertir en un arco dirigido los arcos no dirigidos de X_α, X_β y X_γ.	Todo camino semi-dirigido entre X_α y X_β contiene un nodo en $\Psi_{\alpha\beta}$.	$\mathbf{s}(X_\gamma \mid \Xi^{+\beta}_\gamma \cup \psi^{-\gamma+\alpha}_{\alpha,\beta}) + s(X_\beta \mid \Xi_\beta \cup \psi^{-\gamma}_{\alpha,\beta})$ $-s(X_\gamma \mid \Xi_\gamma \cup \psi^{-\gamma+\alpha}_{\alpha,\beta}) - \mathbf{s}(X_\beta \mid \Xi_\beta \cup \psi_{\alpha,\beta})$

aplicando solo aquel que produzca un mejor puntaje $\mathbf{S}_f$ hasta que no haya posibilidad de un mejor movimiento para cualquier hormiga.

Modelo de la Feromona

El modelo dinámico de la actualización de la feromona se describe de la siguiente manera:

1. Se inicializa la feromona τ_l asociada a todos los movimientos posibles desde un digrafo vacío de la siguiente manera:

$$\tau_l = \frac{1}{\left|S_f(\hat{P})\right|}, i \in \hat{\boldsymbol{s}} \qquad (9)$$

   donde $\mathbf{S}_f$ está dada por la ecuación (5).

2. Se actualiza la feromona asociada a todos los movimientos que se han realizado utilizando una función de evaporación, $\tau_l = (1 - \rho)\tau_i$.

3. En cada iteración *k*, después de que todas las hormigas han construido el mejor camino **s**, cada feromona τ_l es incrementada si el movimiento se encuentra en el dígrafo P^k que representa el camino. Así,

$$\tau_l = (1-\rho)\tau_l + \frac{\rho}{\left|S_f(P^k)\right|}, l \in \mathbf{s} \qquad (10)$$

4. Además se actualiza la feromona de aquellos elementos que se relacionan con los movimientos del mejor camino de todas las iteraciones, así:

$$\tau_l = (1-\rho)\tau_l + \frac{\rho}{\left|S_f(\hat{P})\right|}, l \in \hat{\mathbf{s}} \qquad (11)$$

Información Heurística

La información heurística η_i define el puntaje para un movimiento ***l***. Dado que un movimiento consiste en la transformación de un digrafo acíclico parcial en otro, ese cambio trae consigo un cambio en el puntaje de todo el digrafo. Sin embargo, puede ser simplificado como el cambio en la función local de puntaje $\mathbf{s}(X_i \mid \Xi_i)$ definida en la ecuación (6) para cada operador como se muestra en la tabla 1.

Regla de Probabilidad

Cuando cualquier hormiga compite por escoger el mejor movimiento a partir del grafo parcial ***P***, cada una utiliza una regla de transición pseudoaleatoria (Dorigo, 2005) que permite hacer un balance entre exploración y explotación para obtener el posible siguiente movimiento. Formalmente el siguiente movimiento *l* para una hormiga está dado por,

$$l = \begin{cases} \underset{l \in M}{\arg\max}\, \tau_l^\alpha \, \eta_l^\beta, & q \le q_0 \\ \sim p(l), & q > q_0 \end{cases} \qquad (12)$$

donde $p(l) = \tau_l^\alpha \eta_l^\beta \Big| \sum_{\mu \in M} \tau_\mu^\alpha \eta_\mu^\beta$. Los parámetros α y β en el rango [0,5] (Daly, 2009), son la potencia asignada a la feromona y a la información heurística respectivamente, q_0 es un número aleatorio con distribución unifor-

Tabla 2. Notación utilizada para cada una de las variables, grafos y parámetros dentro del algoritmo.

$\mathbf{O}$	Conjunto de todos los operadores descritos en la Tabla 1.	$\boldsymbol{M}$	Conjunto de todos los posibles movimientos a partir de un digrafo acíclico parcial.
$\mathbf{t}_{máx}$	Número máximo de iteraciones.	$\boldsymbol{m}$	Número de hormigas.
$\hat{\boldsymbol{P}}$	Mejor digrafo acíclico parcial de todas las iteraciones	$\boldsymbol{p}$	Tasa de evaporación de la feromona.
$\hat{\boldsymbol{S}}$	Mejor solución de las hormigas de todas las iteraciones.	$\boldsymbol{P}$	Digrafo acíclico parcial construido por todas las hormigas en una iteración.
$\boldsymbol{q}_0$	Valor entre [0,1] que hace un balance entre la explotación y la exploración al escoger un nuevo movimiento por una hormiga.	β	Influencia que tiene la información heurística en la regla de probabilidad
$\boldsymbol{S}$	Camino construido por todas las hormigas en una iteración.	n	Número de nodos de la red que se desea inferir.

me en el intervalo [0,1] y es el conjunto de todos los posibles movimientos para el grafo parcial .

Descripción del Algoritmo

El algoritmo propuesto, mostrado en la tabla 3, está principalmente basado en el procedimiento para el aprendizaje de estructuras de redes bayesianas donde todas las hormigas colaborativamente construyen un camino por cada iteración a partir de una estructura vacía (Pinto *et al*., 2009). Además, el aprendizaje se hace en el espacio de clases de equivalencia utilizando una representación en la cual cada solución conduce a un digrafo acíclico parcial (Daly, 2009). La notación del algoritmo es presentada en la tabla 2.

Tabla 3. Algoritmo ACO-CE: Algoritmo principal de hormigas competitivas para el aprendizaje de clases de equivalencia de redes bayesianas.

Entrada: $\mathbf{D}, \mathbf{O}, \mathbf{t}_{max}, m, p, q_0, \alpha, \beta, n$
Salida: $\hat{\boldsymbol{P}}$, $\hat{\mathbf{s}}$

M = ***o*(*P*);** $\forall o \in$ **O;** τ_l = **1/|** $S_f(P)$ **|**, $\forall$ **l** $\in \boldsymbol{M}$
Para t = 1 hasta $\boldsymbol{t}_{max}$ **hacer**
  P = ***grafo_vacío***, $s = \langle\ \rangle$
  (*P*,**s**) = $ANTS(P, \mathbf{s}, \mathbf{O}, q_0, \alpha, \beta)$
  Si $S_f(P) > S_f(\hat{P})$ **entonces ,** $\boldsymbol{P} = P$, $\hat{\mathbf{s}} = \mathbf{s}$ **Fin si**
  $\tau_l = (1 - \rho)\tau_l + \rho l |S_f(P)|, l \in \mathbf{s}$
  $\tau_l = (1 - \rho)\tau_l + \rho l |S_f(\hat{P})|, l \in \hat{\mathbf{s}}$
Fin Para

Como describe la tabla 3, por cada iteración las hormigas construyen un camino, se escoge el mejor camino de todas las iteraciones y se actualiza el valor de la feromona asociada a cada movimiento. La construcción de una solución por el conjunto de hormigas es descrito en la función ANTS, en la tabla 4. A partir de un digrafo acíclico parcial $\boldsymbol{P}$ vacío, las hormigas construyen el camino de tal forma que cada una haya todos los posibles movimientos a partir del actual $\boldsymbol{P}$ y escoge uno de los movimientos de manera pseudoaleatoria de acuerdo a la regla de probabilidad dada en la ecuación (12). Al final, se aplica dicho movimiento a $\boldsymbol{P}$ obteniendo un nuevo digrafo $\boldsymbol{P}_k$ para la hormiga k. Cuando todas las hormigas han escogido un posible siguiente movimiento, se escoge la solución s_r, donde $r \sim U[1,...,m]$. Si hubo al menos una hormiga que realizó un mejor movimiento, es decir, si $S_f(P_r) > S_f(P)$, las hormigas vuelven a competir por el siguiente movimiento. De otro modo, se retorna a la solución P, se actualiza el mejor camino de todas las iteraciones y se actualiza el valor de la feromona de acuerdo a las ecuaciones (10) y (11).

Tabla 4. Algoritmo ANTS: Algoritmo que describe la competencia de las hormigas para generar un digrafo acíclico parcial en una iteración del algoritmo principal.

Entrada: P, **s**, **O**, q_0, α, β
Salida: P, **s**

Mientras *switch* = 0 **hacer**
  switch = 1
  Para k = *1* hasta m **hacer**
    $\boldsymbol{M} = \boldsymbol{o}(P); \forall\ o \in \mathbf{O}$
    Si $|M| = 0 \vee \max_{l \in M} \tau_l^\alpha\ \eta_l^\beta \leq 0$ **entonces** Devolver (*P*, **s**) **Fin si**
    $q \sim U[0, 1]$,

$$l = \begin{cases} \arg\max_{l \in M} \tau_l^\alpha\ \eta_l^\beta, & q \leq q_0 \\ \sim p(l), & q > q_0 \end{cases}$$

$$p(l) = \tau_l^\alpha\ \eta_l^\beta \Big/ \sum_{\mu \in M} \tau_\mu^\alpha\ \eta_\mu^\beta \text{ , } P_k = \textit{Aplicar } l \textit{ a } P,$$

    s = $\langle$**s, l**$\rangle$, $\boldsymbol{S}_f(\boldsymbol{P}_k) = \boldsymbol{S}_f(\boldsymbol{P}) + \eta_l$
  Fin para
    $r \sim U[1,...,m]$, $S_b = S_f(P_r)$
    Si $S_b > S_f(P)$ **entonces** $P = P_r$, **s** = $\mathbf{s}_r$ **sino** *witch = 0* **Fin si**
Fin mientras

Aproximación de una red de regulación genética

El algoritmo propuesto puede ser utilizado para mejorar el entendimiento de redes de regulación genética a partir de microarreglos de ADN. Un microarreglo de ADN es un chip que contiene información relativa al nivel de expresión de un conjunto de genes en una célula cuándo ésta última es estimulada. Este valor se obtiene al comparar el nivel de expresión genética de una célula de referencia con la información genética de una célula blanco (Wang, 2011). Por ejemplo, una célula de referencia puede ser una célula sana, y la célula blanco una anormal (Meloni, 2004). Matemáticamente, se puede expresar la relación entre la expresión del gen *i* en la célula blanco B_i con respecto a su expresión en la célula de referencia R_I como $T_i = \log_2(B_i/R_i)$. Cuando $T_i > 1$ se dice que el gen *i* está activado en la célula blanco, y si $T_i <$ i se dice que el gen *i* en la célula blanco está inhibida. De esa forma la discretización mas simple, esto es, de tres estados (inhibido, neutro, activado) es sencilla de hacer (Quackenbush 2002). Generalizando, si el número de células blanco es igual a *m*, para las cuales se desea obtener la información del nivel de expresión de *n* genes, el conjunto de datos final es una matriz **D** de $n \times m$ donde d_{ij} representa el radio T_i para la célula blanco *j*, donde $i = 1,...,n$ y $j = 1,...,m$. A partir de esta matriz es posible obtener una aproximación de una red bayesiana que represente la red de regulación genética deseada, dependiendo de que genes se escojan.

Características del microarreglo utilizado

Como ejemplo, se seleccionó el microarreglo con número de acceso GSE6891 de la base de datos Gene Expression Omnibus (www.ncbi.nlm.nih.gov/geo). Este microarreglo contiene datos de expresión genética de grupos de pacientes con Leucemia Mieloide Aguda (LMA) con el objetivo de estudiar su aplicabilidad en la predicción de subclases asociadas a una anormalidad específica en la LMA (Verhaak, 2009). Dentro de sus resultados se encontró que dentro de una de aquellas subclases, uno de los genes más discriminantes para detectar anomalías de dicha clase fue el gen EVI-1 (o también llamado MECOM). Estas características hacen de este conjunto de datos propicio y viable para ser utilizado en el aprendizaje de la red. Específicamente el conjunto de datos cuenta con información de 461 muestras tomadas de la sangre o de la médula ósea. Para ilustrar el uso del algoritmo de hormigas propuesto en la sección anterior, de este microarreglo se obtuvieron solamente las filas de genes relevantes los cuáles tuvieron cambios de 6 veces o más en su valor cuando EVI-1 cambió su valor. De esta forma se obtiene un conjunto de genes relevantes en diferentes casos con respecto a EVI-1. Este es el conjunto de datos **D** utilizado por el algoritmo para aproximar la red de interacciones.

Obtención de una Red Semilla

Antes de realizar el proceso de aprendizaje es posible extraer conocimiento previo de interacciones moleculares. Esta información es almacenada en bases de datos. Se escogió la base de datos Biogrid (http://thebiogrid.org/) por ser la base de datos de acceso libre más completa, en la que se encontraban más interacciones, y en la que se mantenía el formato de los nombres de las moléculas como en los microarreglos. Para extraer una red inicial a partir de dicha información, se utilizó el algoritmo basado en el método descrito en (Djebari 2008) puesto que minimiza la redundancia en los datos. El algoritmo es el siguiente: (1) Se obtiene la red inicial de la biomolécula en estudio (EVI-1/MECOM), (2) de cada uno de los elementos de la red anterior se obtiene su propia red (ya que las interacciones de EVI-1 son muy pocas), (3) se unen las redes filtrando las interacciones repetidas, (4) se filtran las interacciones de acuerdo a los genes listados en el microarreglo y se forma una red unidireccional. (5) Finalmente se aplica el algoritmo en (Djebbari 2008) basado en la técnica depth-first para convertir la red anterior en una grafo acíclico no dirigido. Teniendo entonces el conjunto de datos con los genes relevantes, entonces de la red inicial son utilizados aquellas interacciones cuyos genes se encuentran en el microarreglo.

Resultados y discusión

Evaluación del Algoritmo de Aprendizaje

La metodología de evaluación del aprendizaje que se utilizó se basó en la descrita en (Daly, 2009), donde: (1) Se selecciona la red ASIA (figura 3(a)) como red teórica o estándar (http://compbio.cs.huji.ac.il/Repository/) para comparar los algoritmos Greedy (Chickering, 2002), ACO-E (Daly 2009) y el nuevo algoritmo propuesto. (2) Se selecciona un conjunto de valores para los parámetros del algoritmo propuesto, manteniendo el número de hormigas *m* igual a 15. Los conjuntos de valores son ρ =[0.05 0.1 0.3 0.5], q_0 =[0.3 0.5 0.7], α =[2 3 4 5] y β =[1 1.5 2]. (3) Para el algoritmo ACO-E se mantuvieron los parámetros obtenidos en (Daly 2009), es decir, $\rho = 0.3$, $q_0 = 0.7$, $\alpha = 1$ y $\beta = 2.5$. (4) Para cada una de las combinaciones de los valores de los parámetros se hicieron las simulaciones respectivas para los tres algoritmos utilizando 20 conjuntos de datos con 20, y 500 muestras generadas a partir de la red bayesiana ASIA. El rendimiento y la red o estructura obtenida por cada uno de los algoritmos se evaluaron utilizando las siguientes métricas: (a) la función de Puntaje ($\boldsymbol{S}_f$) para evaluar la cercanía de la red construida y los datos según la ecuación (5) y (6). Valores altos indican alta probabilidad de que los datos hayan sido generados por el grafo obtenido. (b) Distancia Estructural de Hamming (SHD): Es la diferencia entre los arcos añadidos, omitidos e invertidos (en dirección) entre la estructura generada por el algoritmo y la

estructura de la cual se generaron los datos iniciales. Una distancia menor significa entonces, que el digrafo aproximado es mas cercano al teórico. (c) Número de evaluaciones o llamadas estadísticas (NSC) que mide la complejidad computacional del algoritmo al calcular el número de llamadas que hace el algoritmo a la función objetivo. (5) Para la función de puntaje se escogió el tamaño de muestreo equivalente η = 4 utilizado en (Daly, 2009). En la figura 3(b)-(d) se muestra el resultado del protocolo descrito anteriormente para la red ASIA, para una simulación utilizando 500 muestras. La primera red, es la red teórica de la cual se obtienen los datos, las demás son las redes aproximadas por los algoritmos GES, ACO-E y el algoritmo propuesto, respectivamente. El número total de combinaciones para los valores de los parámetros del algoritmo fue 144. Como se muestra en la tabla 5 estas combinaciones se clasificaron de acuerdo a los valores obtenidos en las métricas S_f, SHD y NSC para los tres algoritmos probados. De acuerdo a la segunda fila de la tabla 5, sin importar el número de muestras y la combinación de parámetros, el nuevo algoritmo propuesto tiene una mayor probabilidad de generar los datos que el algoritmo Greedy.

Para el criterio utilizado con respecto a SHD, se presentaron 21 combinaciones de los parámetros para los cuales el algoritmo propuesto tuvo menor valor con respecto a los otros dos algoritmos, es decir es mas eficiente al obtener un grafo mas cercano al teórico. En promedio, se obtuvo un 2,6 % y 4,8 % menor valor que para los algoritmos Greedy y ACO-E respectivamente (figura 4(a-b)). De acuerdo a la suma cuadrática de las diferencias en los valores del SHD de estas 21 combinaciones, la mayor diferencia fue para la combinación ρ = 0,1, q_0 = 0,3, α = 5 y β = 1.

Tabla 5. Clasificación de los resultados al protocolo descrito. En la tercera y cuarta columna se muestran el número de combinaciones, de las 144 en total, que cumplieron con el criterio de las métricas mencionadas para 20 y 500 muestras de datos iniciales.

	Criterio Clasificación	20 Muestras	500 Muestras
1	$Sf_{Propuesta} > Sf_{ACO\text{-}E}$	1	4
2	$Sf_{Propuesta} > Sf_{GES}$	144	144
3	$SHD_{Propuesta} < SHD_{ACO\text{-}E}$ y $SHD_{Propuesta} < SHD_{GES}$	21	41
4	$SHD_{Propuesta} < SHD_{GES}$	24	66
5	$Sf_{Propuesta} > Sf_{ACO\text{-}E}$ y $SHD_{Propuesta} < SHD_{ACO\text{-}E}$	0	2
6	$NCS_{ACO\text{-}E} > NCS_{Propuesta}$	0	129

Como es descrito en la figura 4(c), la diferencia de combinaciones erróneas del grafo final del algoritmo propuesto con respecto al algoritmo ACO-E fue de 10,7% menor y con el algoritmo Greedy del 5,3 % menor. Los resultados anteriores muestran que el algoritmo propuesto puede obtener estructuras con menos interacciones erróneas que los algoritmos ACO-E y Greedy para redes pequeñas, usando 20 muestras. Para 500 muestras, según la primera fila de la tabla 5, de las 144 combinaciones de los parámetros, en cuatro de ellos el algoritmo propuesto obtuvo mayor valor en S_f con respecto al algoritmo ACO-E. De estas cuatro, aquella combinación con mayor diferencia en la métrica SHD, ρ = 0,5, q_0 = 0,7, α = 5 y β = 1, obtuvo un valor del 10,4 % y 4 % de menos combinaciones erró-

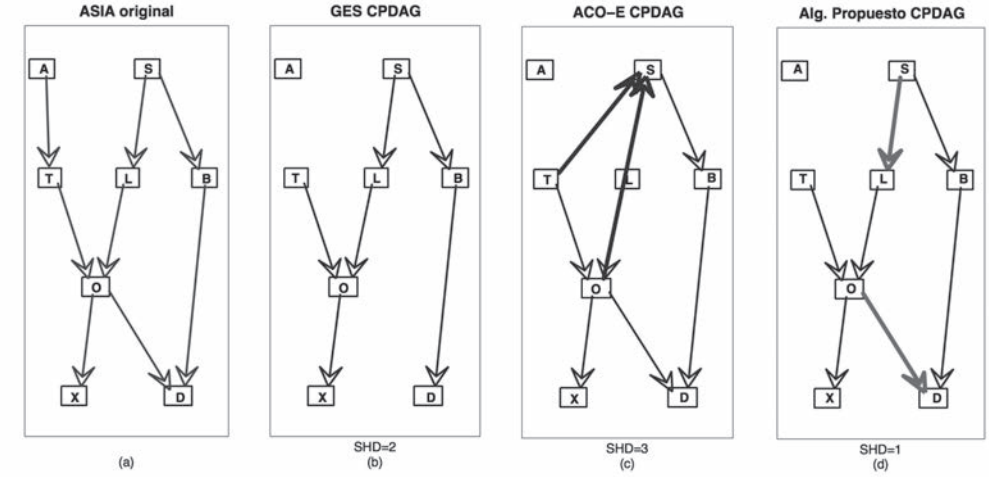

Figura 3. Ejemplo predicción de la estructura de la red bayesiana ASIA. (a) Red original, (b) Red aproximada por el algoritmo GES, (c) ACO-E y (d) el algoritmo propuesto en este artículo con 500 muestras. Los arcos rojos en el grafo aproximado por ACO-E son arcos que no corresponden al grafo original, y los arcos en verde son arcos aproximados por el algoritmo propuesto.

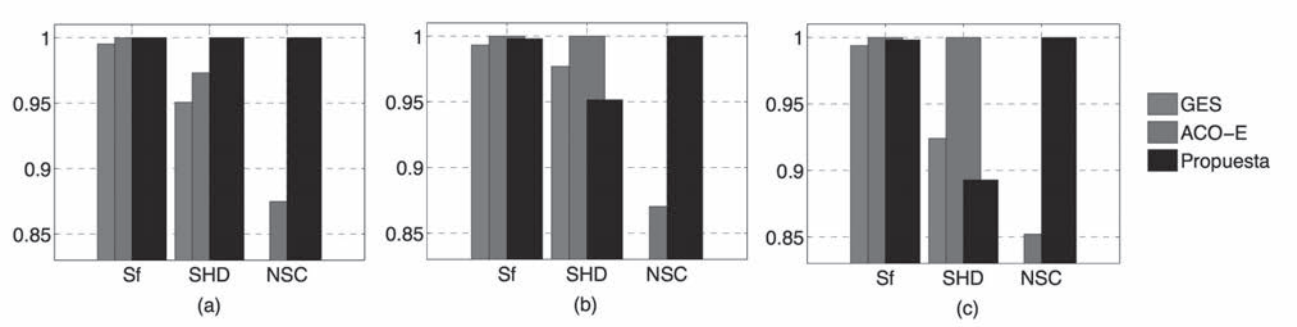

Figura 4. Resultados obtenidos en las métricas S_f, SHD y NSC en el protocolo descrito para los tres algoritmos probados utilizando datos con 20 muestras. Un Sf mayor, un SHD menor y un NSC mayor implican mejor desempeño. (a) Resultados para la combinación ρ = 0,1, q_0 = 0,7, α = 3 y β = 1 referido en la fila 1 de la tabla 5. (b) Resultados promedio para las combinaciones referidas en la fila 3 de la tabla 5. (c) Resultados para la mejor combinación ρ = 0,1, q0 = 0,3, α = 5 y β = 1 de las referidas en la fila 3 de la tabla 5. Los valores están normalizados.

neas en el grafo final con respecto al algoritmo GES y ACO-E respectivamente, (figura 5(a)). En la tercera fila de la tabla 5, de las 144 combinaciones se obtuvieron 41 para las cuales el valor de SHD del algoritmo propuesto fue menor que los otros dos. Específicamente, en promedio el valor fue de un 5% y 9% menor con respecto al algoritmo ACO-E y Greedy respectivamente como se muestra en la figura 5(b). Según la quinta fila de la tabla 5, en dos de ellas se obtuvo un mayor valor para la función de puntaje. Una de ellas es la mostrada en la figura 5(a), y la otra es la combinación $\rho = 0{,}5$, $q_0 = 0{,}3$, $\alpha = 5$ y $\beta = 1$, la cual obtuvo un valor del SHD 3% menor para el algoritmo propuesto con respecto a los otros dos algoritmos. La última fila de la tabla 5 hace referencia a la medida de complejidad del algoritmo NSC. El número de llamadas a la función de puntaje por el algoritmo propuesto fue menor que en el algoritmo ACO-E en el 90% de las combinaciones utilizando 500 muestras.

En promedio esta diferencia fue del 6,8%, denotando que la complejidad del nuevo algoritmo es menor. Adicionalmente, para las combinaciones referidas en la fila 3 de la tabla 5, mostradas en la figura 5(b), en promedio el algoritmo propuesto obtuvo una disminución del 4,6 % en la métrica SHD con respecto al algoritmo ACO-E. Y por último en promedio para la combinación mostrada en la figura 5(c), referida en la fila cinco de la tabla 5 obtuvo un valor del NSC del 5,8 % menor que el algoritmo ACO-E. Todo lo anterior indica que el algoritmo propuesto, para la mayoría de combinaciones, usando 500 muestras de redes pequeñas tiene un orden de complejidad menor que el algoritmo ACO-E. Además, para las combinaciones referidas en la fila 5 de la tabla 5 y mostradas en la figura 5, tienen la posibilidad de obtener estructuras con menos interacciones erróneas (falsos positivos) que los algoritmos ACO-E y Greedy, y a su vez con mayor o muy semejante probabilidad como indica la función de puntaje.

Aproximación de la red de regulación genética

Dado que es posible que haya información espuria en varios casos en la tabla de datos, que produzcan

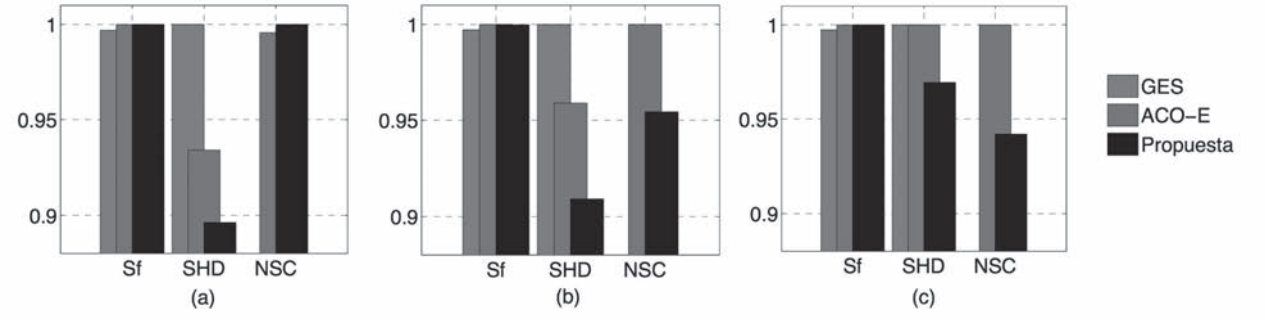

Figura 5. Resultados obtenidos en las métricas S_f, *SHD y NSC* en el protocolo descrito para los tres algoritmos probados utilizando datos con 500 muestras. Los valores están normalizados. (a) Resultados para la mejor combinación ρ = 0,5, q0 = 0,7, α = 3 y β = 1. (b) Resultados promedio para las combinaciones referidas en la fila 3 de la tabla 5. (c) Resultados para la combinación ρ = 0,5, q0 = 0,3, α = 3 y β = 1.

posibles falsos positivos, se infirieron un conjunto de *M* estructuras y se seleccionaron las interacciones en común de todas ellas. Para cada estructura G_i obtenida se obtiene su función de puntaje o probabilidad de aproximar la información en los datos $S_f(G_i)$ y a partir de ellas se obtiene un medidor de confianza por el promedio de las probabilidades, $conf(G) = \sum_{i=1}^{M} S_i(G_i)$. El resultado es entonces un grafo ***G*** con las interacciones en común mencionadas, con un valor de confianza *conf*(*G*) (Pe'er, 2005). Para los datos del microarreglo con los genes relevantes filtrados, el resultante modelo ***G*** inferido por el algoritmo propuesto, con un número de 20 arcos entre 16 componentes. La figura 6(b) describe la comparación del valor de confianza según la ecuación (14) obtenido para las estructuras inferidas con el algoritmo propuesto, y la estructura ***G*** es presentada en en la figura 6(a) con las nuevas interacciones en rojo. Como muestran los valores, el valor de confianza obtenido por el algoritmo propuesto tiene un puntaje del 10% mayor para la estructura inferida que para la estructura inicial. El algoritmo es capaz de inferir interacciones de naturaleza estadística, y a pesar de que tiene la posibilidad de obtener estructuras con menos interacciones erróneas, las interacciones reales sólo son comprobables a través de métodos experimentales. Sin embargo, estas nuevas interacciones obtenidas con el algoritmo son de potencial importancia al elucidar la exacta interacción que hay entre los elementos.

Conclusiones

El principal resultado de este trabajo fue el desarrollo de un algoritmo de aprendizaje de clases de equivalencia basado en modelos de búsqueda inspirado en colonia de hormigas. Además se describieron las ventajas del enfoque de puntaje y búsqueda entre clases de equivalencia de redes bayesianas. Principalmente, se evita el movimiento redundante por medio de operadores. Los resultados mostraron que el algoritmo propuesto para redes pequeñas, usando 20 muestras, y en específicas combinaciones de sus parámetros, tiene la posibilidad de obtener estructuras con menos interacciones erróneas (falsos positivos) que los algoritmos ACO-E y Greedy. Además, los resultados mostraron que, para 500 muestras, y redes pequeñas, el algoritmo propuesto tiene la posibilidad de obtener estructuras con menos interacciones erróneas (falsos positivos) que los algoritmos ACO-E y Greedy, y a su vez con mayor o muy semejante probabilidad como indica la función de puntaje y con un orden de complejidad menor que el algoritmo ACO-E. Además, se describió la aplicación de la inferencia de interacciones de naturaleza estocástica entre moléculas que responden en relación a cambios de la molécula EVI-1 y su aporte al descubrimiento de interacciones reales entre ellas. Se describieron los conceptos básicos sobre microarreglos necesarios para la aplicación del algoritmo de aprendizaje de redes bayesianas. Se mostró como es posible emplear datos de microarreglo para la selección de genes relevantes de acuerdo a cambios significativos cuando hay cambios en las condiciones celulares. También se mostró la utilidad de utilizar información a priori de la literatura científica para direccionar mas precisamente el aprendizaje. Por último se mostró los resultados de la aplicación del algoritmo a un microarreglo obtenido por muestras de pacientes con Leucemia Mieloide Aguda (LMA) con el fin de encontrar interacciones con dependencias estadísticas como potenciales interacciones biológicas con alta probabilidad. Este último resultado es de gran importancia puesto que aporta en la generación de conocimiento en investigaciones de como interactúan los genes relacionados con enfermedades como la

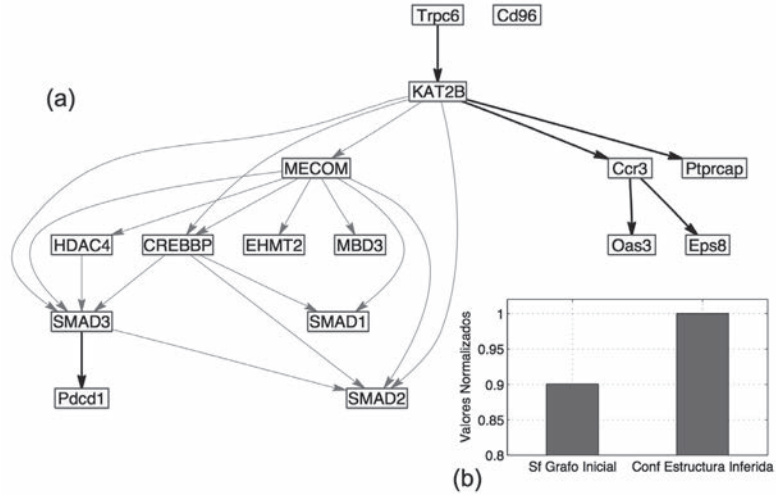

Figura 6. (a) Estructura inferida por el algoritmo a partir de los datos del microarreglo con los genes relevantes. Se infirieron 6 interacciones nuevas, de las cuales tres fueron con el gen SMAD3 y 5 con KAT2B. (b) Comparación del valor de puntaje de la red semilla inicial y el valor de confianza estimado por la estructura inferida con el algoritmo propuesto.

LMA. Por ejemplo, es posible que al expresarse uno de los genes directamente regule al otro en la dirección aproximada por el algoritmo. También es posible que la regulación entre los genes esté en la dirección inferida, pero puede ser mediada por otros genes que no fueron tenidos en cuenta como genes relevantes en la sección anterior. Por ejemplo los genes SMAD3 y Pdcd1 aproximados por el algoritmo podrían tener genes intermedios. Por último, las interacciones pueden llegar a ser inversas, es decir, en cualquiera de los casos anteriores es posible que la dirección de la regulación sea contraria a la que se aproximó con el algoritmo.

Apéndice A: Deducción de la función de puntaje

Siguiendo el razonamiento realizado en (Cooper, 1992; Heckerman, 1995) para obtener la probabilidad P(**D**|G) en la ecuación (5) y (6), se tiene en cuenta que:

1. Se supone que el muestreo de las variables de **X** en **D** tiene distribución multinomial para cualquier estructura *G*. Entonces, dado **D** como el arreglo de los d_{ih} y como los valores de Ξ_i en $\boldsymbol{d}_h$ donde $\sigma_i = \{\ell | X_\ell \in$, para todo *G* en $\mathbf{E}_B$, existe un conjunto de valor positivo Θ tal que,

$$p(d_{ih} = v_k \left| D_{\sigma_i h} = w_j \right.) = \theta_{ijk} \tag{13}$$

donde $\sum_{k=1}^{r} \theta_{ijk}$ Dada la existencia de los parámetros $\Theta=\{\theta_{ijk}\}$, se define la ecuación de puntaje (4) teniendo en cuenta la incertidumbre que se tiene de ellos, calculando el promedio de la probabilidad $P(\mathbf{D}|G)$ sobre todos los posibles valores de los parametros en *G*, así:

$$p(D|G) = \int_{\Theta} p(D|G,\Theta) f(\Theta|G) d\Theta \tag{14}$$

Entonces se requiere definir de (14) la probabilidad $p(\mathbf{D}|G,\Theta)$ con respecto a los datos y $f(\Theta|G)$ con respecto a los parámetros.

2. Se asume que cada caso $\mathbf{d}_h$ es *independiente* de los demás y para cada variable X_i solo se puede tener un conjunto finito de estados, de tal forma que:

$$p(\boldsymbol{D}|G,\Theta) = \prod_{h=1}^{m} p(d_h|G,\Theta) \tag{15}$$

3. Se supone que *existe un valor* para todas las variables en todos los casos en **D**, entonces la expresión (15) se puede escribir como,

$$p(\boldsymbol{D}|G,\Theta) = \prod_{i=1}^{n} \prod_{h=1}^{m} p(d_{ih}|G,\Theta) \tag{16}$$

Sin embargo, dado que *G* establece los elementos de Ξ, se puede descomponer la expresión anterior de acuerdo a las contribuciones locales de cada variable,

$$p(\boldsymbol{D}|G,\Theta) = \prod_{i=1}^{n} \prod_{h=1}^{m} p(d_{ih}|G,\Theta) \tag{17}$$

Además, si se define N_{ijk} como el número de veces que d_{ih} es igual a un valor v_k y $D_{\sigma_i h}$ a un elemento $w_j \in \Omega$ para todo *h* en **D**, la ecuación anterior puede reagruparse de la siguiente manera,

$$p(\boldsymbol{D}|G,\Theta) = \prod_{i=1}^{n} \prod_{j=1}^{q_i} \prod_{k=1}^{r} p(d_{ih} = V_k \left| D_{\sigma_i h} = w_j \right.)^{N_{ijk}} \tag{18}$$

Usando (13) y (18) se tiene que

4. Se asume que los parámetros asociados con cada variable en la estructura son independientes, entonces se tiene que:

$$p(\boldsymbol{D}|G,\Theta) = \prod_{i=1}^{n} \prod_{j=1}^{q_i} \prod_{k=1}^{r} p(d_{ih} = V_k \left| D_{\sigma_i h} = w_j \right.)^{N_{ijk}} \tag{19}$$

A esta propiedad se le llama Independencia Paramétrica Global.

$$f(\Theta|G) = \prod_{i=1}^{n} f(\Theta_i|G) \tag{20}$$

5. Se asume que los parámetros asociados con cada instancia de los valores de los padres de cada variable son independientes. Esto se puede expresar como:

$$f(\Theta|G) = \prod_{i=1}^{n} \prod_{j=1}^{q_i} f(\Theta_{ij}|G) \tag{21}$$

Esta propiedad tiene el nombre de Independencia Paramétrica Local. Así a partir de (19) y (21), la ecuación (14) se puede reescribir de la siguiente forma:

$$p(\boldsymbol{D}|G) = \prod_{i=1}^{n} \prod_{j=1}^{q_i} \int_{\Theta} \left[\prod_{k=1}^{r} \theta_{ijk}^{N_{ijk}} \right] f(\Theta_{ij}|G) d\Theta_{ij} \tag{22}$$

6. Por último, se supone que la función de densidad para los parámetros en Θ_{ij}, sin conocer **D** (o *a priori*), sigue una distribución de Dirichlet principalmente porque, al actualizarse cuando se hace un muestro multinomial **D**, siguen siendo Dirichlet (Heckerman 1995). Así, la función de densidad para los parámetros es como sigue,

$$p(\boldsymbol{D}|G) = \prod_{i=1}^{n} \prod_{j=1}^{q_i} \int_{\Theta} \left[\prod_{k=1}^{r} \theta_{ijk}^{N_{ijk}} \right] f(\Theta_{ij}|G) d\Theta_{ij} \tag{23}$$

donde $N_{ijk'}$ es la información a priori para las probabilidades numéricas θ_{ijk} sin tener **D**. Basado en las suposiciones anteriores, y dado que la integral en (22) describe $E\left[\prod_{k=1}^{r} \theta_{ijk}^{N_{ijk}}\right]$ con respecto a $\boldsymbol{f}(\Theta_{ij}|\mathbf{G})$, la ecuación resultante para (14) solucionando la integral según (Heckerman 1995) es:

$$p(\boldsymbol{D}|G) = p(G) \prod_{i=1}^{n} \prod_{j=1}^{q_i} \frac{\Gamma(N_{ij})}{\Gamma(N_{ij} + N'_{ij})} \prod_{k=1}^{r} \frac{\Gamma(N_{ijk} + N'_{ijk})}{\Gamma(N'_{ijk})} \tag{24}$$

Referencias bibliográficas

Acid, S. and de Campos, L. M. 2003. Searching for bayesian network structures in the space of restricted acyclic partially directed graphs. *Journal of Artificial Intelligence Research*. 18: 445–490.

Castro, P. A. D. and Von Zuben, F. J. 2005. An immune-inspired approach to bayesian networks. in *HIS '05: Proceedings of the Fifth International Conference on Hybrid Intelligent Systems*, (Washington, DC, USA), pp. 23–28, IEEE Computer Society, 2005.

Chickering, D. M. 1996. Learning bayesian networks is np-complete. Pp. 121–130.

Chickering, D. M. 2002. Learning equivalence classes of bayesian-network structures. *The Journal of Machine Learning Research*. 2: 445 – 498.

Cooper, G. F. and Herskovits, E. 1992. A bayesian method for the induction of probabilistic networks from data. *Machine Learning*. 9(4): 309–347.

Daly, R. and Shen, Q. 2009. Learning bayesian network equivalence classes with ant colony optimization. *Artificial Intelligence Research*. 35: 391–447.

Daly, R. Shen; Q. Aitken, S. 2011. Learning Bayesian networks: approaches and issues. *The Knowledge Engineering Review*. 26(2): 99–157.

de Campos, L. M.; Fernández-Luna, J. M.; Gámez, J. A., and Puerta, J. M.. 2002. Ant colony optimization for learning bayesian networks. *International Journal of Approximate Reasoning*. 31(3): 291 – 311.

Djebbari, A.; Quackenbush, J. 2008. Seeded bayesian networks: Constructing genetic networks from microarray data. *BMC Systems Biology*. 2(1): 57.

Dorigo, M.; Blum, C. 2005. Ant colony optimization theory: A survey. *Theoretical Computer Science*. 344(2-3): 243 – 278.

Du, Z.; Wang, Y.; Ji; Z. June 2009. A new structure learning method for constructing gene networks. *Bioinformatics and Biomedical Engineering, ICBBE 2009. 3rd International Conference*. Pp. 1–4.

Gillispie, S. B. 2006. Formulas for counting acyclic digraph markov equivalence classes. *Journal of Statistical Planning and Inference*. 136(4): 1410 – 1432.

Heckerman, D. 1995. A tutorial on learning with bayesian networks. *Tech. Rep., Microsoft Research*.

Jensen, F. V. and Nielsen, T. D. 2007. *Bayesian Networks and Decision Graphs*. *Springer Science + Business Media*, LLC.

Larrañaga, P.; Karshenas, H.; Bielza, C. and Santana R. 2013. A review on evolutionary algorithms in Bayesian network learning and inference tasks. *Information Sciences*. 233: 109–125.

Meloni, R.; Khalfallah, O., and Faucon Biguet, N. 2004. DNA microarrays and pharmacogenomics. *Pharmacological Research*. 49(4): 303 – 308.

Needham, C.; Manfield, I.; Bulpitt, A.; Gilmartin, P.; and Westhead, D. 2009. From gene expression to gene regulatory networks in arabidopsis thaliana. *BMC Systems Biology*. 3(1): 85.

Pe'er, D. 2005. Bayesian Network Analysis of Signaling Networks: A Primer. *Sci. STKE*, 281: l4.

Pinto, P.; Nagele, A.; Dejori, M.; Runkler, T.; and Sousa, J. 2009. Using a local discovery ant algorithm for bayesian network structure learning. *Evolutionary Computation, IEEE Transactions on*. 13: 767–779.

Quackenbush, J. 2002. Microarray data normalization and transformation. *Nature Genetics*. 32: 496-501.

Verhaak, R.G.W.; Wouters, B.J.; Erpelinck, C.A.J.; Abbas, S.; Beverloo, H.B.; Lugthart, S.; Lwenberg, B.; Delwel, R.; and Valk, P.J.M. 2009. Prediction of molecular subtypes in acute myeloid leukemia based on gene expression profiling. *Haematologica*. 94(1): 131– 134.

Wang, L. and Li, Paul C.H. 2011. Microfluidic DNA microarray analysis: A review. *Analytica Chimica Acta*. 687(1): 12 – 27.

Wieser, R. 2007. The oncogene and developmental regulator evi1: Expression, biochemical properties, and biological functions. *Gene*. 396(2): 346 – 357.

Wong, M. L. and Leung, K. S. Aug. 2004. An efficient data mining method for learning bayesian networks using an evolutionary algorithm-based hybrid approach. *Evolutionary Computation, IEEE Transactions*. 8: 378–404.

Zhang, Y.; Zhang,W.; Xie, Y. 2013. Improved heuristic equivalent search algorithm based on Maximal Information Coefficient for Bayesian Network Structure Learning. *Neurocomputing*. 117: 186–195.