Una propuesta genetica y bayesiana para resolver problemas de clasificación en aplicaciones médicas

Alfredo Reyes M., Abraham Sánchez L., Enrique Mote R.

Benemérita Universidad Autónoma de Puebla,
Computer Science Department,

reyes-fred@hotmail.com, asanchez@cs.buap.mx,
mote.enrique.iti@gmail.com

Resumen. Diversos problemas del mundo real pueden ser modelados a través de las redes bayesianas. Este trabajo presenta el desarrollo de una red bayesiana clasificadora para aplicaciones médicas, dicha propuesta es un algoritmo hibrido combinado con algoritmos genéticos. En este artículo utilizamos una base de datos de 267 conjuntos de imágenes SPECT descargadas de SPECT Heart Data Set. Estos datos se procesan, utilizando dos algoritmos: K2 y nuestra propuesta genética-bayesiana. Una vez hecho esto, se determina si el paciente es catalogado como normal o anormal. Finalmente estos resultados son corroborados con el uso del software Netica para constatar el porcentaje de error que estos presentan y como se comportan cada uno de los algoritmos.

Palabras clave: Aplicaciones médicas, redes bayesianas.

1. Introducción

En [1] se presenta una propuesta para la evaluación de las redes bayesianas a partir de datos médicos. Los autores utilizan una cantidad importante de algoritmos de clasificación propuestos en la literatura, incluidos los de naturaleza bayesiana. Según los autores, los resultados obtenidos dependen tanto del proceso de clasificación de las redes bayesianas, como de la experiencia del médico, quien es el que lleva a cabo el proceso de captura e interpretación de los padecimientos presentados.

En [2], los autores proponen la aplicación de redes bayesianas en el modelado de sistemas expertos de triaje en los servicios de urgencias médicas. La propuesta del trabajo se realiza mediante la clasificación de datos provenientes de experiencias de triaje y de la opinión de médicos expertos.

El corazón humano es un sistema complejo ya que que brinda una diversa cantidad de características para el análisis de su estado y funcionamiento. Dentro de un diagnóstico de tomografías computarizadas de la emisión de protones únicos cardiacos (SPECT), algunos radionucleidos emisores de gamma se inyectan primero en el torrente sanguíneo de un paciente. Posteriormente, una

cámara gamma plana se hace girar alrededor del paciente para adquirir múltiples proyecciones 2D. La distribución 3D de una fuente de radionucleidos se puede construir a partir de las proyecciones 2D. Uno de los principales inconvenientes de la SPECT es su tiempo de formación de imágenes dinámicas. Una red bayesiana permite la obtención de una red clasificadora de nodos [3]. Es posible generar estos nodos a partir de los atributos que posee un data set, el cual es resultado del registro y tratamiento de datos correspondientes a mediciones tomadas de tomografías. Los resultados de la generación de una red bayesiana a partir de un algoritmo son muy diferentes, estos dependen de la clasificación y del ordenamiento que se lleva a cabo. Las redes bayesianas en este trabajo nos ayudarán a determinar si un paciente es normal o anormal, siempre y cuando cumpla cierto porcentaje en cada una de las características.

La generación de éstas redes y el procesamiento que tienen una vez generadas es diferente, puesto que la clasificación de nodos, el orden y la conexión entre ellos es diferente. A lo largo de este trabajo se utilizan dos algoritmos principales, el algoritmo K2 y un algoritmo genético hibrido, con estas dos propuestas se obtendrá una red bayesiana [4]. Además se utiliza el software Netica para observar gráficamente la red que se construye a partir de los datos. El desarrollo de estos algoritmos fue realizado en Matlab. Se presentan en la sección de resultados, los experimentos realizados que comprueban la efectividad de nuestra propuesta.

2. Redes bayesianas

Una red bayesiana, o red de creencia, es un modelo probabilístico multivariado que relaciona un conjunto de variables aleatorias mediante un grafo dirigido que indica explícitamente una influencia causal. Gracias a su motor de actualización de probabilidades, el Teorema de Bayes, las redes bayesianas son una herramienta extremadamente útil en la estimación de probabilidades ante nuevas evidencias. Una red bayesiana es un tipo de red causal.

Las redes bayesianas son un formalismo basado en la teoría de probabilidades y los grafos.

Una red bayesiana está definida por:

- un grafo orientado sin circuito $G = (V, E)$, donde V es el conjunto de nodos de G y E el conjunto de los arcos de G,
- un espacio de probabilidad finito (Ω, Z, p)
- un conjunto de variables aleatorias asociadas a los nodos del grafo y definido en (Ω, Z, p), tal que:

$$p(V_1, V_2, ..., V_n) = \prod_{i=1}^{n} p(V_i | C(V_i)) \tag{1}$$

donde $C(V_i)$ es el conjunto de las causas (padres) de V_i en el grafo G.

Una red bayesiana es por lo tanto un grafo causal que ha sido asociado con un representación probabilística subyacente. Como sabemos, esta representación

permite proporcionar un caracter cuantitativo a los razonamientos sobre la causalidad que se pueden hacer en el grafo.

Normalmente las redes bayesianas consideran variables discretas o nominales, por lo que si no lo son, hay que discretizarlas antes de construir el modelo. Los métodos de discretización se dividen en dos tipos principales (i) no supervisados y (ii) supervisados. Los métodos no supervisados no consideran la variables clase, así que los atributos continuos son discretizados independientemente. El método más simple es dividir el rango de valores de cada atributo, $[Xmin, Xmax]$, en k intervalos, donde k está dado por el usuario o se obtiene usando una cierta medida de información sobre los valores de los atributos. Los métodos supervisados consideran la variables clase, es decir los puntos de división para formar rangos en cada atributo y estos son seleccionados en función del valor de la clase. El problema de encontrar el número óptimo de intervalos y de los límites correspondientes se puede considerar como un problema de búsqueda. Es decir, podemos generar todos los puntos posibles de división para formar intervalos sobre la gama de valores de cada atributo, y estimamos el error de clasificación para cada partición posible [5].

3. Procesamiento de señales

La base de datos de 267 conjuntos de imágenes SPECT se procesa para extraer características que resumen las imágenes SPECT originales. Estos conjuntos de datos fueron descargados de SPECT Heart Data Set (disponible en https://archive.ics.uci.edu).

Este conjunto de datos nos describe el diagnóstico de tomografías computarizadas de la emisión de protones únicos cardiacos. Cada uno de los pacientes se clasifican en dos categorías: normales y anormales. Como resultado, se han creado 44 patrones de operación continuos para cada paciente. El patrón se procesa adicionalmente para obtener 22 patrones de funciones binarias. El algoritmo clip3 se utiliza para generar reglas de clasificación de estos patrones [6].

- Número de instancias: 267
- Número de atributos: 23 (22 binarios + 1 clasificación binaria)

El repositorio pone a nuestra disposición 3 archivos los cuales contienen una distribución de clases distintas.

Datos enteros

Clase	Número de ejemplo
0	55
1	212

Datos entrenados

Clase	Número de ejemplo
0	40
1	40

Datos de prueba

Clase	Número de ejemplo
0	15
1	172

4. Algoritmos

4.1. Algoritmo K2

El algoritmo K2 está basado en la optimización de una medida. Esa medida se usa para explorar, mediante un algoritmo de ascenso de colinas, el espacio de búsqueda formado por todas las redes que contienen las variables de la base de datos. Se parte de una red inicial y esta se va modificando (añadiendo arcos, borrándolos o cambiándolos de dirección) obteniendo una nueva red con mejor medida. En concreto la medida de K2 para una red G y una base de datos D es la siguiente [7]:

$$f(G:D) = \log P(G) + \sum_{i=1}^{n} \left[\sum_{k=1}^{si} \left[\log \frac{\Gamma(n_{ik})}{\Gamma(N_{ik}, n_{ik})} + \sum_{j=1}^{ri} \log \frac{\Gamma(N_{ik}, n_{ik})}{n_{ik}} \right] \right] \quad (2)$$

donde:

- N_{ik} es la frecuencia de las configuraciones encontradas en la base de datos D de las variables x_i.
- n es el número de variables, tomando su j-ésimo valor y sus padres en G tomando su k-ésima configuración,
- si es el número de configuraciones posibles del conjunto de padres,
- Γ_i es el número de valores que puede tomar la variable x_i,
- $N_{ik} = \sum_{j=1}^{\Gamma_i} N_{ik}$ y Γ es la función Gamma.

4.2. Algoritmo genético

Los algoritmos genéticos (AGs) son métodos adaptativos que pueden usarse para resolver problemas de búsqueda y optimización [9]. Están basados en el proceso genético de los organismos vivos. A lo largo de las generaciones, las poblaciones evolucionan en la naturaleza de acorde con los principios de la selección natural y la supervivencia de los más fuertes, postulados por Darwin.

Los algoritmos genéticos usan una analogía directa con el comportamiento natural. Trabajan con una población de individuos, cada uno de los cuales representa una solución factible a un problema dado. A cada individuo se le asigna un valor o puntuación, relacionado con la bondad de dicha solución. En la naturaleza esto equivaldría al grado de efectividad de un organismo para competir por unos determinados recursos.

Una generación se obtiene a partir de la anterior por medio de los operadores de reproducción. Existen dos tipos: Cruza: Se trata de una reproducción de tipo

sexual. Se genera una descendencia a partir del mismo nuúmero de individuos de la generación anterior. Existen varios tipos que no se detallarán en este trabajo. Copia: Se trata de una reproducción de tipo asexual. Un determinado número de individuos pasa sin sufrir ninguna variación directamente a la siguiente generación. Si desea optarse por una estrategia elitista, los mejores individuos de cada generación se copian siempre en la población temporal, para evitar su pérdida. A continuación comienza a generarse la nueva población en base a la aplicación de los operadores genéticos de cruza y/o copia. Una vez generados los nuevos individuos se realiza la mutación con una probabilidad Pm. La probabilidad de mutación suele ser muy baja, por lo general entre el 0,5 % y el 2 %. Se sale de este proceso cuando se alcanza alguno de los criterios de parada, establecidos en el problema a resolver.

Para la aplicación dentro del problema de la generación de estructuras DAG (grafo acíclico dirigido) para redes bayesianas, hemos desarrollado un algoritmo genético que funciona en colaboración con el software Netica, el cual contiene una API de desarrollo útil para la evaluación de resultados de implementaciones de redes bayesianas y la evaluación de las estructuras de las mismas.

El algoritmo que usamos se basa en la descripción que se ha dado anteriormente, se inicia con un DAG aleatorio, el cual es nuestra población inicial, seguido de este paso, se debe evaluar el rango de error de los nodos usando la API del software Netica para obtener un parámetro comparable con las siguientes generaciones que se puedan obtener. Este parámetro se almacena y la siguiente generación se genera para su evaluación, lo que nos otorga un nuevo DAG que debe ser evaluado nuevamente con la API. Hemos establecido un criterio de permanencia del parámetro óptimo (rango de error de los nodos) para poder detener las iteraciones del algoritmo genético cuando se ha encontrado una solución factible, este criterio indica que el rango del DAG almacenado debe prevalecer durante 30 iteraciones para poder ser considerado como un DAG factible para formar la red bayesiana correspondiente a los datos en el dataset. Estableciendo el algoritmo de manera general, quedaríde la siguiente manera.

1. Generación del DAG inicial (aleatorio)
    a) Evaluación del rango de error del DAG
    b) Almacenamiento de rango de error y el DAG
    c) Aumentar número de iteraciones en 1
2. Repetir hasta que el número de iteraciones llegue a 30
    a) Generar nuevo DAG (población)
    b) Evaluar rango de error del nuevo DAG
    c) Si rango de error del nuevo DAG es menor al rango de error del DAG almacenado
        i) Almacenar el nuevo DAG para comparar
        ii) Reiniciar conteo de iteraciones
    d) sino
        i) Aumentar conteo de iteraciones
3. Si el número de iteraciones llega a 30
    i) Devolver el DAG almacenado

5. Resultados experimentales

Se realizaron distintas pruebas para obtener distintos resultados con el uso del algoritmo K2. El algoritmo fue implementado con las libreras existentes de MATLAB. Los entrenamientos de la estructura de la red se llevaron en 2 pruebas distintas con el mismo data set; se tomaron el data set de entrenamiento (80 casos) y el data set de prueba (187 casos). Además, se usaron solamente las primeras 14 características, más el atributo objetivo. El orden de entrada de los nodos para ambos casos fue el siguiente: 10 14 9 4 2 8 12 1 3 5 7 13 11 15 6.

A continuacin se muestran imágenes de las topologías de red bayesiana que se obtuvieron.

```
DAG =

     0     0     0     0     0     0     0     0     0     0     0     0     0     0     0
     0     0     0     0     0     1     0     0     0     0     1     0     0     0     0
     0     0     0     0     0     0     1     0     0     0     0     0     1     0     0
     0     1     0     0     1     0     0     1     0     0     0     0     0     0     0
     0     0     0     0     0     1     0     0     0     0     0     0     0     0     1
     0     0     0     0     0     0     0     0     0     0     0     0     0     0     0
     0     0     0     0     0     0     0     0     0     0     0     0     0     0     0
     1     0     1     0     0     0     0     0     0     0     0     1     1     0     0
     1     0     0     1     0     0     0     0     0     0     0     0     0     0     0
     0     0     0     0     1     0     0     0     1     0     0     0     0     1     1
     0     0     0     0     0     0     0     0     0     0     0     0     0     0     0
     1     0     1     0     0     0     1     0     0     0     0     0     0     0     0
     0     0     0     0     0     0     0     0     0     0     0     0     0     0     0
     1     0     0     1     0     0     0     0     1     0     0     0     0     0     0
     0     0     0     0     0     0     0     0     0     0     0     0     0     0     0
```

Fig. 1. DAG en forma de matriz para el dataset de 187 casos.

El siguiente paso es verificar el margen de error de las redes obtenidas mediante el software Netica.

Lo siguiente es entrenar la red y verificar su rango de error. Una vez hecho esto, obtenemos:

Para la parte de comparación, obtuvimos los siguientes datos del algoritmo genético: Con un total de 25 ejecuciones para el algoritmo, pudimos observar que la configuración para la red bayesiana del data set fue la siguiente, ver la figura 3.

El algoritmo genético nos proporcionó una matriz binaria, la cual pudimos mapear a la red bayesiana presentada en la figura 3. Al entrenar la red con los casos del data set, pudimos observar los resultados de los rangos de error para cada nodo de la red; estos datos se presentan en la tabla siguiente.

Se presenta a continuación una gráfica en la cual se puede observar la diferencia en el rango de error para cada nodo de la red bayesiana, para el algoritmo K2 y para el algoritmo genético.

El rango de error obtenido para cada nodo de la red bayesiana en el algoritmo genético indica el porcentaje de equivocación que tiene cada nodo de la red

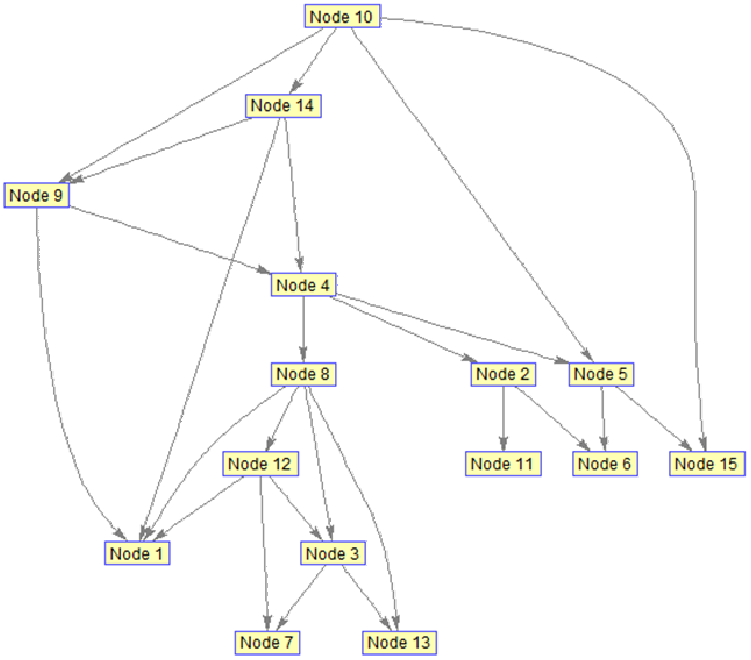

Fig. 2. Red bayesiana resultante.

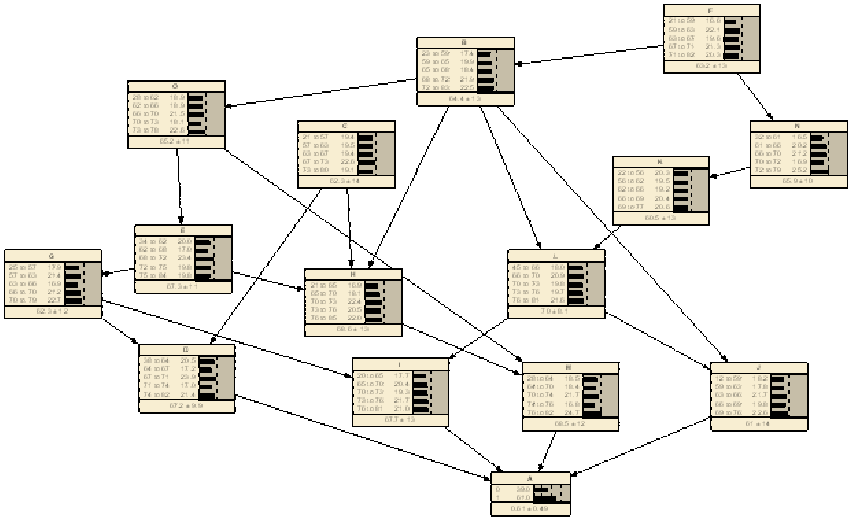

Fig. 3. Red bayesiana obtenida del DAG.

Tabla 1. Rango de error de los nodos para el DAG obtenido del algoritmo K2.

Nodo	Rango de error
A	8.021 %
B	80.21 %
C	79.68 %
D	75.4 %
E	77.54 %
F	80.75 %
G	77.54 %
H	77.54 %
I	77.54 %
J	80.21 %
K	78.07 %
L	78.07 %
M	66.84 %
N	80.75 %
O	73.8 %

Tabla 2. Rango de error de los nodos para el DAG obtenido del algoritmo genético.

Nodo	Rango de error
A	7.034 %
B	76.47 %
C	77.01 %
D	75.4 %
E	78.07 %
F	81.28 %
G	77.54 %
H	79.14 %
I	79.14 %
J	75.94 %
K	79.86 %
L	76.47 %
M	75.4 %
N	72.73 %
O	77.54 %

en base a la evaluación del data set y de la forma en la que los nodos están relacionados.

Es posible observar que dentro de estas algoritmos de clasificación se presentan resultados muy semejantes, sin una variación muy grande. Llegamos a la conclusión de determinar que el algoritmo genético nos brinda mejores resultados, en base a los porcentajes de error que Netica nos permite conocer a través de su software. En estos porcentajes se puede observa una pequeña variación, aunque cabe resaltar que el algoritmo K2 en ciertos nodos el porcentaje es menor al genético. La propuesta del algoritmo genético aún es muy reciente, por lo que existen diversas variaciones que se pueden agregar a este para obtener

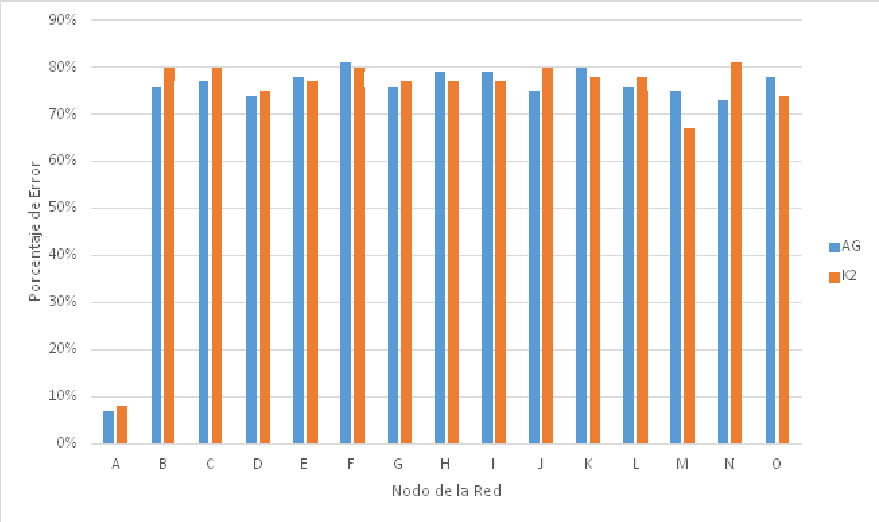

Fig. 4. Gráfica comparativa entre los algoritmos K2 y genético.

una mejor evaluación de las instancia.

Los tiempos de ejecución no se pueden determinar como una medida cualitativa para reportar puesto que estos fueron muy pequeños para ambos casos. En la corrida de los algoritmos para esta base de datos fue de apenas unos segundos, sin una variación realmente significativa, que nos permitiría dar una respuesta favorable a cierto algoritmo.

6. Conclusiones y trabajo futuro

Los resultados que obtuvimos para la estructura de la red bayesiana en el algoritmo K2 y el algoritmo genético (AG) muestran discrepancias, sin embargo, el porcentaje de error que se obtiene en los nodos de la red del AG es menor en el mayoría de los casos, lo que indica que la estructura es más favorable para el diagnóstico del nodo objetivo (A). Para cada uno de los algoritmos, se exige un trabajo mayor, debido a la complejidad de cada uno, ya que, por ejemplo, el algoritmo K2 exige una introducción de los nodos de manera ordenada, sin embargo, este orden debe ser generado de manera óptima para poder asegurar una estructura de la red más efectiva.

Por otro lado, el algoritmo genético exige un mayor número de ejecuciones para poder otorgar una estructura más confiable, ya que pueden darse casos de atasco en mínimos o máximos locales.

El trabajo a futuro de esta propuesta consiste en implementar los métodos de optimización de parámetros para ambos algoritmos, en cuestiones de orden y número de ejecuciones. También se tiene como objetivo futuro extender el rango

de aplicación del data set para poder cambiar el objetivo de los nodos y el sentido de la red.

Referencias

1. Barrientos, M. R., Cruz, R. N., Acosta, M. H. G., Rabatte, S. I., Pavón L. P., Gogeascoechea, T. M. C., Blázquez, M. M. S.: Evaluación del potencial de redes bayesianas en la clasificación en datos médicos. Revista Médica de la Universidad Veracruzana, Vol. 8, No. 1 (2008)
2. Abad-Grau, M. M., Ierache, J. S., Cervino, C.: Aplicación de redes bayesianas en el modelado de un sistema experto de triaje en servicios de urgencias médicas. IX Workshop de Investigadores en Ciencias de la Computación, Argentina, pp. 43–47 (2007)
3. Neapolitan, R.E.: Learning Bayesian networks. Pearson - Prentice Hall (2003)
4. Heckerman, D.: A tutorial on learning with Bayesian networks. Microsoft Research (1995)
5. Sierra Araujo, B. (Coordinador): Aprendizaje automático: Conceptos basicos y avanzados. Capítulo 6, Pearson, Prentice-Hall (2006)
6. Cios, K. J., Wedding, D. K., Liu, N.: CLIP3: cover learning using integer programming. Kybernetes, Vol. 26, No. 4-5, pp. 513–536 (1997)
7. Cooper, G. F., Herskovits, E.: A Bayesian method for the induction of probabilistic networks from data. Machine Learning, Vol. 9, pp. 309–348 (1992)
8. Myers, J., Laskey, K. B.: Learning Bayesian networks from incomplete data with stochastic search algorithms. Proceedings of the Genetic and Evolutionary Computation Conference (GECCO), pp. 458–465 (1999)
9. Haupt, R. L., Haupt, S.E.: Practical genetic algorithms. Wiley-Interscience, 2nd Edition (2004)
10. Gelman, A., Carlin, J.B., Stern, H.S., Dunson, D.B., et al.: Bayesian data analysis. CRC Texts in Statistical Science, Third Edition (2013)