Predicción de Mutaciones en Secuencias de la Proteína Transcriptasa Inversa del VIH usando Nuevos Métodos para Aprendizaje Estructural de Redes Bayesianas

Prediction of Mutation HIV-Reverse-Transcriptase Protein using New Methods for Structural Learning of Bayesian

María del Carmen Chávez, MSc., Gladys Casas, Dra., Jorge Moreira, Est., Pavel Silveira, Lic.,
Iosvany Moya, Lic., Rafael Bello, Dr. y Ricardo Grau Dr.
Centro de Estudios de Informática, Universidad Central "Martha Abreu" de Las Villas, Santa Clara, cUBA
mchavez, gladita, jmoreira, psd, rbellop, rgrau}@uclv.edu.cu, moyacuba@gmail.com

Recibido para revisión 14 de Abril de 2008, Aceptado 19 de Mayo de 2008, Versión final 27 de May de 2008

Resumen—**En el análisis de grandes volúmenes de datos es crucial la relación entre las variables. Una de las formas de modelar tales relaciones es usar una red bayesiana. El costo computacional del aprendizaje de una red bayesiana desde datos, crece con el número de variables en la misma y con el número de casos, por consiguiente surge el problema de identificar algoritmos eficientes que aprendan desde los datos. En el trabajo se proponen tres nuevos métodos de aprendizaje estructural de redes bayesianas, dos de ellos se basan en las relaciones de dependencia entre las variables usando la prueba Chi cuadrado y el último hace uso de heurística mediante el algoritmo evolutivo Optimización de Enjambre de Partículas (PSO, de sus siglas en inglés: Particle Swarm Optimization). Los algoritmos propuestos se han probado con varios conjuntos de datos de la UCI Repository de Machine Learning y en el trabajo se muestran resultados en la predicción de mutaciones en secuencias de ADN de la proteína transcriptasa inversa del VIH (Human Immunodeficiency Virus).**

Palabras Clave—**Redes Bayesianas, Clasificación, CHAID (Chi-Squared Automatic Interaction Detector), PSO, (Particle Swarm optimization) y Transcriptasa Inversa**

Abstract— **In the analysis of large volumes of data it is important to take into account the relationship among variables. One of the ways to model these relations is given by the use of Bayesian Networks (BN). The computational cost of learning from data in BN, grows with the increase of the number of variables and cases; therefore, it is important to create efficient algorithms which learn from data. Three different methods for structural learning in BN are proposed in this paper. Two of them are based on the chi-square statistical test and the last one is based on a bio-inspired technique: Particle Swarm Optimization. All the algorithms have been tested using several datasets available on the UCI Repository of Machine Learning. The present work also shows results in the prediction of mutations in DNA sequences of the HIV-reverse-transcriptase protein.**

Keywords—**Bayesian Networks, Classification, CHAID (Chi-Squared Automatic Interaction Detector), PSO, (Particle Swarm optimization) and Reverse Transcriptase.**

I. INTRODUCCIÓN

Las redes bayesianas (RB) son una herramienta poderosa de representación del conocimiento. Una red bayesiana es un grafo acíclico dirigido (GAD) con una distribución de probabilidad asociada a cada nodo, por lo que en ocasiones se les llama también redes probabilísticas. Los nodos en la red representan las variables, atributos o rasgos del dominio de aplicación, y los arcos entre los nodos representan las relaciones de dependencia entre las variables [1].

Encontrar un modelo de red bayesiana consta de dos partes fundamentales: determinar la parte estructural de la red (o sea, los enlaces entre los nodos que representan las variables) y la parte paramétrica (las tablas de probabilidades asociadas a cada nodo).

La búsqueda de la estructura puede interpretarse como un problema de optimización, se transforma en hallar la red de mejor calidad en el espacio de posibles redes, donde la calidad puede ser medida por una métrica que evalúa la red de acuerdo a los datos de partida. Existen varias métricas que evalúan la calidad de las redes, específicamente con enfoque bayesiano, otros basados en la métrica denominada K2, o bien basados en criterios de información o entropía, o en AIC (Akaike Information Criterion) o MDL (Minimum Description Length), entre otras [2]. En el trabajo se muestran tres algoritmos para la confección de redes probabilísticas, en particular para la determinación automática de la estructura de la red. El primero de ellos propone un método que utiliza la técnica de segmentación estadística CHAID (Chi-Squared Automatic

Revista Avances en Sistemas e Informática, Vol.5 No.2, Junio de 2008, Medellín, ISSN 1657-7663

Interaction Detector, Detector Automático de Interacciones basado en Chi-cuadrado) [3] para construir árboles de decisión con las relaciones más significativas. Luego se hace depender la clase o variable objetivo de los árboles hallados.

El segundo método constituye una modificación del primero. En él se utiliza la técnica CHAID mejorando la complejidad de la estructura, pues se hace depender de la clase las variables más significativas y además se consideran otras relaciones entre las variables predictivas.

En el tercer método se hace uso de la estrategia denominada medidas de ajuste y búsqueda, usando un algoritmo de búsqueda heurística. Dentro de estos han sido objeto de estudio los algoritmos bioinspirados, en particular la Inteligencia de Enjambre (Swarm Intelligence, SI) por su simplicidad y robustez [4-8]. Para la búsqueda de la estructura de la red se utiliza un modelo computacional de SI: la optimización en enjambre de partículas, PSO [8].

El artículo se divide como sigue: la sección 2 muestra el concepto de RB, en la sección 3 se resumen las ideas fundamentales de los algoritmos para el aprendizaje estructural de redes bayesianas y en la sección 4 se muestran los resultados de los algoritmos en la predicción de mutaciones en secuencias de ADN de la proteína transcriptasa inversa del VIH.

II. REDES BAYESIANAS

Una red bayesiana es un par (D,P), donde D es un grafo acíclico dirigido, P= $\{p(x_1|\tau_1), \ldots, p(x_n|\tau_n)\}$ es un conjunto de n distribuciones de probabilidad condicionales, una por cada variable x_i (nodos del grafo), y τi es el conjunto de padres del nodo xi en D. El conjunto P define la distribución de probabilidad conjunta asociada, como muestra la ecuación (1):

$$p(x) = \prod_{i=1}^{n} p(x_i|\tau_i) \qquad x = (x_1, x_2, \ldots, x_n) \qquad (1)$$

Las redes bayesianas son un tipo especial de sistema basado en el conocimiento, por lo que es posible hacer inferencia a partir de conocimiento a priori. A este proceso se le llama propagación de evidencias [1, 3].

Al proceso de obtener una red bayesiana a partir de datos se le denomina aprendizaje de redes bayesianas y típicamente se divide en dos aspectos:

Aprendizaje paramétrico: dada una estructura, obtener las probabilidades a priori y condicionales requeridas.

Aprendizaje estructural: obtener la estructura de la red Bayesiana, es decir, las relaciones de dependencia e independencia entre las variables involucradas.

Las técnicas de aprendizaje estructural dependen del tipo de estructura de red: árboles, poliárboles y redes multiconectadas. Otra alternativa es combinar conocimiento subjetivo del experto con aprendizaje. Para ello se parte de la estructura dada por el experto, la cual se valida y mejora utilizando datos estadísticos.

Los primeros trabajos para obtener redes bayesianas desde datos se obtuvieron por Show y Liu [9] y Rebane y Pearl [10] para obtener árboles y poliárboles.

Otras de las estructuras más utilizadas de redes bayesianas son las arquitecturas TAN: clasificador bayesiano simple aumentado con un árbol, [11] y BAN: clasificador bayesiano simple aumentado con una red, [12]. Estas son generalizaciones del clasificador bayesiano conocido como: clasificador Bayesiano Naive (CBN) [13], que supone independencia entre los atributos dada la clase, y su estructura ya esta dada, por lo que solo se tienen que aprender las probabilidades de los valores de los atributos dada la clase.

En el epígrafe siguiente se proponen tres algoritmos para el aprendizaje estructural de redes bayesianas.

III. MÉTODOS DE APRENDIZAJE ESTRUCTURAL DE REDES BAYESIANAS

En esta sección se formulan tres algoritmos para el aprendizaje estructural de redes bayesianas, los dos primeros se basan en la búsqueda de estructuras de dependencias entre variables y en el tercero se usa una estrategia de medidas de ajuste y búsqueda en el que se utiliza el algoritmo de búsqueda heurístico PSO. Este último algoritmo ha mostrado arrojar buenos resultados, ser muy rápido y barato comparado con otros algoritmos.

A. Método I. Aprendizaje de redes bayesianas usando la técnica CHAID (BayNet)

Para obtener la topología de la red se parte de la obtención de árboles de decisión según la técnica CHAID. Dicha técnica surge inicialmente como un método de segmentación. Es muy útil en todos aquellos problemas en que se quiera subdividir una población a partir de una variable dependiente (variable objetivo o clase), y posibles variables predictoras que cambien los valores de la variable dependiente en cada una de las subpoblaciones o segmentos. Ejemplos típicos asociados con su origen son los problemas de estudio de mercado. En estos casos la variable dependiente puede ser la aceptación o no de un producto y las variables predictoras un conjunto de características de la población que pueden influir en esta aceptación. La técnica de CHAID es capaz en este caso de segmentar la población en grupos de acuerdo con determinados valores de esas variables predictoras y sus interacciones que distinguen de forma óptima en algún sentido, diferencias esenciales en el comportamiento de la variable dependiente [3].

Más que segmentar la población en este caso la técnica de CHAID se usa para:

• Conocer cuáles, entre decenas de variables (o rasgos) pueden ser eliminadas.

• Comprender el orden de importancia de los rasgos. (En las investigaciones epidemiológicas puede utilizarse

para comprender el orden de los factores de riesgo en la caracterización de la enfermedad y en particular para ayudar a detectar posibles factores confusores o modificadores de riesgo. En estudios de secuencias puede aplicarse para conocer las posiciones más importantes para el análisis que se hace, ya sea un tipo de mutación, un posible donor o acceptors en problemas de clasificación de splice site, una nueva interacción de proteínas, etc.).

• Para entender cómo interactúan los rasgos unos con otros (en las investigaciones epidemiológicas para entender cómo ciertos factores de riesgo se relacionan con otros, en estudios de secuencias saber como interactúan estas posiciones).

• Para simplificar numerosas tablas de contingencia combinando categorías de variables predictoras que no difieren significativamente.

La cuarta propiedad del CHAID es una de las cosas más interesantes, pues CHAID combina categorías de una variable predictora que no difieren significativamente y que unidas muestran mayor asociación con la variable dependiente. De esta forma se resuelve por ejemplo el problema de los rangos para la edad (considerada como un posible factor de riesgo) para obtener una tabla de contingencia significativa con la enfermedad. Simplemente si se tienen entre 10 ó 12 categorías, CHAID se ocupará de unir las categorías consecutivas que no difieren significativamente y el resultado final mostrará muchos menos rangos de edades, evidenciando las que constituyen verdadero factor de riesgo. Lo mismo es capaz de hacer con variables incluso nominales (por ejemplo la raza o color de la piel), e incluso con variables que tienen un valor perdido, asociando éste a la categoría de la variable respecto a la cual los casos son más parecidos en su comportamiento.

El CHAID permite obtener un árbol de decisión en forma automática con las características mencionadas. La búsqueda se acota por un conjunto de parámetros que impone el usuario cuyos valores no sólo definirán el espacio de búsqueda sino la posible topología a generar. Cada uno de estos parámetros persigue un objetivo lo que repercute en la red que se obtiene. A continuación se comentan cada uno de estos parámetros:

ChiSquare_Max_Significance: Cota máxima de la probabilidad del estadístico Chi-cuadrado que será aceptado por el método como una posible interacción. Note que el dominio de dicho estadístico son los números reales entre 0 y 1. A medida que este valor se aproxima a 0, el algoritmo es más exigente para declarar una interacción y consecuentemente, se observará una disminución en la cantidad de interacciones aceptadas por el método y disminuirá la cantidad de arcos y de nodos en la red. De esta forma el método no solo constituye un método de aprendizaje automatizado sino que además obtiene una selección de atributos.

MinCountOfInstancesToSplit: Mínima cantidad de casos que debe tener una población para que el método considere su posible subdivisión. Esta es una cota necesaria para lograr cierto nivel de fiabilidad ante los test Chi-cuadrado. Debe tratarse de acuerdo al tamaño de la población, la distribución de la clase y los posibles grados de libertad de las tablas de contingencias aspirantes.

MaxDepth: Cota sobre la máxima cantidad de arcos que pudiese tener el camino más largo dentro de la topología generada. Su mayor influencia se sienta en la complejidad de la red a obtener. Tiene amplia repercusión en el espacio de búsqueda cuando las poblaciones son vastas en individuos y rasgos.

La estructura de la red bayesiana se obtiene con sólo invertir el orden del árbol, sin embargo no se descarta la participación de los expertos pues el algoritmo facilita obtener distintos árboles si se cambian parámetros o se hace interactivamente con el usuario, lo que permite que se tenga en cuenta al experto a la hora de seleccionar la topología más adecuada.

Algoritmo BayNet

Paso 1) Obtener un conjunto de árboles mediante la técnica CHAID con las variables más relacionadas con la clase. (Cada uno de los árboles obtenidos establece un subconjunto de variables relacionadas entre sí y con la clase.)

Paso 2) Formar el modelo estructural de la red bayesiana haciendo depender la clase de cada uno de estos árboles. O sea, se establece un arco dirigido de cada variable más significativa en cada uno de los árboles hacia la variable clase.

La complejidad computacional del algoritmo en el peor caso resulta del orden O(D*M*N*T2) donde D: Número de árboles, M: Número de instancias, tamaño máximo de la población o número de casos, N: Número de atributos o rasgos del problema, T: Cantidad de valores diferentes que pueden tomar los atributos. Si no se considera el conjunto de valores posibles para las variables, pues frecuentemente es 2 en problemas de bioinformática, si las variables son dicotómicas y en otro caso un número natural mayor o igual a 2 pero relativamente pequeño, resulta un orden de complejidad O(D*M*N).

B. Método II. Algoritmo BayesChaid

El algoritmo parte de ideas propias de la técnica de segmentación de CHAID con adaptaciones para la generación de topologías más complejas que se ajusten a Redes Bayesianas.

El algoritmo BayesCHAID se basa como su nombre lo indica, en ideas de la técnica de CHAID. La adaptación del algoritmo consiste en que hace una búsqueda de las interacciones entre variables no mediante árboles de decisión, sino que busca a lo ancho y en profundidad en el árbol de interacciones posibles. La búsqueda se acota por un conjunto de parámetros que impone el usuario explicados previamente en el epígrafe 3.1, y en este caso se incluye un nuevo parámetro, MaxNrOfParents que es cantidad de padres que podrán tener los nodos de la red a generar. Esto influye de forma espacial en las tablas de probabilidades condicionales generadas para la red. El valor 15 puede ser una cota máxima a considerar para las arquitecturas de 32 bits.

El método se apoya en dos estructuras de datos que son fundamentales en su desempeño:

Revista Avances en Sistemas e Informática, Vol.5 No.2, Junio de 2008, Medellín, ISSN 1657-7663

• Un listado de las sub-poblaciones generadas por las interacciones de Chi-cuadrado aceptadas por el método. Las subpoblaciones no solo deben especificar un conjunto de individuos sino también el nodo que le dio origen al añadírsele un padre.

• Una matriz para guardar la estructura cuando se forma la red.

El algoritmo por pasos se resume a continuación.

Algoritmo BayesChaid

Paso 1) Inicialización

1a. Inicializar la red como un grafo vacío.

1b. Inicializar la lista con toda la población y como origen la variable clase.

1c. Inicializar un cursor para recorrer la lista

1d. Guardar la población del cursor en P

1e. Guardar el nodo origen de P en B

Paso 2) Aplicar la prueba Chi-cuadrado y determinar las variables significativas para todas las variables considerando la población actual P.

Paso 3) Guardar en A la variable más significativa (la que menor valor de significación tiene acorde a la prueba Chi-cuadrado). Si la significación de esta variable no es menor que el nivel de significación establecido ir al paso 6.

Paso 4) Si el arco A-B no forma ciclo o incumple la restricción de profundidad y A no sobrepasa la cantidad de padres prefijada ir a 5.

Generar la sub-poblaciones según los valores de la variable que esta en A desde todas las sub-poblaciones de la lista cuyo origen sea B y añadirlas a la lista con A como origen si cumple la restricción de cantidad mínima de casos en las sub-poblaciones que se obtienen. Añadir arco A-B a la red.

Paso 5) Redefinir significación de A como uno e ir al paso 3.

Paso 6) Mover el cursor en la lista, si no se desborda la lista ir a 1d.

Paso 7) Devolver la red.

Del análisis de complejidad del algoritmo resulta una cota superior de O(L2*M*n*D), donde se toman en consideración las siguientes variables: n es cantidad de rasgos del problema, M es el tamaño máximo de la población o cantidad máxima de casos, D es una cota máxima de caminos para la red. En este caso no se considera el conjunto de valores posibles para las variables como se explicó en el epígrafe A.

Se añade la variable L para simplificar el proceso de análisis, L es longitud máxima de la lista. Esta longitud es siempre menor al número de combinaciones con permutación de D elementos en un conjunto de N, pues no se admiten todas las combinaciones de arcos (hay que eliminar las que forman ciclos). Así se toma la ecuación 2:

$$L = \frac{n!}{(n-D)!} \qquad (2)$$

Considerando que D es suficientemente pequeño, se tiene que L< nD.

Se pueden identificar dos ciclos anidados entre los pasos 1-6 y 3-5 cuya cantidad máxima de iteraciones corresponden a L y n respectivamente. Un análisis detallado por puntos arroja una complejidad según ecuación 3:

$$O(L^2 * M * n * D) \qquad (3)$$

C. PSO principios básicos

PSO es una metaheurística de optimización estocástica basada en una población. Un enjambre se define como una colección estructurada de organismos (agentes) que interactúan. La inteligencia no está en los individuos sino en la acción de todo el colectivo. Cada organismo (partícula) se trata como un punto en un espacio N dimensional el cual ajusta su propio "vuelo" de acuerdo a su propia experiencia y la experiencia del resto de la banda. La banda (swarm) "vuela" por el espacio de búsqueda localizando regiones o partículas prometedoras [7].

1) Método III. PSO en el aprendizaje de redes bayesianas

La búsqueda de la estructura de la red puede formularse como un problema de optimización en el espacio de las posibles redes Ω, en otras palabras, determinar $X_{ópt} \in \Omega$, $H(_{Xópt}) \geq H(X_i)$, $\forall\ X_i \in \Omega$, donde la función objetivo H considerada es una métrica de calidad de las descritas en el capitulo 4 de la tesis de Bouckaert [2] para el caso de búsqueda local, o medidas que miden la exactitud en el caso de una búsqueda global cuando se trabajan con validaciones cruzadas de los datos según implementaciones en el ambiente Weka (Waikato Environment for Knowledge Analysis) [14], y otras medidas propuestas en la literatura en el caso de conjuntos de datos con clases desbalanceadas [15].

En la modelación de nuestro problema de búsqueda a partir del algoritmo PSO se define cada partícula como una red bayesiana la cual se representa como una matriz de adyacencias $B=b_{ij}$ donde b_{ij} =1 si el atributo i es padre del atributo j, (si existe un arco de i a j) y b_{ij} =0 en otro caso. Se puede pensar que el espacio de búsqueda Ω tiene cardinal 2^{n^2} ; de hecho se puede trabajar con dicho espacio, pero habría que chequear que no existan ciclos. Esto se puede lograr, por ejemplo, eliminando de forma aleatoria arcos que formen parte de ciclos existentes [16]. Se propone entonces una forma de generar el espacio de búsqueda garantizando que no existan ciclos, o sea, partiendo de que un grafo dirigido representa un ordenamiento topológico, si y solo sí este no presenta ciclos, es posible a partir de una permutación formar un grafo acíclico dirigido [16].

2) Algoritmo PSO binario

El problema de optimización que se propone es binario, por lo que el algoritmo PSO original [17],[18] debe ser adaptado.

X_i es una partícula (matriz del espacio Λ), $\{X_1, X_2, \ldots\}$ es una bandada (conjunto de partículas), $\{V_1, V_2, \ldots\}$ son velocidades (matrices del espacio Ω asociadas a cada partícula que indican

su movimiento), $\{X_{pBest1}, X_{pBest2}, \ldots\}$ son los mejores puntos del espacio localizados por cada partícula, X_{gBest} es el mejor punto localizado por la bandada.

```
Algoritmo
Inicializar valores;
t=0;
Repeat
   Generar red acíclica Gπ
   For each i=1,..., cantPart
      Calcular Vi (t+1) y limitarla a  [-Vmax, +Vmax]
      Calcular S(Vi);          Actualizar Xi;
     For all j, k:  If rand ( ) < S(Vijk)
        then Xijk (t+1)=       (Gπ) jk  else Xijk (t+1)= 0
   endFor each
For each i=1,..., cantPart
       Evaluar Xi    //Aplicar la métrica
      Actualizar XpBest i
   endFor each
   Actualizar XgBest
 Incrementar t
Until  t > CantGeneraciones
```

donde Inicializar valores asigna aleatoriamente valores a la población de X_i , V_i y $XpBest_i$ de cada partícula como copia de X_i y X_{gBest} con el mejor valor. CantGeneraciones es la cantidad de "generaciones" que van a interactuar las partículas y cantPart es la cantidad de partículas que van a existir en cada generación. La variable t se utiliza como contador de generaciones. La subrutina Generar red acíclica G_π genera una red acíclica como se vio en [16] a partir de una permutación aleatoria π de (1, 2,…, n) con distribución uniforme, dicha red se representa como una matriz de adyacencia. La actualización de las partículas se logran añadiendo la velocidad a cada partícula obtenida en la iteración t, la velocidad se obtiene utilizando las expresiones que se muestran en la ecuación 5.

$$V_i = wV_i + c_1\, rand\,(XpBest_i - X_i) + c_2\, rand(XgBest - X_i)$$

$$S(V_{ijk}) = \frac{1}{1 + e^{(-V_{ijk})}} \qquad (4)$$

En la expresión de V_i, el primer termino es la memoria de la partícula, el segundo la parte cognitiva o conocimiento privado, el tercero la parte social que permite la colaboración, w es el peso de inercia, c_1 y c_2 son los llamados factores de aprendizaje cognitivo y social respectivamente y rand es un número aleatorio entre 0 y 1. La selección de estos parámetros tiene impacto en la velocidad de convergencia y la velocidad del algoritmo para encontrar el óptimo. Entre los valores recomendados en [16], se tomaron los valores $c_1 = c_2 = 2$, pero en realidad se recomienda en el trabajo que c_1 y c_2 no tomen necesariamente el mismo valor sino, que se generen aleatoriamente con distribución en el intervalo [0,2], en [15] se recomienda que su suma sea mayor o igual a 4, w = 0.5 + rand /2. Otros valores para la transformación, S: □/[0,1] aparecen en [17], [18], [19].

La complejidad computacional de este algoritmo resulta del oden O(MaxGen*P*max (N2, O(métrica)), donde Margen: Número de iteraciones P: Cantidad de partículas y N: Número de rasgos, además se debe conocer el orden de la métrica que mide la calidad de la red que se obtiene.

La implementación de los tres métodos se realizó haciendo extensiones a la plataforma Weka [14] con el objetivo de minimizar el tiempo de implementación y rehusar código libre. El algoritmo de búsqueda heurística PSO permite seleccionar distintas funciones objetivos o fitness: la basada en validaciones cruzadas o la basada en el método LOO-CV, ambas ya implementadas en Weka. Alternativamente se puedieran utilizar otras funciones, como la que tiene en cuenta la dependencia de los atributos con la clase basada en conjuntos aproximados y propuesta por [17], y dos medidas utilizadas en [20] cuando las bases de datos se encuentran desbalanceadas, pero que a su vez se consideran medidas de calidad robustas para clasificación. [21]. El primer método inicialmente se aplicó utilizando paquetes matemáticos y estadísticos profesionales, y actualmente se cuenta con una implementación en lenguaje Borland Delphi, el software del mismo nombre ByNet [22]. Para hacer inferencias bayesianas se utiliza el software JavaBayes [23] el que también se incorporó como una extensión a Weka.

IV. APLICACIÓN DE LOS MÉTODOS PROPUESTOS PARA PREDECIR MUTACIONES DE PROTEÍNAS DEL VIH

Uno de los virus que más afecta a la humanidad es el virus de inmunodeficiencia humano (VIH). Cada año causa más de 3 millones de muertes. El VIH se considera un virus con una altísima capacidad de mutación en las proteínas que lo componen. La mayoría de los fármacos antiretrovirales aprobados para el tratamiento del VIH tratan de inhibir dos de las proteínas más importantes: la proteasa y la transcriptasa inversa, pero el fenómeno de resistencia a estos fármacos está asociado a la capacidad de mutación, pues los cambios de aminoácidos alteran la estructura de las enzimas de tal manera que el fármaco no puede inhibir su función. Partiendo de este problema, este trabajo se enfoca en el estudio de secuencias de mutaciones de la proteína transcriptasa inversa del VIH. El objetivo es probar los métodos propuestos en la obtención de redes bayesianas que permitan predecir mutaciones de esta proteína, o sea, utilizar las redes bayesianas para clasificar tipos de mutaciones, o predecir mutaciones en cualquier parte de la secuencia perteneciente a un nuevo caso.

A. Resumen de la base de datos de las secuencias de ADN de la proteína trascriptasa inversa

Se utilizan secuencias de ADN obtenidas en la base de datos (BD) de la Universidad de Stanford, la cual tiene 419 secuencias de mutaciones de esta proteína con 603 pares de base cada una. Esta BD esta disponible online[1].

Para obtener los valores de cada atributo, se utiliza la lista de mutaciones en conjunción con la secuencia de referencia

1 http://hivdb.stanford.edu/cgi-bin/PIResiNote.cgi

HXB2 (número de acceso a GenBank K03455). Las secuencias son todas del mismo tamaño, y no hay valores desconocidos [24], [25], [26].

Las secuencias resultantes se representan tomando como referencia los resultados del grupo álgebra del Genoma [27], [28], [29], [30], o sea, G↔00; A↔01; T↔10; C↔11, obteniéndose 1206 variables predictivas y la clase (dos grupos de mutaciones obtenidos empíricamente por métodos clásicos de detección de clusters [31].

1) Resultados

En la Figura 1 se muestran las redes obtenidas con el software BayNet para tres niveles de profundidad en los árboles, nivel de significación del 5%, y cantidad de casos en las subpoblaciones igual a 30 (a), 100 (b) y 200 (b). Se puede apreciar que las posiciones más significativas se mantienen en todos los casos. Estas son 20 (segundo número binario del codón cuatro), 24 (último número binario del codón cuatro), 474 (último número binario del codón setenta y nueve) y 1023 (tercer número binario del codón 170). Cuando las subpoblaciones son de 100 y 200 casos se tiene una variable menos en la red (la variable V1053).

Cada rama en la red constituye uno de los árboles creados y convertidos en nodos de la red, pues aunque en la representación no se representa la red en forma de grafo las flechas van en la dirección de la clase en cada caso. Las redes pueden utilizarse para inferir la clase (un tipo de mutación de los dos prefijados previamente) o inferir una de las posiciones en la secuencia ante una nueva mutación (nuevo caso). Los resultados de las redes pueden ser exportados en el formato XMLBIF (eXtensible Markup Language Interchange Format for Bayesian Networks[2]) . Este permite importar las redes desde otros paquetes de software para hacer inferencias bayesianas y poder validar los resultados de las redes obtenidas. Las redes exportadas desde el sofware ByNet se importan en JavaBayes en formato XMLBIF, y se pueden usar para realizar inferencias tanto de la clase como de posiciones en la secuencia del nuevo caso. Por ejemplo, si la evidencia es que la variable V20 toma valor 0, se puede inferir la clase 2 con probabilidad 0.96, de otro modo si la clase es la 2, se pueden inferir posiciones con determinada probabilidad, es decir V20=1 con probabilidad 0.99, V24=0 con probabilidad 0.93, V1023=1 con probabilidad 0.99, V474 =0 con probabilidad 0.92, de este modo se puede utilizar la red como clasificador o para inferencia en cualquier dirección ante determinadas evidencias.

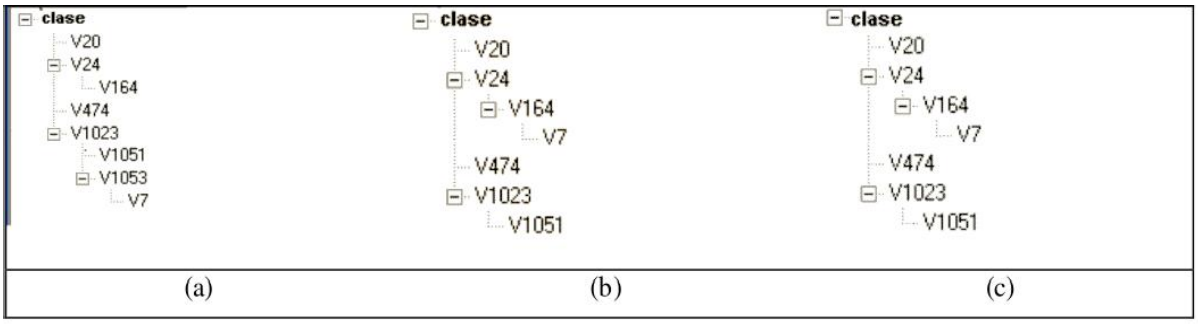

Figura 1. Redes bayesianas obtenidas con el método Baynet. Los parámetros en este caso son nivel de profundidad 3, nivel de significación 5%, y número de casos mínimo en las subpoblaciones (a) 30, (b) 100 y (c) 200 respectivamente.

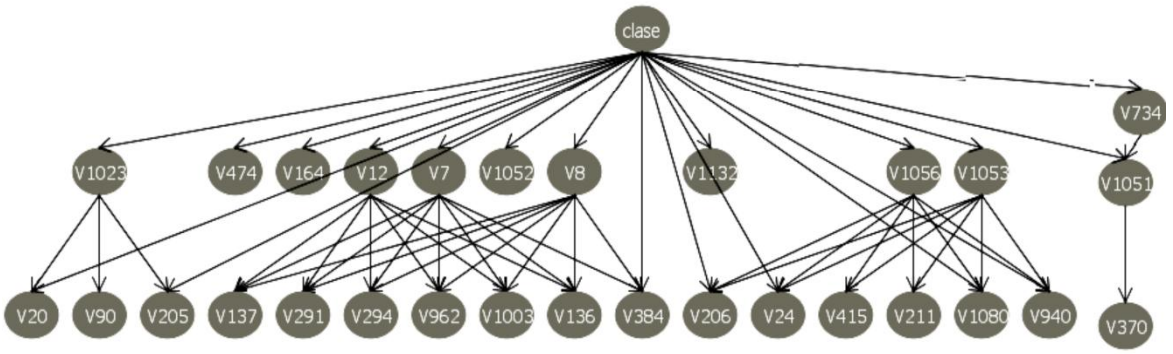

Figura 2. Red byesiana obtenida con el método BayesChaid, cuando el número de casos mínimo permisible en las subpoblaciones es 30. Para este método los resultados que validan los modelos son similares, a los que se muestran en la tabla 1, para los tamaños de subpoblaciones fijados (30, 100, 200). En el modelo se pueden hacer inferencias con una cantidad mayor de variables.

2 http://www.cs.cmu.edu/~fgcozman/Research/InterchangeFormat/

El problema del método consiste en que se debe limitar el número de árboles a obtener pues si se obtienen todos los árboles de las variables más significativas, se hace depender la clase de muchas variables y se pueden cometer el error de sobreajuste u overfitting. Además el aprendizaje paramétrico se ve afectado pues no existe un cubrimiento completo del dominio para todas las combinaciones posibles de las variables. Este problema se presenta mucho en las bases de datos de Bioinformática, sin embargo el método ha sido aplicado con resultados satisfactorios en problemas de diagnóstico médico, pues se ha contado con datos que no presentan esta dificultad [32], [33], [34] [35], [27], además de que en el dominio médico se cuenta con expertos capaces de validar la probabilidad con que se puede tener determinado resultado.

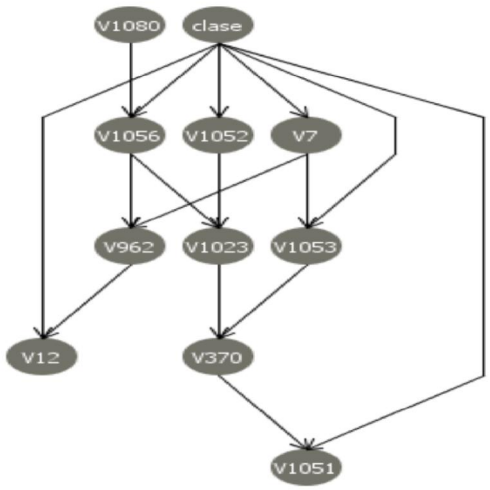

Figura 3. Red bayesiana obtenida con el método III basado en la búsqueda PSO, con tres padres, 40 partículas, 100 iteraciones y previa selección de atributos con el propio método.

Los métodos dos y tres se aplicaron como clasificadores, pues ellos se añadieron como una extensión a Weka como parte de la clase bayesnet con el objetivo de aprovechar las facilidades para la validación ya implementadas, pero se puede obtener con este software redes bayesianas representadas en formato XMLBIF y utilizarlas para la inferencia de posiciones en la secuencia con el JavaBayes de la misma forma que se hizo en el método I.

En la Figura 2 se muestra la red obtenida con el método uno, BayesChaid y en la Figura 3 una red obtenida con el método tres. En la tabla 1 se publican los resultados de la clasificación para las redes de las figuras 2 y 3, respectivamente.

Tabla 1. Resultados de la clasificación con los métodos 1 y 2

Medida	Algoritmo K2	Algoritmo TAN	Búsqueda de independencias condicionales
Exactitud	98.56	98.56	98.09
TP	0.982	0.982	0.97
TN	0.988	0.988	0.988
AUC	0.999	0.998	0.997
Precisión	0.987	0.988	0.982

Como se puede observar en la tabla I los resultados para el algoritmo ByNet como clasificador son peores que para los otros dos algoritmos que se proponen. Los resultados de los algoritmos BayesChaid y PSO son comparables con otros algoritmos para esta tarea considerando para todas las medidas clásicas para evaluar la clasificación, el valor del algoritmo ByNet está en la necesidad en algunos casos de tener en cuenta relaciones entre subgrupos de posiciones en las secuencias.

Tabla 2. Resultados de la clasificación con los algoritmos K2, TAN y basado en pruebas de independencia condicional

Medida	Algoritmo K2	Algoritmo TAN	Búsqueda de independencias condicionales
Exactitud	98.56	98.56	98.09
TP	0.982	0.982	0.97
TN	0.988	0.988	0.988
AUC	0.999	0.998	0.997
Precisión	0.987	0.988	0.982

Si se comparan los resultados de las tablas 1 y 2 se aprecia que los resultados de otros algoritmos clásicos para el aprendizaje de redes bayesianas son similares a los que muestran nuestros algoritmos. Los métodos que se proponen para el aprendizaje estructural de redes bayesianas se están aplicando a otros problemas de Bioinformatica [35], [27, 36], [37], [38], [39] y los resultados son satisfactorios. También se han aplicado a problemas médicos, como por ejemplo al diagnóstico de la Hipertensión arterial con buenos resultados [36], [40], [41].

V. CONCLUSIONES

En el trabajo se proponen tres nuevos métodos de aprendizaje estructural de redes bayesianas. Los algoritmos basados en Chi-cuadrado presentan como ventaja que poseen criterios estadísticos (significación del Chi-cuadrado) que establecen una semántica clara en la determinación de la estructura de la red (sustituyendo criterios de umbrales definidos a priori). El segundo algoritmo en particular, utiliza criterios similares que incorporan la posible interrelación entre las variables. Los resultados que se obtienen en un problema de clasificación son comparables finalmente a los reportados por otros clasificadores bayesianos; pero el perfeccionamiento de la estructura facilita la interpretabilidad de los mismos. El tercer algoritmo, por su parte presenta como ventaja que facilita la búsqueda a través de todo el espacio de redes posibles, utilizando técnicas aleatorias como no pueden hacerlo determinísticamente las técnicas clásicas por ser un problema NP. Los nuevos algoritmos se han probado en un problema de predicción de tipos de mutaciones en la secuencia de la transcriptasa inversa del VIH. Se confirma que los resultados de la clasificación con estos modelos son satisfactorios y se mostró además como aplicar los modelos obtenidos para la inferencia de posiciones mutadas de la proteína ante un nuevo caso.

REFERENCIAS

[1] Castillo, E., Gutiérrez, J. M., and Hadi, A. S., Expert Systems and Probabilistic Network Models. 1997: Springer-Verlag, New York.

[2] Bouckaert, R.R., Bayesian Belief Networks: From Construction to Inference. 1995, Utrecht UniversityUtrecht University.

[3] CHAID, W., CHAID para SPSS sobre Windows. Técnicas de segmentación basadas en razones de verosimilitud Chi-cuadrado. User Manual. SPSS Inc., 1994.

[4] Kennedy, J., Eberhart, R.C., Particle swarm optimization. In: Proceedings of IEEE International Conference on Neural Networks, Perth, 1995: p. 1942–1948.

[5] Kennedy, J., Eberhart, R.C., A new optimizer using particle swarm theory. In: Sixth International Symposium on Micro Machine and Human Science. Nagoya, 1995: p. 39–43.

[6] Kennedy, J., The particle swarm: social adaptation of knowledge. In: IEEE International Conference on Evolutionary Computation, April 13–16, 1997: p. 303–308.

[7] Kennedy, J., Spears, W. M., Matching algorithms to problems: an experimental test of the particle swarm and some genetic algorithms on the multimodal problem generator. Proceedings of the IEEE International Conference on Evolutionary Computation, 1998: p. 39- 43.

[8] Kennedy, J., Swarm Intelligence. Morgan Kaufmann Publishers, 2001.

[9] Chow, C., Liu, C., Approximating discrete probability distribution with dependence trees. IEEE Transactions on Information Theory, 1968. 114: p. 462- 467.

[10] Rebane, G., Pearl, J., The recovery of causal poly- trees from statistical data. Int. J. Approxi. Reasoning, 1988. 2 (3): p. 341.

[11] Friedman, N., Goldszmidt, M., Building Classifiers using Bayesian Networks. In Proceedings of thirteen National Conference on Artificial Intelligence, 1996.

[12] Larrañaga, P., Aprendizaje automatico de Modelos Graficos II. Aplicaciones a la Clasificación Supervisada. 2000.

[13] Duda, R.O., Hart, P.E., Pattern Classification and scene analysis. Jonh Wiley Sons, 1973.

[14] Witten, I.H., Frank, E., Data Mining Practical Machine Learning Tools and Techniques. segunda ed. 2005: Morgan Kaufman.

[15] Beielstein, T., Parsopoulos K. E., Vrahatis M.N. , Tuning PSO parameters through sensitivity analysis. Technical Report of the Collaborative Research Center, University of Dortmund, 2002.

[16] Chávez, M.C., Silveira, P., Casas, G., Perz, B., Grau, Aprendizaje estructural de Redes Bayesianas utilizando PSO. COMPUMAT-07, 2007.

[17] Wang, X., Yang J., Teng X., Xia W., Jensen R, Feature Selection Based on Rough Sets and Particle Swarm Optimization. Pattern Recognition Letter, Elsevier, 2006.

[18] Ferat, S., Yavuz, M. C.,, Arnavut, Z., Uluyol, Ö., Fault diagnosis for airplane engines using Bayesian networks and distributed particle swarm optimization. Parallel Computing , Elsevier

[19] 2007. 33: p. 124–143.

[20] Mahamesd, G.H.O., Andries P.E., Ayed S., Dynamic Clustering using PSO with Application in Unsupervised Image Classification. Transactions on Engineering, computing and Technology, 2005. 9.

[21] Eitrich, T., Kless, A. , Druska, C., Meyer, W., Grotendorst, J. , Classification of Highly Unbalanced CYP450 Data of Drugs Using Cost Sensitive Machine Learning Techniques. J. Chem. Inf. Model., 2007. 47: p. 92-103.

[22] Fawcett, T., ROC Graphs: Notes and Practical Considerations for Researchers. SiteSeer.IST, siteseer.ist.psu/646695.html, 2004.

[23] Rodríguez, A., Mondeja, Y., Díaz, Y., Herramienta computacional para hacer inferencias Bayesianas, aplicaciones a Bioinformática Trabajo de Diploma, 2006.

[24] Gagliardi, F., Java Bayes Versión 0.346 Bayesian Network in Java. User Manual. 2001.

[25] Hunter, L., Artificial Intelligence and Molecular Biology. p. 500.

[26] Murray, R.J., Predicting Human Immunodeficiency Virus Type 1 Drug Resistance From Genotype Using Machine Learning. Master of Science School of Informatics. University of Edinburgh, 2004.

[27] Mellors, J.W., Brendan, A. L., Schinazi, R. F., Mutations in HIV-1 Reverse Transcriptase and Protease Associated with Drug Resistance.

[28] Grau, R., Galpert, D., Chávez, M., Sánchez, R., Casas, G., Morgado, E., Algunas aplicaciones de la estructura booleana del Código Genético. Revista Cubana de Ciencias Informáticas, 2005. 1.

[29] Sánchez, R., Morgado, E., Grau, R., A genetic code boolean structure I. The meaning of boolean deductions. Bulletin of Mathematical Biology 2005. 67: p. 1–14.

[30] Sánchez, R., Morgado, E., Grau, R., The genetic code boolean lattice. MATCH Communications in Mathematical and in Computer Chemistry, 2004. 52 p. 29-46.

[31] Sánchez, R., Morgado, E., Grau, R., Genetic Code Boolean Algebras. WSEAS Transactions on Biology and Biomedicine 2004. 1(2): p. 190-197.

[32] Berkhin, P., Survey of Clustering Data Mining Techniques. Academic Press, 2002: p. 15-18.

[33] Chávez, M.C., Grau, R., García, M.M., Sistemas de Inferencia Estadística. Tesis de Maestria, 1996.

[34] Chávez, M.C., Grau , R., García , M., Un método para construir Redes Bayesianas. Revista de Ingeniería de la Universidad de Antioquia, 1999.

[35] Chávez, M., Grau, R., Red Bayesiana de pronóstico de trastornos neuropsiquicos leves. Informática 2000.

[36] Chávez, M.C., Grau , R., Sánchez, R., Construcción de árboles filogenéticos a partir de secuencias de ADN y su integración en una red bayesiana, in Informática 2005.

[37] Chávez, M.C., Casas, G., Martínez, N., Grau, R., Red bayesiana a partir de factores de riesgo de la Hipertensión Arterial. III Simposio Internacional de Hipertensión Arterial, 2006.

[38] Chávez, M.C., Casas, G., González, E., Grau, R., BYNET Herramienta computacional para aprendizaje e inferencias de redes bayesianas en aplicaciones Bioinformáticas. Memorias de Informática, 2007.

[39] Chávez, M.C., Silveira, P.,Casas, G., Grau, R., Bello, R., Aprendizaje estructural de redes bayesianas utilizando PSO. Memorias de Compumat, 2007.

[40] Chávez. M.C., C., G., Moya, I., Grau, R., A new Method for Learning Bayesian Networks. Application to Data Splice site Classification. Second Workshop on Bioinformatics Cuba – Flanders, 2008.

[41] Chávez, M.C., Casas, G., González, E., Grau, R., Uso de las redes bayesianas combinado con técnicas estadísticas para el diagnostico de la Hipertensión arterial. Memorias de Convención Internacional de Ing. Eléctrica, CIE, 2007.

[42] Chávez, M.C., Casas, G., Moreira, J., González, E., Bello, R., Grau, R., Uso de redes bayesianas obtenidas mediante Optimización de Enjambre de Partículas para el diagnóstico de la Hipertensión Arterial. Octavo Congreso Internacional de Investigación de Operaciones, 2008.

María del Carmen Chávez. Nació en Placetas, en 1962. Se graduó de Licenciatura en Cibernética Matemática en la Universidad Central de Las Villas en el año 1985. Trabaja como profesora Auxiliar del Departamento Ciencia de la Computación en la Facultad de Matemática Física y Computación. En 1999 defendió su tesis de Maestría en Computación Aplicada. Su grupo de investigación actual es el de Bioinformática. Sus intereses científicos actuales están relacionados con la aplicación de la estadística y las técnicas de inteligencia artificial a problemas de medicina y bioinformática.

Gladys Casas. Nació en Santa Clara, en 1971. Se graduó de Licenciatura en Cibernética Matemática en 1994 en la Universidad Central de Las Villas. Trabaja como Profesora Auxiliar en la facultad de Matemática, Física y Computación, específicamente en el Centro de Estudios de Informática. Terminó sus estudios de maestría en Matemática Aplicada en 1998 y obtuvo el título de Doctora en Ciencias Técnicas en el año 2004 desarrollando aplicaciones sobre técnicas de detección de conglomerados en epidemiología. Sus intereses científicos actuales están relacionados con la aplicación de la estadística y las técnicas de inteligencia artificial a problemas de medicina y bioinformática.

Ricardo Grau. Nació en Remedios, en 1950. Se graduó de Licenciatura en Ciencias Matemáticas en la Universidad Central de Las Villas en el año 1971. Trabaja como Profesor Titular en la Facultad de Matemática Física y Computación, particularmente en el Centro de Estudios de Informática. En 1987 defendió su Tesis de Doctorado en Ciencias Matemáticas. Sus intereses científico fundamentales están relacionados con la aplicación de las matemáticas, la estadística y la inteligencia artificial a problemas de medicina y bioinformática. Un trabajo del Grupo de Bioinformática que dirige recibió en el año 2005 un Premio Nacional de la ACC. Ricardo Grau ostenta la Orden Carlos J. Finlay

Universidad Nacional de Colombia Sede Medellín
Facultad de Minas

Escuela de Ingeniería de Sistemas

Grupos de Investigación

Grupo de Investigación en Sistemas e Informática

Categoría A de Excelencia Colciencias
2004 - 2006 y 2000.

GIDIA: Grupo de Investigación y Desarrollo en Inteligencia Artificial

Categoría A de Excelencia Colciencias
2006 – 2009.

Grupo de Ingeniería de Software

Categoría C Colciencias 2006.

Grupo de Finanzas Computacionales

Categoría C Colciencias 2006.

Centro de Excelencia en Complejidad

Colciencias 2006

Escuela de Ingeniería de Sistemas
Dirección Postal:
Carrera 80 No. 65 - 223 Bloque M8A
Facultad de Minas. Medellín - Colombia
Tel: (574) 4255350 Fax: (574) 4255365
Email: esistema@unalmed.edu.co
http://pisis.unalmed.edu.co/