Centro de Investigación en Matemáticas, A.C.

Aplicación de redes bayesianas en el análisis de supervivencia.

T E S I S

Que para obtener el grado de
Maestro en Ciencias
con especialidad en
Probabilidad y Estadística

P r e s e n t a:
Gustavo Bermejo Quezada

Director de tesis:
Dra. L. Leticia Ramírez Ramírez

Autorización de la versión final

Guanajuato, Gto. Agosto, 2019.

Centro de Investigación en Matemáticas, A.C.

Acta de Examen de Grado

Acta No.: 156

Libro No.: 002

Foja No.: 156

En la Ciudad de Guanajuato, Gto., siendo las 16:00 horas del día 01 de octubre del año 2019, se reunieron los miembros del jurado integrado por los señores:

DR. JOHAN JOZEF LODE VAN HOREBEEK **(CIMAT)**
DR. ENRIQUE RAÚL VILLA DIHARCE **(CIMAT)**
DR. ROGELIO RAMOS QUIROGA **(CIMAT)**

bajo la presidencia del primero y con carácter de secretario el segundo, para proceder a efectuar el examen que para obtener el grado de

MAESTRO EN CIENCIAS
CON ESPECIALIDAD EN PROBABILIDAD Y ESTADÍSTICA

Sustenta

GUSTAVO BERMEJO QUEZADA

en cumplimiento con lo establecido en los reglamentos y lineamientos de estudios de posgrado del Centro de Investigación en Matemáticas, A.C., mediante la presentación de la tesis

"APLICACIÓN DE REDES BAYESIANAS EN EL ANÁLISIS DE SUPERVIVENCIA"

Los miembros del jurado examinaron alternadamente al (la) sustentante y después de deliberar entre sí resolvieron declararlo (a):

APROBADO

DR. JOHAN JOZEF LODE VAN HOREBEEK
Presidente

DR. ENRIQUE RAÚL VILLA DIHARCE
Secretario

DR. ROGELIO RAMOS QUIROGA
Vocal

CIMAT
DIRECCIÓN GENERAL

DR. VÍCTOR MANUEL RIVERO MERCADO
Director General

Dedicado

A mis padres:

Tomás Bermejo Ortega

Josefina Quezada Rangel

Agradecimientos

Primero que nada, gracias a dios por permitirme concluir este nivel de estudios. A mi familia, por su apoyo, confianza y amor incondicional, por la motivación de siempre seguir adelante.

Al CIMAT y todos mis profesores que me formaron en este largo trayecto. Gracias por ayudarme a crecer tanto intelectual como personalmente. Además, agradezco al CONACYT por la beca No. 489606 con No. de Registro 635341, la cual permitió mi estancia en Guanajuato al brindarme su apoyo económico para la realización de mis estudios de maestría.

A mi asesora, la Dra. Leticia Ramírez, por su orientación, gran paciencia y apoyo en la elaboración de este trabajo de investigación, gracias por siempre estar disponible para sacar a delante este trabajo. También, a mis sinodales, el Dr. Johan Jozef Lode Van Horebeek, el Dr. Enrique Raúl Villa Diharce y el Dr. Rogelio Ramos Quiroga por su ayuda en la revisión de mi tesis.

A mis compañeros del cubo, roomies y en general a todos los que me brindaron algún tipo de apoyo. Muchas gracias por su ayuda y por estar ahí en momentos difíciles.

Resumen

En el análisis de supervivencia el modelo más utilizado es el modelo de riesgos proporcionales de Cox. Este modelo es capaz de representar una relación entre un conjunto de riesgos y su efecto en común. Por otro lado, las redes bayesianas se han convertido en una alternativa atractiva para el estudio de tiempos de falla con alto poder de modelado y amplias aplicaciones. Este trabajo explora la propuesta de Kraisangka & Druzdzel (2018), donde se propone un método de selección y ajuste de redes bayesianas a través del modelo de riesgos proporcionales de Cox. Por otro lado proponemos extenderlo al considerar otros métodos de selección de red bayesiana. Entre estos métodos destaca el hacer el aprendizaje de la estructura de la red bayesiana a partir de los datos mediante un algoritmo diferencial evolutivo propuesto por Baioletti et al. (2018), el cual introduce un marco algebraico que permite aplicar evolución diferencial a problemas combinatorios en los que el espacio de búsqueda es un grupo finamente generado. Este trabajo de investigación presenta las redes bayesianas como una alternativa para el estudio de los tiempos de supervivencia de un objeto de interés. Gracias al algoritmo diferencial evolutivo se propone fácilmente una restricción propia de un modelo de supervivencia. Se contrastan los resultados con algunas alternativas.

Palabras Clave

Redes bayesianas, Análisis de supervivencia, Modelo de riesgos proporcionales de Cox.

índice

Índice de figuras

Índice de tablas

Introducción

Es de gran interés conocer el tiempo esperado de vida de un organismo biológico o el tiempo esperado de falla de un objeto en un sistema. A estos tiempos de interés se les conoce como tiempos de falla o tiempos de supervivencia. El análisis estadístico de los tiempos de falla es un tema importante en muchas áreas, incluidas las ciencias biomédicas, de ingeniería y sociales. Algunos métodos para tratar con tiempos de falla son bastante antiguos, pero a partir de 1970 el campo se expandió rápidamente con respecto a la metodología, la teoría y los campos de aplicación y los paquetes de software para el análisis de tiempos de falla han estado disponibles desde aproximadamente 1980.

Por otro lados las redes bayesianas son modelos gráficos probabilísticos capaces de modelar la distribución de probabilidad conjunta sobre un conjunto finito de variables aleatorias y sus dependencias condicionales a través de un grafo acíclico dirigido. Las redes bayesianas son bien estructuradas, intuitivas y fáciles de implementar computacionalmente. Tienen la capacidad de modelar explícitamente las dependencias entre los factores de riesgo, manejar la complejidad del modelo y ofrecer más flexibilidad en la interpretación del modelo.

Las redes bayesianas tienen múltiples aplicaciones, aunque se utilizan a menudo para repre-

sentar relaciones causales, pero no tienen que estar restringidas a tales casos. Según Mittal (2007), las redes bayesianas han mostrado un rendimiento superior en comparación con las redes neuronales, las máquinas de soporte vectorial, árboles de decisión, etc., para varias tareas de clasificación, como la extracción de datos, el monitoreo de fallas, la bioinformática.

Las redes bayesianas han surgido como una alternativa atractiva con alto poder de modelado y amplias aplicaciones. Por lo que en esta investigación presenta las redes bayesianas como una alternativa para el estudio de los tiempos de supervivencia de un objeto de interés. El objetivo principal de este trabajo es abundar en la modelación y uso de redes bayesianas en el caso particular de análisis de supervivencia. Es importante mencionar que este trabajo aborda el uso redes bayesianas con variables discretas, por lo que la variable de supervivencia, si es continua, se discretiza con el fin de hacer el análisis.

Existen ya aplicaciones de redes bayesianas para el análisis de supervivencia encontradas en la literatura. En la industria (Jones et al. (2010)); en medicina (Bandyopadhyay et al. (2015), Štajduhar & Dalbelo-Bašić (2010) y Berzuini et al. (1992)).

En este trabajo de investigación se pretende extender el artículo de Kraisangka & Druzdzel (2018), el cual propone un método de construcción y ajuste de red bayesiana para el estudio de tiempos de falla donde presenta como alternativa de ajuste, el uso de un modelo de riesgos proporcionales de Cox. Sin embargo las redes resultantes no considera la relación entre covariables, lo que desperdicia el potencial de las redes bayesianas. Así, de lo anterior, este trabajo de investigación pretende abundar en la modelación y uso de redes bayesianas en el caso particular de análisis de supervivencia. Es decir, el problema de este trabajo es estudiar las ventajas y desventajas del uso, ajuste e inferencia de las redes bayesianas como una alternativa a modelos de supervivencia.

Una familia de propuestas de aprendizaje de la red radica en los algoritmos evolutivos. En es-

ta tesis de aborda el recientemente algoritmo de evolución diferencial propuesto por Baioletti et al.. Aunque originalmente se propuso la evolución diferencial para problemas continuos, este trabajo presenta un marco algebraico propuesto por Baioletti et al. que permite aplicar evolución diferencial a problemas combinatorios en los que el espacio de búsqueda es un grupo finamente generado. Su trabajo ofrece una representación novedosa de la estructura de red bayesiana que permite ver el espacio de búsqueda de todas las estructuras de red bayesiana de un conjunto de vértices fijos como un grupo de productos (operaciones de un grupo algebraico). De esta manera, es posible aplicar evolucion diferencial al problema de aprendizaje de red bayesiana en términos de encontrar la estructura con la puntuación máxima que mejor represente los datos dados.

En esta tesis proponemos utilizar una modificación al algoritmo diferencial evolutivo que restringe las redes originadas para adecuarse a redes que modelen tiempos de falla. Usando algunos datos sintéticos, esta propuesta se contrasta con métodos de aprendizaje competitivos.

Este trabajo de investigación esta estructurado de la siguiente manera, el Capítulo 1 presenta a grandes rasgos como el análisis de supervivencia hace el estudio de tiempos de falla, en el Capítulo 2 se muestran los principales conceptos y propiedades referentes a redes bayesianas, por otra parte, el Capítulo 3 presenta la unificación de la teoría de análisis de supervivencia y las redes bayesianas para el estudio de tiempos de falla. En éste capítulo se presenta el algoritmo de evolución diferencial y su marco algebraico, así como los experimentos computacionales. La implementación computacional se programó usando el software estadístico R (R Core Team (2013)) y el código de programación que se utiliza en este trabajo de investigación puede consultarse en https://github.com/gustavoberzada/Codigo_Tesis_Gustavo. Finalmente en el Capítulo 4 se exhiben las conclusiones obtenidas de este trabajo.

CAPÍTULO 1

Fundamentos teóricos del análisis de supervivencia

El análisis de supervivencia intenta responder preguntas como: ¿Cuál es la proporción de una población que sobrevivirá más allá de un cierto tiempo?, ¿cúal es número de horas promedio que funciona un equipo electrónico nuevo?, ¿cúal es el periodo de garantía que se puede dar a un producto que cause reclamos de menos del 1 %?, ¿con qué probabilidad una persona que pierde el empleo encontraría uno nuevo antes de un año?.

En el análisis de supervivencia, el interés se centra en un grupo o grupos de individuos para cada uno de los cuales se define un evento puntual, a menudo llamado *falla*, que ocurre después de un período de tiempo llamado *tiempo de falla*. En esta tesis nos enfocamos en el caso en el cual el fracaso puede ocurrir a lo sumo una vez en cualquier individuo y su ocurrencia o identificación está libre de incertidumbre. Ejemplos de tiempos de falla incluyen la vida útil de las componentes de la máquina en la confiabilidad industrial, la duración de las huelgas o los períodos de desempleo de las personas en una economía, los tiempos tomados por los sujetos para completar tareas específicas en la experimentación psicológica, las longitudes de

las pistas en una placa fotográfica en la física de partículas y los tiempos de supervivencia de los pacientes en un ensayo clínico. El análisis de supervivencia es el conjunto de métodos estadísticos que tienen como objetivo modelar la relación entre un conjunto de variables predictoras y una variable de respuesta, en particular, es de interés la predicción del momento en que ocurre el evento de interés.

Una fuente especial de dificultad en el análisis de los datos de supervivencia es la posibilidad de que algunas personas o unidades no sean observadas durante el tiempo de estudio. Por ejemplo, al final de un experimento de confiabilidad en la industria puede que no todos los componentes fallen. A las observaciones incompletas donde el valor de una observación sólo se conoce parcialmente en el tiempo de estudio de una componente se les llama datos censurados. El análisis de supervivencia es capaz de trabajar con estos datos incompletos.

En este capítulo se abordan las bases de los elementos estadísticos presentes en el análisis de supervivencia. En la Sección 1.1 se define la función de supervivencia, así como la función de riesgo, siendo estas funciones la base en la que el análisis de supervivencia comienza la construcción de su teoría. En la Sección 1.2 se presentan dos tipos de observaciones que son ampliamente estudiadas en el análisis de supervivencia. La Sección 1.3 expone los modelos no paramétricos comúnmente utilizados en el análisis de supervivencia para estimar la probabilidad de falla y el riesgo al evento, respectivamente. Finalmente, en la Sección 1.4 aborda el modelo de riesgos proporcionales de Cox, el cual es uno de los modelos más utilizados en el análisis de supervivencia.

1.1. Introducción análisis de supervivencia

El análisis de supervivencia es una rama de la estadística que modela el tiempo hasta que uno o más eventos ocurren, tal como la muerte de un organismo o la falla mecánica de un sistema. Sin embargo, el análisis de supervivencia no sólo permite conocer cualidades como

los tiempos esperados de vida, sino que puede utilizarse para efectos de predicción y establecer la importancia de covariables en la incidencia del evento, con lo que se pueden delinear potenciales acciones para alargar la vida o funcionamiento.

Esto ya que se puede realizar un análisis de supervivencia para estimar el tiempo hasta que un evento de interés ocurra para un grupo, comparar el tiempo hasta que el evento ocurra entre dos o más grupos, o para estudiar la relación entre las variables y los tiempos de ocurrencia del evento.

El análisis de supervivencia recibe distintos nombres dependiendo del área en la que se utiliza el término. Por ejemplo, en estadística también se le conoce como *tiempo al evento (time-to-event analysis)*; en ingeniería, al relacionarse fuertemente con confiabilidad, el análisis de supervivencia adopta este nombre *confiabilidad (reliability analysis)*, en economía, se nombra *análisis de duración*, en sociología se le llama *historia de eventos (event history analysis)*. Debido a esta diversidad de aplicaciones, el tiempo donde ocurre el evento de interés se puede referir como *tiempo del evento, tiempo de de supervivencia o tiempo de falla*. En este trabajo de investigación nos referiremos al análisis de supervivencia como *análisis de supervivencia* o *estudio de tiempos de falla*.

Algunos métodos para tratar con tiempos de falla son bastante antiguos, pero de acuerdo con Lawless (2011) es a partir de 1970 que el campo se expandió rápidamente tanto con respecto a la metodología, y su teoría como los campos de aplicación. Los programas computacionales para el análisis de tiempos de falla han estado ampliamente disponibles desde aproximadamente 1980, con la aparición frecuente de nuevas características y paquetes. Sin embargo, uno de los precursores de la temprana metodología en el análisis de supervivencia es el ingles John Graunt (1620-1674), quien se considera como uno de los primeros demógrafos y fundadores de la bioestadística y epidemiología con su publicación en 1662 *Natural and Political Observations Made upon the Bills of Mortality*. De acuerdo con Meeker et al. (1998),

en relación a la confiabilidad, esta área emergió en aplicaciones tecnológicas después de la Primera Guerra Mundial y se utilizó conectada a la seguridad operacional de aeroplanos. El evento que se medía era el número de accidentes por hora durante operación. Ya en la década de los treintas, se establecieron bases teóricas para el uso de métodos estadísticos en el control de calidad de productos industriales, cuyo uso se dispersó hasta después de terminada la Segunda Guerra Mundial. Hacia finales de la década de los cincuentas, en Estados Unidos, el interés tecnológico y su calidad, se concentró en misiles de largo alcance.

En este capítulo se presentan las definiciones básicas del análisis de supervivencia, con énfasis en el caso de que el evento de interés T tenga una distribución continua, sin embargo se tienen definiciones y resultados análogos para variables aleatorias discretas. Parte del material presentado en este trabajo de investigación se ha tomado de Lawless (2011), Collett (2015), Klein & Moeschberger (1997), Kraisangka & Druzdzel (2018).

En el análisis de supervivencia, una de las principales funciones que se define es la función de supervivencia, la cual indica la probabilidad de que el individuo llegue vivo a un cierto tiempo.

Definición 1.1.1. *Para un tiempo dado t la función de supervivencia se define como*

$$S(t) = P(T > t). \tag{1.1}$$

En esta definición, T es una variable que denota el tiempo de ocurrencia de un evento de interés. Así pues T se considerará una variable aleatoria con soporte en $[0, \infty)$. Por lo que la función de supervivencia representa la probabilidad de sobrevivir más allá de un tiempo t luego de un origen (nacimiento, construcción, inicio de funcionamiento). En el caso continuo se tiene que $S(t) = 1 - F(t)$ (donde $F(t)$ denota la función de distribución del tiempo T), pero no es así para el caso discreto. Por otro lado cabe resaltar que alguna literatura de análisis de supervivencia define la función de supervivencia como $S(t) = P(T \geq t)$.

Otra función de gran importancia, es la función de riesgo (*hazard function* o *hazard rate*), función que describe el riesgo instantáneo de ocurrencia del evento de interés.

Definición 1.1.2. *La función de riesgo se define como*

$$h(t) = \lim_{\Delta t \to 0} \frac{P(t \leq T < t + \Delta t | T \geq t)}{\Delta t}. \tag{1.2}$$

La función de riesgo representa el peligro de que ocurra el evento de interés en el instante t. Esta función de riesgo también se le denomina tasa de riesgo, la cual es una medida de riesgo en un pequeño intervalo de tiempo Δt. De la Definición 1.1.1 es fácil verificar que

$$h(t) = \frac{f(t)}{S(t)}, \tag{1.3}$$

donde $f(t)$ denota la función de densidad de T. También se puede establecer relación entre la función de riesgo y la función de supervivencia:

$$h(t) = -\frac{d}{dt} \log S(t), \tag{1.4}$$

o equivalentemente, se tiene que

$$S(t) = \exp\left\{ - \int_0^t h(u) du \right\}, \tag{1.5}$$

si se supone que $S(0) = 1$. De la ecuación (1.5) es natural definir la función de riesgo acumulado, la cual se interpreta como el peligro que tiene el evento de ocurrir en el intervalo de tiempo $[0, t]$.

Definición 1.1.3. *La función de riesgo acumulado hasta el tiempo t denotada como $H(t)$ se define como:*

$$H(t) = \int_0^t h(u) du. \tag{1.6}$$

Así que de (1.3) se tiene:

$$f(t) = h(t)\exp\{-H(t)\}. \tag{1.7}$$

Así pues con las expresiones anteriores es posible estimar la probabilidad de supervivencia a partir de la función de riesgo. En el análisis de supervivencia, la función de riesgo puede representarse mediante cualquier distribución de probabilidad o puede estimarse con modelos no paramétricos o modelarse y estimarse mediante técnicas de regresión.

Para estimar la probabilidad de supervivencia, las distribuciones de probabilidad más utilizadas en el análisis de supervivencia corresponden a los modelos: Exponencial, Gama, Weibull, Pareto, Log-normal, Log-logistico, Inversa Gausiana. Estos modelos son llamados *modelos paramétricos*, ya que las correspondientes distribuciones dependen de parámetros. Por otra parte, como ya se mencionó la función de riesgo puede modelarse mediante modelos no paramétricos y técnicas de regresiones.

Para los modelos no paramétricos se tienen los estimadores *Nelson-Aalen* y *Kaplan-Meier* de la función de riesgo acumulado y de la función de supervivencia respectivamente, y para técnicas de regresión se tiene el modelo de riesgos proporcionales de Cox y variaciones de este, por ejemplo, el modelo de riesgos proporcionales adaptado a variables dependientes del tiempo.

Se definen otras funciones auxiliares que permiten identificar fácilmente cualidades de la función de supervivencia, tales como la función de vida media residual denotada como $m(t)$ y que mide la esperanza de vida restante para un individuo.

Definición 1.1.4. *La vida media residual se define como:*

$$m(t) \quad = \quad E[T - t|T > t]. \tag{1.8}$$

Entonces, la función de vida media residual es el tiempo restante esperado de vida de un individuo que ha llegado a la edad t. Se puede probar que si la función de vida media residual

$m(t)$ existe entonces, está dada por:

$$m(t) \quad = \quad \int_t^{\infty} \frac{S(x)}{S(t)} dx. \tag{1.9}$$

Es importante notar que la función de vida media residual evaluada en cero es igual a la media de la distribución, esto es:

$$m(0) = \int_0^{\infty} S(t)dt, \tag{1.10}$$

ya que T es una variable aleatoria no negativa. Por otro lado, es posible verificar la relación

$$S(t) = \frac{m(0)}{m(t)} \exp\left\{ - \int_0^t \frac{du}{m(u)} \right\}, \tag{1.11}$$

siempre que $m(t)$ exista para todo $t \geq 0$.

1.2. Características de los datos

Comúnmente los datos de supervivencia no se pueden observar completamente. Ya sea por el diseño de los experimentos, limitaciones en los presupuestos para hacer seguimientos, el sesgo en la incorporación de los individuos en la observación, la información sobre los tiempos de falla no son observaciones de realizaciones de T sino una modificación de ésta. Ahora describimos dos tipos de modificaciones o características que los datos presentan en muchas ocasiones.

En este trabajo se presentan la censura por la derecha y el truncamiento por la izquierda. Por lo tanto se pueden analizar datos, que posiblemente vendrán como una mezcla de observaciones completas e incompletas. En cada uno de los tipos de datos que se presentan en este trabajo se expresan sus respectivas verosimilitudes. En este sentido, los distintos tipos de observaciones, censuradas y/o truncadas contribuyen a la verosimilitud de diferente manera.

En el análisis de supervivencia los *datos censurados* son aquellos donde no se conoce el tiempo hasta la aparición del fracaso o éxito del evento de interés, ya sea porque el individuo se retiró del estudio, o bien porque se acabó el estudio. Existen distintos tipos de datos censurados, por ejemplo censura por la izquierda, por la derecha, por intervalo, o incluso aleatoria. Una segunda característica de muchos estudios de supervivencia, a veces confundida con censura, es el truncamiento. El truncamiento de los datos de supervivencia ocurre cuando sólo se observan aquellos individuos cuyo tiempo de evento se encuentra dentro de un cierto intervalo de tiempo, no se observa a una persona cuyo tiempo de falla no está en este intervalo y no hay información disponible sobre el número de individuos que no se incluyeron en la observación. Esto contrasta con la censura donde hay al menos información parcial sobre cada individuo. Debido a que sólo conocemos individuos con tiempos de eventos en el intervalo de observación, la inferencia para datos truncados está restringida a la estimación condicional (ver Sección 1.2).

Censura por la derecha

Este tipo de censura considera observaciones que están por encima de un cierto umbral, pero se ignora *cuánto* más está por encima. Se supone que n individuos tienen tiempos de vida representados por variables aleatorias $T_1, \ldots, T_n$. Para simplificar, se asume que los tiempos T_i son independientes e idénticamente distribuidos. Muchas veces en lugar de los valores observados para cada tiempo de vida T_i se tiene un tiempo t_i que se sabe que es el tiempo de vida o el tiempo de censura. Se define la variable $\delta_i = \mathbb{1}_{(T_i = t_i)}$, donde $\mathbb{1}_{(A)}$ denota la función indicadora del conjunto A, esto es,

$$\mathbb{1}_A(x) = \begin{cases} 1; & \text{si } x \in A, \\ 0; & \text{si } x \notin A. \end{cases} \tag{1.12}$$

Así $\delta_i = 1$ si $T_i = t_i$ y $\delta_i = 0$ si es que $T_i > t_i$, es decir, si $\delta_i = 1$ se tiene un dato completo y si $\delta_i = 0$ se tiene un dato censurado por la derecha. Por lo tanto los datos con los que se cuentan son $\{(t_i, \delta_i)\}_{i=1}^n$. El resultado más importante es que para observaciones con censura

por la derecha, la verosimilitud está dada de la forma:

$$L = \prod_{i=1}^{n} f(t_i)^{\delta_i} S(t_i)^{1-\delta_i}, \tag{1.13}$$

donde $f(t_i)$ es la función de densidad de los datos y finalmente $S(t_i)$ es la función de supervivencia.

Truncamiento por la izquierda.

El truncamiento se produce cuando los valores más allá de un límite se excluyen al momento de la recopilación de los datos. El truncamiento por la izquierda se interpreta como el tiempo de retraso en la entrada a un estudio. Los individuos a veces se seleccionan y siguen de forma prospectiva hasta el fracaso o la censura, pero su tiempo de vida actual en la selección no es $t = 0$, sino algún valor $u > 0$. La definición de un estudio prospectivo es que la información de tiempo de vida después del momento de la selección constituye la respuesta. La selección de un individuo en el tiempo u_i requiere que $T_i > u_i$, y los datos observados para el individuo i consistan en $\{(u_i, t_i, \delta_i)\}$, donde $t_i > u_i$ es un tiempo de vida o de censura. Se dice que el tiempo de vida T_i es truncado por la izquierda en u_i (si $u_i > 0$).

Sea $S(t)$ la función de supervivencia de T. El problema crucial que afecta la inferencia al momento de trabajar con datos truncados por la izquierda, es la distribución de T dado u, y el hecho de que $T \geq u$ está dada por la distribución truncada con función de supervivencia $S(t)/S(u)$ para $t \geq u$. Más específicamente, en términos de la función de riesgo se necesita que:

$$P(T = t | T \geq t, u, T \geq u) = P(T = t | T \geq t). \tag{1.14}$$

Suponiendo que se cumple la ecuacón (1.14) y si se consideran n individuos tienen tiempos de vida representados por variables aleatorias $T_1, \ldots, T_n$ independientes e idénticamente

distribuidos, se tiene que la verosimilitud esta dada por:

$$L = \prod_{i=1}^{n} \left[\frac{f(t_i)}{S_i(u_i)}\right]^{\delta_i} \left[\frac{S(t_i)}{S_i(u_i)}\right]^{1-\delta_i}. \tag{1.15}$$

1.3. Modelos no paramétricos

Las gráficas y los resúmenes de datos son fundamentales ya que su descripción y análisis preliminar permiten conocer sus características principales y eliminar modelos que obviamente les contradicen. En el contexto de supervivencia este análisis preliminar está estrechamente relacionados con estimaciones no paramétricas de las características de distribución. Los principales estimadores son los estimadores de *Nelson-Aalen y Kaplan-Meier.*

1.3.1. Nelson-Aalen

El estimador Nelson-Aalen (NA) es un estimador de la función de riego acumulado $H(t)$. Éste considera datos $\{(t_i, \delta_i)\}_{i=1}^{n}$ de n individuo. De los cuales los tiempos con los que se cuentan son posiblemente censurados, y se ordenan de tal forma que $t_1 < t_2 < t_3 < \ldots < t_k$ para $k \leq n$. El estimador Nelson-Aalen se define como:

$$\widehat{A}(t) = \sum_{j:t_j \leq t} \frac{d_j}{n_j}\,, \tag{1.16}$$

donde $d_j = \sum_{i=1}^{n} \mathbb{1}_{(t_i=t_j,\delta_i=1)}$, $n_j = \sum_{i=1}^{n} \mathbb{1}_{(t_i \leq t_j)}$. Así d_j corresponde al número de fallas observadas en los datos, al tiempo t_j y n_j indica el número de entes en riesgo al tiempo t_j ; es decir el número de peronas que potencialmente se pudieron fallar al tiempo t_j. Notar que n_j incluye a las personas censuradas al tiempo t_j porque estaban aún en observación al tiempo $t_j - \epsilon$.

El estimador Nelson-Aalen considera las contribuciones al riesgo, infinitesimales. Si no se observan muertes en un intervalo, su riesgo se estima como 0 y si tiene d_j fallecimientos con n_j individuos que puede que hayan fallecido durante este intervalo, entonces la contribución

se puede estimar como d_j/n_j. El estimador NA es el mismo para cuando el tiempo T es discreto o continuo, sin embrago para el caso continuo en ocasiones este estimador es utilizado como una estimación alternativa de la función de supervivencia a través de la relación $S(t) = \exp(-H(t))$.

Es fácil verificar que el estimador Nelson-Aalen cuando T es discreto es creciente y escalonado, con posibles saltos en $\{t_j\}$ y cumple que es una función continua por la derecha (ver el Ejemplo 1.3.1).

Ejemplo 1.3.1. *Se obtiene el estimador NA para los siguientes datos:*

$$13, 12, 14, 12, 13+, 15+$$

donde + denota dato censurado por la derecha: Ahora, usando R para el estimador de

Tiempo	Status	t_i	d_i	Y_i	$\hat{A}(t) = \sum_{j:\, t_j \leq t} \frac{d_j}{n_j}$
0	1	0	0	6	0
12	1	12	2	6	2/6
12	1	13	1	4	2/6+1/4
13	1	14	1	2	2/6+1/4+1/2
13	0	15	0	1	2/6+1/4+1/2
14	1				
15	0				

Tabla 1.1: Estimador Nelson-Aalen

Nelson-Aalen, se tiene la Figura 1.1.

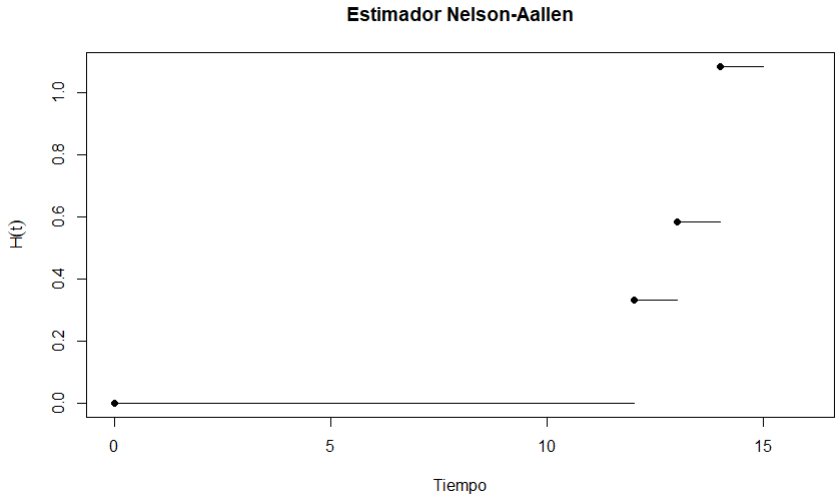


Figura 1.1: Estimador de Nelson-Aalen

La Figura 1.1 muestra el estimador Nelson-Aalen para la función de riesgo acumulado de los datos de la Tabla 1.1. Se puede ver que este estimador es creciente y escalonado, con saltos en $\{t_j\}$ *y cumple que es una función continua por la derecha.*

Intervalos de confianza

El conocer estimaciones de la varianza del estimador de Nelson-Aalen permite la construcción de intervalos de confianza así como la realización de pruebas de hipótesis. Dos aproximaciones a la varianza del estimador de NA son las siguientes:

- Con las propiedades del estadístico de máxima verosimiltud para muestras grandes, se puede probar que:

$$\widehat{\text{Var}}\left(\widehat{A}(t)\right) = \sum_{j:t_j \leq t} \frac{d_j(n_j - d_j)}{(n_j)^3}, \tag{1.17}$$

  donde d_j corresponde al número de fallas observadas en los datos, al tiempo t_j y n_j indica el número de entes en riesgo al tiempo t_j.

- Aalen et al. (2008) hizo una estimación de la varianza al ver este estimador como un

proceso de renovación y usando propiedades de martingalas, resultando:

$$\widehat{\text{Var}}\left(\widehat{A}(t)\right) = \sum_{j:t_j \leq t} \frac{d_j}{(n_j)^2}, \tag{1.18}$$

con d_j el número de fallas y n_j indica el número de individuos en riesgo al tiempo t_j.

1.3.2. Kaplan-Meier

Es un estimador de la función de supervivencia $S(t)$. También se conoce como el estimador producto límite. Al igual que el estimador de Nelson-Aalen, consideremos $\{(t_i, \delta_i)\}$ representan los tiempos los cuales son distintos y ordenados. éstos pueden corresponder a tiempos de falla o censura observados. El estimador de Kaplan-Meier (KM) se define como:

$$\widehat{S}(t) = \begin{cases} 1 & \text{si } t < t_1 \\ \prod_{i:t_i \leq t} \left(1 - \frac{d_i}{n_i}\right) & \text{en otro caso,} \end{cases} \tag{1.19}$$

donde $d_i = \sum_{i=1}^{n} \mathbb{1}_{(T_j = t_i, \delta_i = 1)}$ y $n_i = \sum_{i=1}^{n} \mathbb{1}_{(T_j > t_i)}$. Cuando los datos no tienen observaciones censuradas, este estimador coincide con el estimador empírico de la supervivencia

$$\widehat{S}(t) = \frac{\sum_{i=1}^{n} \mathbb{1}_{(T_i \geq t)}}{n}. \tag{1.20}$$

Intervalos de confianza

Una aproximación a la varianza del estimador KM se obtiene a través de la fórmula de Greenwood, donde usando propiedades de martingalas se obtiene que:

$$\widehat{\text{Var}}\left(\widehat{S}(t)\right) = \widehat{S}(t)^2 \sum_{j:t_j \leq t} \frac{d_j}{n_j(n_j - d_j)}, \tag{1.21}$$

con d_j el número de fallas y n_j indica el número de individuos en riesgo al tiempo t_j.

Ejemplo 1.3.2. *La base de datos "tongue" de Klein & Moeschberger (1997) proporciona información de individuos con cáncer de lengua, cuenta con 80 filas y 3 columnas. Esta base de datos contiene las siguientes columnas: tipo de tumor (1 = tumor aneuploide, 2 = tumor*

diploide), tiempo hasta la muerte o tiempo de estudio (en semanas), indicador de muerte (0 = vivo, 1 = muerto). Se obtiene el estimador Kaplan-Meier para la base de datos "tongue" del paquete survival de R:

$$1, 3, 3, 4, 10, 13, 13, 16, 16, 24, 26, 27, 28, 30, 30, 32, 41, 51, 65, 67, 70, 72, 73, 77,$$
$$91, 93, 96, 100, 104, 157, 167, 61+, 74+, 79+, 80+, 81+, 87+, 87+, 88+, 89+,$$
$$93+, 97+, 101+, 104+, 108+, 109+, 120+, 131+, 150+, 231+, 240+, 400+$$

donde + denota dato censurado por la derecha.

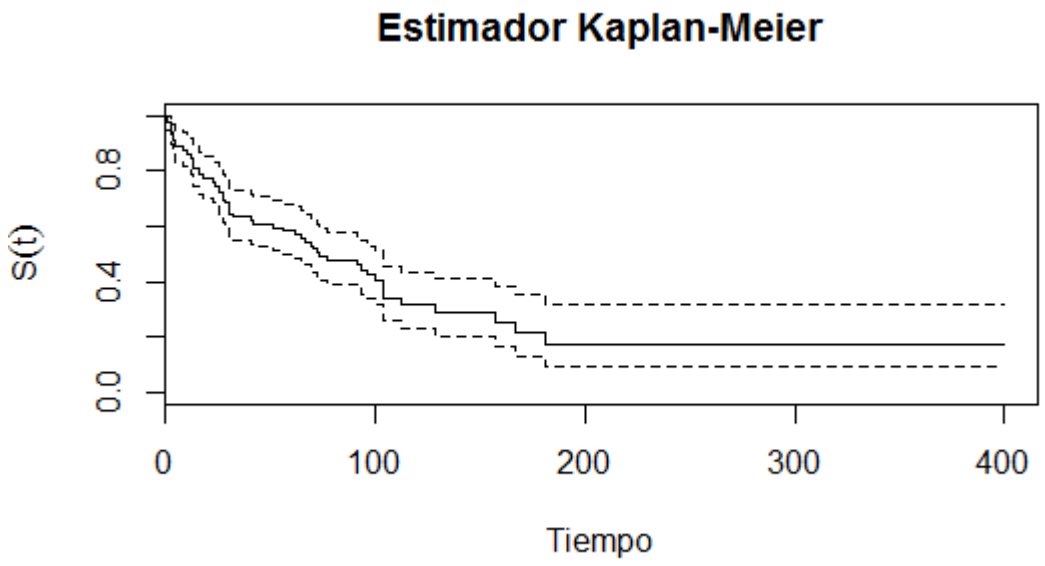


Figura 1.2: Estimador Kaplan-Meier para la base de datos tongue

Utilizando R se obtiene la Figura 1.2, la cual muestra la estimación de la función de supervivencia de la base de datos "tongue"y su banda de confianza de aproximadamente el 90 % creada a partir de 1.21.

1.4. Modelo de riesgos proporcionales de Cox

Los estimadores Nelson Aalen y Kaplan-Meier consideran que la población es homogénea, en el sentido que cada uno de los individuos en el estudio tienen exactamente la misma exposición al riesgo. Esto es equivalente a considerar que T_i son variables aleatorias independientes e idénticamente distribuidas. Cuando se tienen subpoblaciones, el procedimiento común es hacer la estimación de la función de superveniencia por separado para cada una de ellas. En contraste con este procedimiento, el modelo de riesgo proporcionales de Cox (CPH por las iniciales de su nombre en inglés, *Cox's proportional hazard*), también se conoce como regresión de Cox o simplemente modelo de Cox, incorpora toda la información de la población y trata de describir la supervivencia en función de las covariables que describan la heterogeneidad de la población al riesgo de presentar el evento. Así, éste modelo no sólo permite integrar todas las observaciones en un sólo modelo, sino que proporciona una evaluación de la supervivencia basada en factores de riesgo asociados con el evento. La función de riesgo en el modelo de Cox es

$$h(t) = h_0(t) \exp(\boldsymbol{\beta' X}), \tag{1.22}$$

donde $h_0(t)$ corresponde a la función de riesgo base para todos los individuos, $\boldsymbol{\beta}$ son los coeficientes que modelan el efecto de las covariables $\boldsymbol{X} = X_1, \ldots, X_n$. La función de riesgo base determina los riesgos cuando todos los factores de riesgo están ausentes. Cuando las covariables son constantes en el tiempo, el modelo propuesto por Cox (1972) establece que la relación entre los riesgos de muerte entre dos individuos expuestos a factores (covariables) distintos son proporcionales. A partir de éste modelo es posible expresar la función de riesgos acumulado de la siguiente forma

$$H(t) = H_0(t) \exp(\boldsymbol{\beta' X}). \tag{1.23}$$

Consecuentemente por la ecuación (1.7) se tiene:

$$S(t) = S_0(t)^{\exp(\boldsymbol{\beta' X})}. \tag{1.24}$$

Para hacer el ajuste del modelo de regresión de Cox dado en la ecuación (1.22) a un conjunto observado de datos de supervivencia implica estimar los coeficientes desconocidos $\beta_1, \beta_2, \ldots, \beta_n$ de las variables explicativas, $X_1, X_2, \ldots, X_n$, en el componente lineal del modelo y la función de riesgo base, $h_0(t)$. Resulta que estos dos componentes del modelo pueden estimarse *por separado*, según Collett (2015). Así los valores β's se estiman primero y luego se usan estas estimaciones para construir una estimación de la función de riesgo base. Este es un resultado importante, ya que significa que para hacer inferencias sobre los efectos de las variables explicativas, $X_1, X_2, \ldots, X_n$, en el riesgo relativo, $h_i(t)/h_0(t)$, no se necesita una estimación de $h_0(t)$.

De lo anterior, los coeficientes $\boldsymbol{\beta}$ en el modelo de regresión de Cox pueden estimarse mediante máxima verosimilitud. Para operar de esta forma, primero obtenemos la probabilidad conjunta de los datos observados considerados como una función de los parámetros desconocidos en el modelo asumido. La verosimilitud para el modelo de regresión de Cox está en función de los tiempos de supervivencia observados y los parámetros $\boldsymbol{\beta}$ de la componente lineal del modelo. Las estimaciones de los $\boldsymbol{\beta}$ son entonces aquellos valores que son más probables en la base de los datos observados.

Con el fin de hacer la estimación de $\boldsymbol{\beta}$ y de lo descrito en el párrafo anterior, Cox (1975) propuso usar una expresión llamada verosimilitud parcial la cual depende solo de $\boldsymbol{\beta}$ (parámetros de interés) y no de la función de riesgo base. Esta verosimilitud parcial se puede usar para la inferencia de muestras grandes, exactamente como una verosimilitud ordinaria, es decir, la verosimilitud parcial cumple las mismas propiedades asintóticas de la verosimilitud ordinaria.

Para la construcción de la verosimilitud parcial se consideran n individuos con $n - k$ datos censurados, es decir se conoce el tiempo de muerte de k individuos. Luego para $\{(t_i, \delta_i)\}_{i=1}^{n}$ se tiene los tiempos de muerte ordenados $t_{(1)}, t_{(2)}, \ldots, t_{(k)}$. La verosimilitud parcial esta dada

por:

$$L = \prod_{i=1}^{k} \frac{\exp\{\boldsymbol{\beta}' \boldsymbol{X}_{(i)}\}}{\sum_{l \in N(t_{(i)})} \exp\{\boldsymbol{\beta}' \boldsymbol{X}_{l}\}}. \tag{1.25}$$

donde $\boldsymbol{X}_{(i)}$ es la covariable correspondiente al individuo que muere al tiempo $t_{(i)}$ y $N(t_{(i)})$ denota el conjunto de individuos en riesgo al tiempo $t_{(i)}$. Las estimaciones de máxima verosimilitud de los parámetros $\boldsymbol{\beta}$ en el modelo de regresión de Cox se pueden encontrar al maximizar esta función de probabilidad logarítmica utilizando métodos numéricos. Esta maximización se logra generalmente utilizando el procedimiento de Newton-Raphson. Afortunadamente, la mayoría del software estadístico para el análisis de supervivencia permite el ajuste del modelo de regresión de Cox, por ejemplo R. Dicho software también proporciona los errores estándar de las estimaciones de parámetros en el modelo ajustado.

La verosimilitud parcial es útil, ya que resulta más simple hacer la estimación de $\boldsymbol{\beta}$ (parámetros de interés) pues no involucra el parámetro de estorbo $h_0(t)$. Con la verosimilitud parcial Cox proporcionó un procedimientos constructivo para encontrar probabilidades parciales útiles, resumió en una expresión toda o casi toda la información en la verosimilitud parcial y encontró que la expresión en (1.25) es una aproximación a la verosimilitud total y cumple con las mismas propiedades asintóticas que la verosimilitud total. Así que para hacer la estimación de la verosimilitud total del modelo de Cox, se hace uso de la verosimilitud parcial, en la cual el riesgo base se cancela y esta solo de los datos de censura.

CAPÍTULO 2

Fundamentos teóricos de redes bayesianas

Las redes bayesianas son modelos gráficos que representan las relaciones probabilísticas entre un número de variables. Constituyen un marco formal para la representación de decisiones bajo incertidumbre. Las redes bayesianas, que llevan el nombre de Thomas Bayes (1702-1761), surgieron de varias investigaciones matemáticas realizadas en la década de 1980, y en particular de trabajos sobre redes de creencias, redes causales y diagramas de influencia.

Estos modelos se propusieron por primera vez en la década de 1990 como *Probabilistic Expert Systems*, inspirada en el libro de Dechter & Pearl (1988) *Probabilistic Reasoning in Intelligent Systems*, quien fue pionero en el enfoque probabilístico de la inteligencia artificial y se reconoce como el fundador de las redes bayesianas. Durante las últimas décadas se ha trabajado mucho en materia de aprendizaje e inferencia con las redes bayesianas y sus posibles usos. En particular, se ha visto un aumento masivo en la aplicación de redes bayesianas a problemas del mundo real, incluidos el diagnóstico, el pronóstico, el control de fabricación,

la recuperación de información, la predicción e incluso la planificación. Casi todos los campos científicos y técnicos han visto el uso exitoso de estas redes como una herramienta para modelar las relaciones complejas entre un gran número de variables y para hacer inferencias. Mittal (2007) señala que las principales aplicaciones de redes bayesianas han sido en tecnologías de la información y la comunicación, biomedicina, genómica y bioinformática.

Las redes bayesianas son modelos robustos y existen muchas buenas razones para elegir redes probabilísticas como el marco de modelado, incluido el manejo coherente y matemáticamente sólido de la incertidumbre y la toma de decisiones, la construcción automatizada y la adaptación de modelos basados en datos, la representación intuitiva y compacta de las relaciones causa-efecto y las relaciones de dependencia condicional. Las redes bayesianas se utilizan comúnmente para representar relaciones causales. Sin embargo, las redes bayesianas son más generales.

En este capítulo presentaremos las ideas fundamentales detrás de las redes bayesianas y su interpretación básica. En este trabajo de investigación nos centraremos en el modelado de redes bayesianas discretas, aunque las definiciones principales correspondientes a redes bayesianas se dan de forma general. En la Sección 2.1 se introducen las redes bayesianas, sus principales usos y propiedades. Posteriormente en la Sección 2.2 se presenta la forma en que se construyen a partir de un conjunto de datos, así como la estimación de los parámetros. Finalmete, en la Sección 2.3 se menciona como las redes bayesianas son capaces de modelar variables observables y no observables.

2.1. Introducción redes bayesianas

Las redes bayesianas también se les conoce con el nombre de red de Bayes o red de creencia. Las redes bayesianas representan modelos de probabilidad conjunta entre variables dadas. Cada variable está representada por un vértice en un grafo. Las dependencias directas

entre las variables están representadas por aristas dirigidas entre los vértices correspondientes y las probabilidades condicionales para cada variable, es decir, los arcos dirigidos representan dependencias probabilísticas directas; por lo tanto, si no hay arco que conecte dos vértices, las variables correspondientes son independientes o condicionalmente independientes dado un subconjunto de las variables restantes. Para dar la definición formal de red bayesiana es necesario primero precisar algunos conceptos referente a teoría de grafos.

Definición 2.1.1. *Un grafo G es un par ordenado $G = (V, E)$ consiste en un conjunto finito de vértices V y un conjunto de aristas E que relaciona los vértices.*

En las redes bayesianas es importante la dirección en la cual se estable la relación entre las variables, siendo de suma importancia quien antecede en la relación de dependencias determinada por la arista. Un par ordenado $(u, v) \in E$ denota una arista dirigida desde el vértice u al vértice v, y se dice que u es un padre de v y v un hijo de u. El conjunto de padres de un vértice v se denotará por $pa(v)$. Así, es importante definir quien es el padre y quien es el hijo en una relación entre dos vértices.

Definición 2.1.2. *Un grafo dirigido es un grafo $G = (V, E)$ donde $V \neq \emptyset$ y $E \subseteq \{(a, b) \in V \times V : a \neq b\}$ es un conjunto de pares ordenados de elementos del conjunto de vértices V.*

De esta definición se tiene que el conjunto de aristas E es un par ordenado de vértices donde $(a, b) \neq (b, a)$ permitiendo establecer la relación padre-hijo.

Por otro lado en las redes bayesianas es importante no formar ciclos al momento de establecer la relación entre las variables de la red. Por tanto, se define un grafo acíclico, para ello es importante tener en cuenta que un *camino* es una secuencia de vértices distintos $v_1, \ldots, v_n$ tal que v_i se conecta con v_{i+1} para cada $i = 1, \ldots, n-1$; la longitud del camino es $n - 1$.

Definición 2.1.3. *Un grafo dirigido es acíclico (GAD) si no hay un camino dirigido $A_1 \to A_2 \to \cdots \to A_n$ tal que $A_1 = A_n$, con $A_1, \ldots, A_n \in V$.*

Ahora bien, con los conceptos definidos anteriormente se puede definir el soporte gráfico de una red bayesiana, sin embargo las redes bayesianas consta de dos partes, una parte cua-

litativa y una parte cuantitativa. La parte cualitativa corresponde al grafo acíclico dirigido y la parte cuantitativa corresponde a un conjunto de parámetros. La parte cualitativa de la red proporciona información acerca de la relación de dependencia entre variables, la forma en que se factoriza la distribución conjunta. Por otra parte la parte cuantitativa modela las distribuciones de las dependencias entre las variables.

Definición 2.1.4. *Una red bayesiana RB se define como el conjunto (G, Θ), donde $G = (V, E)$ es un grafo dirigido acíclico y Θ es un conjunto de parámetros.*

De la naturaleza de Θ se tiene que existen tres tipos de redes bayesianas: redes bayesianas discretas, continuas e híbridas. Sin importar el tipo de red bayesiana, la probabilidad conjunta sobre todas las variables de la red se puede calcular utilizando la regla de la cadena para redes bayesianas.

Teorema 2.1.1. ***Regla de la cadena para redes bayesianas***
Sea RB una red bayesiana con vértices $\boldsymbol{X} = \{X_1, \ldots, X_n\}$. Entonces RB especifica una distribución de probabilidad conjunta única $P(\boldsymbol{X})$ dada por el producto de todas las tablas de probabilidad condicional especificadas en RB

$$P(\boldsymbol{X}) = \prod_{i=1}^{n} P(X_i | pa(X_i); \boldsymbol{\Theta_i}) \quad \textit{con } X_i \in V, \textit{ para } i = 1, \ldots, n; \tag{2.1}$$

donde cada X_i ($i = 1, 2, \ldots, n$) es una variable aleatoria representada por un vértice del grafo, $\boldsymbol{\Theta_i} = \{\theta_{i,j}\}_{j \in pa(X_i)}$ modelan la probabilidad condicionada $P(X_i|pa(X_i))$ para $i = 1, 2, \ldots, n$ y $pa(X_i)$ representa es el conjunto de padres de X_i (es decir, los vértices apuntando directamente a X_i a través de una sola arista). También es común denotar $pa(X_i)$ como Π_{X_i}. La distribución de probabilidad multivariable de $\boldsymbol{X}$ se denomina distribución global de los datos, mientras que las univariadas asociadas con cada $X_i \in \boldsymbol{X}$ se denominan distribuciones locales.

La regla de la cadena para redes bayesianas permite conocer la estructura de una red, es decir, al conocer como se expresa la probabilidad conjunta de un conjunto finito de variables en una red bayesiana, se conoce quien antecede a cada vértice. En comparación con otros modelos donde no se conoce la estructura de dependencia entre las variables, es importante notar que las redes bayesianas reducen el número de parámetros a estimar para encontrar la distribución conjunta de las variables aleatorias. Siendo ésta una ventaja del uso de redes bayesianas para la modelación de un conjunto finito de variables aleatorias. El Ejemplo 2.1.5 muestra como se expresa la función de probabilidad conjunta usando el Teorema de la regla de la cadena.

Ejemplo 2.1.1. *Considere la red bayesiana de Scutari & Denis (2014) simple e hipotética cuyo objetivo es investigar los patrones de uso de diferentes medios de transporte de individuos en una población, con un enfoque en automóviles y trenes.*

- *Edad (A): la edad, registrada como joven (j) para personas menores de 30 años, adulto (a) para personas entre 30 y 60 años, y personas mayores v para personas mayores de 60 años.*

- *Sexo (S): el sexo biológico del individuo, registrado como masculino (m) o femenino (f).*

- *Educación (E): el nivel más alto de educación registrado, ya sea preparatoria ($prepa$) o el título universitario (uni).*

- *Ocupación (O): si la persona es un empleado (em) o un trabajador por cuenta propia (cp).*

- *Residencia (R): el tamaño de la ciudad en que vive el individuo, registrado como pequeña (pe) o grande (g).*

- *Transporte (T): el medio de transporte preferido por el individuo ya sea coche (c), tren (t) u otro (o).*

La Figura 2.1 muestra la red bayesiana anteriormente propuesta, donde del Teorema de la regla de la cadena se concluye que la función de distribución esta dada por:

$$P(A, S, E, O, R, T) \quad = \quad P(A)P(S)P(E|A, S)P(O|E)P(R|E)P(T|O, R).$$

Las Tablas 2.1, 2.2, 2.3, 2.4, 2.5 y 2.6 presenta las probabilidades condicionales de los vértice (A, E, R, O, S, T respectivamente) siguiendo la estructura dada por la red bayesiana de la Figura 2.1.

La independencia condicional es la noción probabilística clave en las redes bayesianas. Con el fin de establecer las independencias condicionales (y por defecto las dependencias condicionales) entre las variables en una red bayesiana, se hace uso de la definición de independencia condicional entre dos eventos.

Definición 2.1.5. ***Independencia condicional***

Los eventos A y B son condicionalmente independientes dado el evento C si y sólo si

$$P(A \cap B \mid C) \quad = \quad P(A \mid C)P(B \mid C). \tag{2.2}$$

La independencia condicional de A y B dado C es denotado como $(A \perp\!\!\!\perp B) \mid C$.

El vínculo entre la *separación gráfica* (independencia condicional en el grafo) se denota como $\perp\!\!\!\perp_G$ y está indicada por la ausencia de un arco. Por otro lado, la independencia probabilística se denota $\perp\!\!\!\perp_P$ proporciona una forma directa y fácilmente interpretable de expresar las relaciones entre las variables.

Para establecer criterios de separación gráfica es importante definir las conexiones fundamentales de una red bayesiana. Siguiendo el trabajo seminal de Dechter & Pearl (1988), distinguimos tres formas posibles de configurar tres vértices y dos aristas. De estas configuraciones posibles, la literatura define las conexiones fundamentales en una red bayesiana. Estas conexiones fundamentales son las siguientes

- Las estructuras de la forma $S \to E \to R$ en la Figura 2.1 se conocen como conexiones en serie, ya que ambos arcos tienen la misma dirección y siguen uno tras otro.

Edad	j	a	v
	0.30	0.50	0.20

Tabla 2.1: Probabilidad $P(A)$.

	Educación	
Edad y Sexo	$prepa$	uni
j & m	0.75	0.25
a & m	0.72	0.28
v & m	0.88	0.12
j & f	0.64	0.36
a & f	0.70	0.30
v & f	0.90	0.10

Tabla 2.2: Tabla de probabilidad condicional $P(E|A, S)$.

	Residencia	
Educación	pe	g
$prepa$	0.25	0.75
uni	0.2	0.8

Tabla 2.3: Tabla de probabilidad condicional $P(R|E)$.

	Ocupación	
Educación	em	cp
$prepa$	0.96	0.04
uni	0.92	0.08

Tabla 2.4: Tabla de probabilidad condicional $P(O|E)$.

Sexo	m	f
	0.30	0.20

Tabla 2.5: Probabilidad $P(S)$.

	Transporte		
Residencia y Ocupación	c	t	o
pe & em	0.48	0.42	0.10
pe & cp	0.56	0.36	0.08
g & em	0.58	0.24	0.18
g & cp	0.70	0.21	0.09

Tabla 2.6: Tabla de probabilidad condicional $P(T|O, R)$.

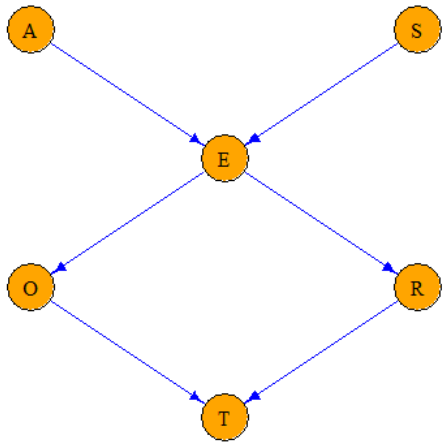


Figura 2.1: Red bayesiana.

- Estructuras como $R \leftarrow E \rightarrow O$ en la Figura 2.1 se conocen como conexiones divergentes, porque los dos arcos tienen direcciones divergentes desde un vértice central.
- Estructuras como $A \rightarrow E \leftarrow S$ en la Figura 2.1 se conocen como conexiones convergentes, porque los dos arcos convergen en un vértice central. Cuando no hay arco que une a los dos padres (es decir, ni $A \rightarrow S$ ni $A \leftarrow S$), las conexiones convergentes se denominan v *estructuras*.

Los tres casos anteriores cubren todas las formas de dependencia de tres variables, permitiendo definir un criterio de separación gráfica conocido como d*-separación*. Primero veamos como establecer la independencia condicional entre las variables de un grafo.

Definición 2.1.6. ***(Mapeos)*** *Sea M la estructura de dependencia de la distribución de probabilidad de $\boldsymbol{X}$, es decir, el conjunto de relaciones de independencia condicional que vincula cualquier terna A, B, C de subconjuntos de $\boldsymbol{X}$. Un grafo G es un mapeo de dependencia (D-mapeo) de M si hay una correspondencia uno a uno entre las variables aleatorias en $\boldsymbol{X}$ y los vértices V de G de tal manera que para todos los subconjuntos disjuntos A, B, C de $\boldsymbol{X}$ tenemos*

$$A \perp\!\!\!\perp_P B|C \implies A \perp\!\!\!\perp_G B|C. \tag{2.3}$$

De manera similar, G es un mapeo de independencia (o I-mapeo) de M si

$$A \perp\!\!\!\perp_P B|C \impliedby A \perp\!\!\!\perp_G B|C. \tag{2.4}$$

Se dice que G es un mapeo perfecto de M si es tanto un D-mapeo como un I-mapeo, es decir

$$A \perp\!\!\!\perp_P B|C \iff A \perp\!\!\!\perp_G B|C, \tag{2.5}$$

y en este caso se dice que G es isomorfo a M.

En el caso de un D-mapeo, la distribución de probabilidad de $\boldsymbol{X}$ determina qué arcos están presentes en el GAD G. Los vértices que están conectados en G corresponden a variables dependientes en $\boldsymbol{X}$; sin embargo, los vértices que están separados en G no necesariamente

corresponden a variables condicionalmente independientes en $\boldsymbol{X}$. Por otro lado, en el caso de un I-mapeo tenemos que los arcos presentes en el GAD G determinan qué variables son condicionalmente independientes en $\boldsymbol{X}$. Por lo tanto, los vértices que se encuentran separados en G corresponden a variables condicionalmente independientes en $\boldsymbol{X}$, pero los vértices que están conectados en G no necesariamente corresponden a variables dependientes en $\boldsymbol{X}$. En el caso de un mapeo perfecto, existe una correspondencia de uno a uno entre la separación gráfica en G y la independencia condicional en $\boldsymbol{X}$.

La separación gráfica se establece utilizando la d-separación, que se define formalmente a continuación. La d-separación es una regla que describe las relaciones entre dos nodos X e Y con respecto a otro nodo Z, es decir, X e Y están separados por Z si no hay información entre ellos cuando se observa Z. El Ejemplo 2.1.2 muestra como se relacionan los conceptos de *Mapeos* y la d-separación.

Definición 2.1.7. ***(d-separación)*** *Sean X, Y y Z tres subconjuntos disjuntos de vértices en un grafo dirigido acíclico D; entonces se dice que Z d-separa a X e Y si y sólo si a lo largo de todo camino no dirigido entre cualquier vértice de X y cualquier vértice de Y existe un vértice intermedio v tal que,*

1. *v es un vértice de aristas convergentes en el camino y ni v ni sus descendientes están en Z, o bien*

2. *v no es un vértice de aristas convergentes en el camino y v está en Z.*

Ejemplo 2.1.2. *Considere las tres conexiones fundamentales ($S \rightarrow E \rightarrow R$, $O \leftarrow E \rightarrow R$ y $A \rightarrow E \leftarrow R$) que se muestran en la red bayesiana 2.1. La primera conexión es en serie, y estamos investigando si $S \perp\!\!\!\perp_G R|E$. El nodo E, desempeña el papel de v en la Definición 2.1.7, coincide con la segunda condición y d-separa a S y R. Como resultado, podemos concluir que $S \perp\!\!\!\perp_G R|E$ se mantiene y, a su vez, podemos determinar que S y R son condicionalmente independientes $S \perp\!\!\!\perp_P R|E$ utilizando la Definición 2.1.6. Un razonamiento idéntico lleva a la conclusión de que $O \perp\!\!\!\perp_G R|E$ y $O \perp\!\!\!\perp_P R|E$ es válido para la conexión*

divergente formada por E, O y R. Por otro lado, en la conexión convergente formada por A, S y E tenemos que $A \not\perp\!\!\!\perp_G S|E$. A diferencia de las conexiones en serie y divergentes, el nodo en el medio de la conexión no separa d las otras dos, ya que E no coincide con ninguna de las dos condiciones en la Definición 2.1.7.

Además las conexiones fundamentales permiten definir lo que se entiende por grafos equivalentes. Para ésto se define el esqueleto de un grafo acíclico dirigido.

Definición 2.1.8. *El esqueleto de un grafo acíclico dirigido es el grafo no dirigido resultante de eliminar la dirección de todos sus aristas.*

Dada la definición de un esqueleto, se establece la equivalencia de dos GADs mediante la siguiente definición.

Definición 2.1.9. ***Clases equivalentes***
Dos GAD definidos sobre el mismo conjunto de variables son equivalentes si y sólo si tienen el mismo esqueleto y las mismas v estructuras.

Para indagar sobre la estructura de un grafo acíclico dirigido uno se interesa en las aristas que forman parte de una o más v *estructuras*. Esto con el fin de de que el grafo no contenga ciclos y el soporte de la red bayesiana siga siendo un GAD. Esto es, la mayoría de los métodos dedicados a investigar la estructura de red bayesiana están interesados en indagar las conexiones fundamentales entre nodos para determinar la independencia entre variables a través de la d-separación.

Como puede observarse, las redes bayesianas son redes probabilísticas con representaciones ideales de conocimiento para su uso en muchas situaciones que involucran razonamiento. Su uso se puede extender a la toma de decisiones bajo incertidumbre. Las opciones de decisión y las utilidades asociadas con estas opciones pueden incorporarse explícitamente en el modelo, en cuyo caso el modelo se convierte en un diagrama de influencia que se utilizan para calcular las utilidades esperadas de todas las opciones de decisión, dada la información conocida en el momento de la decisión. Así, para Kjaerulff & Madsen (2008) los diagramas de influencia

son redes bayesianas donde se agregan variables de decisión y son aplicables para una amplia gama de áreas de dominio con incertidumbre inherente.

Por otro lado, dentro de las aplicaciones de redes bayesianas, se sabe que si se tiene un modelo de red bayesiana, éste puede usarse para la tarea de clasificación. Dado que los clasificadores Naive Bayes y Tree Augmented Naive Bayes son fáciles de aprender y ampliamente usados, y dado que son muy flexibles con respecto a los valores perdidos, estos son usados para el aprendizaje de estructura de red bayesiana a través de datos (ver Sección 2.2).

2.2. Construcción y selección de redes bayesianas

Las redes bayesianas son estructuras gráficas que representan las relaciones probabilísticas entre un gran número de variables. Durante la década de 1980, se realizaron muchas investigaciones relacionadas con el desarrollo de redes bayesianas para redes causales, algoritmos para realizar inferencias con ellas y aplicaciones. En la década de 1990 surgieron excelentes algoritmos para el aprendizaje de redes bayesianas a partir de los datos. Sin embargo, para el año 2000 todavía no parecía haber una fuente accesible para "aprender redes bayesianas". Al día de hoy la literatura recopila muchos algoritmos para aprender las redes bayesianas a partir de datos, para hacer inferencias en redes bayesianas y diagramas de influencia, sin embargo estas tareas aún resultan ser complicadas.

La causalidad juega un papel importante en el proceso de construcción de modelos de red probabilísticos, sin embargo las redes bayesianas son más generales. La idea principal de la causalidad es suponer la ocurrencia de algún evento c que causa el efecto e y se sabe que la relación entre c y e esta determinada. Entonces, obviamente, observando c se puede concluir e. Observar e, por otro lado, no permite concluir c, a menos que se sepa que c es la única causa de e. La relación entre el valor tomado por una variable y los valores tomados por sus predecesores se especifica mediante una distribución de probabilidad condicional. Cuando se

proporciona una estructura gráfica y los supuestos de modelado permiten una interpretación causal, entonces las estimaciones de las tablas de probabilidad condicional obtenidas de los datos pueden usarse para inferir un sistema de causalidad a partir de un conjunto de distribuciones de probabilidad condicional.

Aprender el GAD de una red bayesiana es una tarea compleja, por dos razones. En primer lugar, el espacio de los posibles GAD es muy grande; el número de GAD aumenta exponencialmente a medida que crece el número de vértices. Como resultado, sólo una pequeña fracción de sus elementos se puede investigar en un tiempo razonable. Además, este espacio es muy diferente de los espacios reales, este espacio no es continuo y tiene un número finito de elementos. Por lo tanto, se requieren algoritmos para explorarlo.

Una red probabilística se puede construir *manualmente*, *semi-automáticamente* a partir de datos, o mediante una *combinación de los dos métodos* anteriores. Como se comenta anteriormente, una red probabilística consta de dos componentes, estructura y parámetros.

La construcción manual de redes probabilísticas puede ser una tarea difícil y puede requerir de mucho tiempo, por lo cual se recomienda construir la red a través del aprendizaje automatizado. En el campo de las redes bayesianas, la selección y estimación de modelos se conocen colectivamente como aprendizaje, un nombre tomado de inteligencia artificial y aprendizaje automático. En Scutari & Denis (2014) el aprendizaje de la red bayesiana se realiza generalmente como un proceso de dos pasos:

1. Aprendizaje de la estructura del GAD;

2. Aprendizaje de los parámetros dada la estructura del GAD aprendida en el paso anterior.

Ambos pasos pueden realizarse como aprendizaje no supervisado, utilizando la información proporcionada por un conjunto de datos, o como aprendizaje supervisado, entrevistando a

expertos en los campos relevantes para el fenómeno que se está modelando. A menudo, la información previa disponible sobre el fenómeno no es suficiente para que un experto especifique completamente una red bayesiana. Incluso especificar la estructura GAD es a menudo imposible, especialmente cuando se trata de un gran número de variables.

Sea D un conjunto de datos (con p observaciones) y $G = (V, E)$ un GAD. Si se denota con $\boldsymbol{\Theta}$ los parámetros de la distribución conjunta de $\boldsymbol{X} = \{X_1, \ldots, X_n\}$ con $X_1, \ldots, X_n \in V$, entonces $RB = (G, \boldsymbol{\Theta})$ es una red bayesiana y el aprendizaje de la RB puede formalizarse como

$$\underbrace{P(G, \boldsymbol{\Theta}|D)}_{\text{Aprendizaje}} = \underbrace{P(G|D)}_{\text{Aprendizaje-Estructura}} \times \underbrace{P(\boldsymbol{\Theta}|G, D)}_{\text{Aprendizaje-Parámetros}} . \tag{2.6}$$

La descomposición de $P(G, \boldsymbol{\Theta}|D)$ en (2.6) refleja los dos pasos descritos anteriormente, y muestra la lógica detrás del proceso de aprendizaje. El aprendizaje de la estructura se puede hacer encontrando el GAD G que maximiza

$$P(G|D) \propto P(G)P(D|G) = P(G) \int P(D|G, \boldsymbol{\Theta})P(\boldsymbol{\Theta}|G)d\boldsymbol{\Theta}$$

utilizando el teorema de Bayes para descomponer la probabilidad posterior $P(G|D)$ en el producto de la distribución *a priori* sobre los posibles grafos acíclicos $P(G)$ y la probabilidad de los datos $P(D|G)$. Claramente, no es posible calcular este último sin estimar también los parámetros $\boldsymbol{\Theta}$ de G; por lo tanto, $\boldsymbol{\Theta}$ debe integrarse para que $P(G|D)$ sea independiente de cualquier elección específica de $\boldsymbol{\Theta}$. La distribución previa $P(G)$ proporciona una manera ideal de introducir cualquier información previa disponible sobre las relaciones de independencia condicional entre las variables en $\boldsymbol{X}$. La opción más común para $P(G)$ es $P(G) \propto 1$ la cual es una previa no informativa sobre el espacio de los posibles GADs, asignando la misma probabilidad a cada GAD.

El obtener $P(D|G)$ también es una problemática desde un punto de vista computacional como algebraico, por lo que se hace uso de estadística bayesiana para hacer la estimación

de $P(D|G)$. A partir de la descomposición en distribuciones locales, podemos factorizar aún más $P(D|G)$ de la siguiente manera

$$\begin{aligned} P(D|G) &= \int \prod_{i=1}^{n} [P(X_i|pa(X_i), \mathbf{\Theta}_{X_i}) P(\mathbf{\Theta}_{X_i}|pa(X_i))] \, d\mathbf{\Theta} \\ &= \prod_{i=1}^{n} \left[\int P(X_i|pa(X_i), \mathbf{\Theta}_{X_i}) P(\mathbf{\Theta}_{X_i}|pa(X_i)) d\mathbf{\Theta}_{X_i} \right] \\ &= \prod_{i=1}^{n} E_{\mathbf{\Theta}_{X_i}} P(X_i|pa(X_i)). \end{aligned}$$

Si todos los valores esperados de la ecuación anterior se pueden calcular, entonces $P(D|G)$ se puede calcular en un tiempo razonable incluso para grandes conjuntos de datos. Por ejemplo lo anterior es posible tanto para la distribución multinomial asumida para las redes bayesianas discretas (a través de su posterior conjugada Dirichlet) como para la distribución multivariada gaussiana asumida para redes bayesianas continuas (mediante su distribución conjugada Wishart-Inversa). Sin embargo en este trabajo de investigación estamos interesados en redes discretas, por lo que se asume una distribución multinomial para las redes bayesianas, así que $P(D|G)$ se puede estimarse mediante la función de puntuación *Bayesian Dirichlet equivalent uniform* (BDeu) de Heckerman et al. (1995). Comúnmente a BDeu se le denomina simplemente BDe. Esta puntuación asume una *a priori* plana sobre el espacio de los GAD y el espacio de parámetros de cada nodo, es decir:

$$P(G) \propto 1 \qquad \text{y} \qquad P(\mathbf{\Theta}_i|pa(X_i)) = \alpha_{ij} = \frac{\alpha}{|\mathbf{\Theta}_i|} \quad j \in pa(X_i),$$

donde $\alpha_{i,j}$ el parámetro j de la distribución Dirichlet para la variable X_i, esto es, se cumple que $\sum_{j \in pa(X_i)} \alpha_{i,j} = 1$. Así, el único parámetro de BDe es el tamaño de muestra imaginario α asociado con la previa Dirichlet que, su expresión es complicada (ver Heckerman et al.) por lo que no se reporta aquí, determina la cantidad de peso que se asigna a la distribución previa (como el tamaño de una muestra imaginaria). Bajo estas suposiciones, BDe toma la

forma

$$\begin{aligned} \text{BDe}(G;D) &= \prod_{i=1}^{n} \text{BDe}(X_i, pa(X_i)) \\ &= \prod_{i=1}^{n} \prod_{j=1}^{q_i} \frac{\Gamma(\alpha_{ij})}{\Gamma(\alpha_{ij}+N_{ij})} \prod_{k=1}^{r_i} \frac{\Gamma(\alpha_{ijk}+N_{ijk})}{\Gamma(\alpha_{ijk})} \end{aligned}$$

donde n es el número de vértices en G; r_i es el número de categorías para el nodo X_i; q_i es el número de configuraciones de las categorías de los padres de X_i; N_{ijk} es el número de muestras que tienen la categoría j para el nodo X_i y la k configuración de sus padres.

Como resultado de las dificultades descritas anteriormente, se han desarrollado dos alternativas al uso de Pr (D — G) en el aprendizaje de estructuras.Se han desarrollado dos alternativas al uso de $P(D|G)$ en el aprendizaje de estructuras. *La primera* es el uso del criterio de información bayesiano (*Bayesian Information criterion*, BIC) como una aproximación de $P(D|G)$. Esta aproximación fue hecha por Schwarzet al. (1978), donde mostró lo siguiente

$$BIC(G|D) \rightarrow \log \text{BDe}(G,D) \quad \text{cuando } n \rightarrow \infty.$$

De lo anterior, se tiene que BIC puede usarse como una aproximación de $P(D|G)$. Ademas BIC se puede descomponer en términos de la verosimilitud.

$$BIC(G|D) = \sum_{i=1}^{p} \left[\log P(X_i|pa(X_i)) - \frac{\mathbf{\Theta}_i}{2} \log n \right]$$

lo que hace que sea muy fácil de calcular. La *segunda alternativa* es evitar la necesidad de definir una medida de bondad de ajuste para el GAD y usar pruebas de independencia condicional para aprender la estructura de un GAD de un arco a la vez. Este procedimiento es más común en el caso de redes continuas como las Gaussianas.

Las dos alternativas mencionadas previamente dan lugar a los principales métodos de aprendizaje de la estructura de red la bayesiana descritos en la siguiente sección.

2.2.1. Aprendizaje de la estructura

Todos los métodos de aprendizaje de la estructura se reducen a tres enfoques. Estos enfoques se basan en

1. **Restricciones**: estos algoritmos aprenden la estructura de la red analizando las relaciones probabilísticas con pruebas de independencia condicional y luego construyen un gráfico que satisfaga la correspondiente d-separación. Los modelos resultantes a menudo se interpretan como modelos causales incluso cuando se aprenden de datos observacionales.

2. **Puntuación**: estos algoritmos asignan una puntuación a cada red bayesiana candidata e intentan maximizarla con algún algoritmo de búsqueda heurística. Los algoritmos de búsqueda codiciosos como el de hill climbing o el de búsqueda tabú son una opción común, pero se puede utilizar casi cualquier tipo de procedimiento de búsqueda.

3. **Híbridos**: combina algoritmos basados en restricciones y basados en puntajes para compensar las debilidades respectivas. Se componen principalmente de dos pasos:

   - Restringir: se utilizan algoritmos basados en restricciones para reducir el conjunto de GAD candidatos.
   - Maximizar: se utilizan algoritmos basados en la puntuación para encontrar un GAD óptimo del conjunto reducido.

Dentro de las aplicaciones de redes bayesianas esta la tarea de clasificación, por lo que es importante mencionar que el estudiar la estructura de red bayesiana muchas veces es posible al ver las redes bayesianas como clasificadores (ver (Nielsen & Jensen (2009))). Entre los modelos de redes bayesianas vistos como clasificadores destacan: *Naive Bayes* y *Tree Augmented Naive Bayes*. Para hacer aprendizaje de la estructura de la red bayesiana visto como clasificador se usan las tres métodos de aprendizaje anteriormente mencionados.

En el clasificador ingenuo de Bayes (Naive Bayes) visto como red bayesiana, cada variable tiene un único padre, donde el padre representa una clase. Esto significa que la estructura es fija, y la única tarea involucrada en el aprendizaje es estimar los parámetros. La característica importante del modelo Naive Bayes es que tiene supuestos de independencia muy fuertes. El supuesto de independencia en este modelo es que, dada una variable que representa una clase, todas las variables aleatorias son independientes entre sí. Pero esta suposición rara vez resulta ser cierta en la realidad. Además, este método es manejable solo para conjuntos de configuración pequeña.

Para un mejor rendimiento de clasificación, y al necesitar una red bayesiana que codifique la estructura del modelo Naive Bayes y también capture las correlaciones entre las variables en el sistema, surgen las redes bayesianas ingenuas aumentadas (Augmented Naive Bayes). Una red bayesiana ingenua aumentada, mantiene la estructura de la red bayesiana ingenua y la aumenta agregando aristas entre las variables para capturar las correlaciones entre los atributos. Evidentemente este proceso aumenta la complejidad computacional pero para reducir la complejidad computacional y también tener en cuenta las correlaciones entre las variables, se imponen restricciones al nivel de interacción entre las variables. Uno de estos modelos, es el modelo de árbol aumentado del clasificador ingenuo de Bayes (Tree Augmented Naive Bayes). En éste se impone una restricción en el nivel de interacción entre muchas las variables a una. Todas las variables están conectadas a las variables de clase por medio de aristas. Además de éso, cada variable se puede conectar a otra variable en la red. Es decir, cada variable en el gráfico puede tener dos padres, a saber, el nodo de clase y otro nodo de variable, excepto una variable que se llama raíz. La complejidad computacional se reduce considerablemente, ya que cada variable tiene un máximo de dos padres. Por lo tanto, el árbol aumentado de bayes mantiene la robustez y la complejidad computacional del modelo Naive Bayes y al mismo tiempo muestra una mayor precisión.

Dada las dificultades del aprendizaje de la red bayesiana vista como clasificador han surgido

otros modelos para la tarea de indagar la estructura de red bayesiana. De lo anterior, varios otros modelos de aprendizaje se han implementado en programas computacioneles. En el paquete *bnlearn* de Scutari (2009),del software R, existe una gran variedad de los métodos de aprendizaje (del tipo restricciones, puntuación e híbridos) de la estructura de la red bayesiana.

Los algoritmos de aprendizaje basados en restricciones disponibles en *bnlearn* son

- PC (pc.stable): una implementación moderna del primer algoritmo práctico de aprendizaje de estructura basado en restricciones.
- Grow-Shrink (gs): basado en Grow-Shrink Markov Blanket, el primer algoritmo de detección de manto de Markov utilizado en un algoritmo de aprendizaje de estructuras.
- Incremental Association (iamb): basada en el algoritmo de manto de Markov, que se basa en un esquema de selección de dos fases.
- Fast Incremental Association (fast.iamb): una variante de IAMB para reducir el número de pruebas de independencia condicionales.
- Asociación incremental intercalada (inter.iamb): otra variante de IAMB.

La complejidad computacional de estos algoritmos es polinomial en el número de pruebas, generalmente $O(N^2)$, donde N es el número de variables. El tiempo de ejecución se escala linealmente con el tamaño del conjunto de datos.

Por otra parte algunos algoritmos de aprendizaje basados en puntuación disponibles en *bnlearn* son

- Hill-Climbing (hc): una búsqueda codiciosa de los gráficos dirigidos. La implementación optimizada utiliza el almacenamiento en caché de puntuación, la capacidad de descomposición y la equivalencia de puntuación para reducir el número de pruebas duplicadas.

- Búsqueda de tabú (tabu): una modificación de (hc) capaz de escapar de los óptimos locales seleccionando una red que mínimamente disminuye la función de puntuación.

También para algoritmos de aprendizaje de estructura basado en híbridos se cuenta con una gran variedad de algoritmos y métodos de aprendizaje estructural de la red. Los algoritmos de aprendizaje híbrido disponibles en el paquete *bnlearn* son:

- Max-Min Hill-Climbing (mmhc): un algoritmo híbrido que combina el algoritmo Max-Min Parents and Children (algoritmo perteneciente al enfoque de restricciones que sirve para restringir el espacio de búsqueda) y el algoritmo de Hill-Climbing (para encontrar la estructura de red óptima en el espacio restringido).

- Maximización restringida (rsmax2): una implementación más general de Max-Min Hill-Climbing, que puede usar cualquier combinación de algoritmos basados en restricciones y basados en puntajes (tabu y hc).

Muchas métricas de puntuación han sido propuestas con el fin de encontrar la estructura optima de acuerdo a la puntuación a optimizar. Las más utilizadas en la literatura son la puntuaciones $K2$ y BDe, las cuales dado un GAD G y un conjunto de datos D, se definen respectivamente como

$$\begin{aligned} K2(G;D) &= \prod_{i=1}^{n}\prod_{j=1}^{q_i}\frac{(r_i-1)!}{(N_{ij}+r_i-1)!}\prod_{k=1}^{r_i}N_{ijk}! \\ \text{BDe}(G;D) &= \prod_{i=1}^{n}\prod_{j=1}^{q_i}\frac{\Gamma(N'_{ij})}{\Gamma(N'_{ij}+N_{ij})}\prod_{k=1}^{r_i}\frac{\Gamma(N'_{ijk}+N_{ijk})}{\Gamma(N'_{ijk})} \end{aligned}$$

donde para cada variable X_i, r_i es la cardinalidad del dominio de X_i, q_i es el número de posibles combinaciones de $pa(X_i)$, y N_{ijk} el número de veces de los datos donde X_i toma el k–ésimo valor y $j \in pa(X_i)$. Además, el parámetro N'_{ijk} para cada tripleta i, j, k son fijadas como $N'/q_i r_i$, donde N' es llamada muestra de tamaño equivalente (comúnmente $N' = 1$). Cabe mencionar que estas métricas ya están implementadas en el paquete *bnlearn* de R.

Los algoritmos basados en puntuación por otro lado, son simplemente aplicaciones de varios algoritmos de búsqueda heurística de propósito general, tales como hill-climbing, tabu search, simulated annealing (recocido simulado) y varios *algoritmos genéticos*. Muchos algoritmos genéticos han sido propuestos para el aprendizaje de la estructura de redes bayesianas, sin embargo ninguno de estos algoritmos genéticos han sido implementados en el paquete *bnlearn* de R.

2.2.2. Aprendizaje de los parámetros

Una vez que la estructura de la red bayesiana se ha aprendido de los datos, la tarea de estimar y actualizar los parámetros de la distribución global (distribución conjunta) se simplifica en gran medida mediante la descomposición en distribuciones locales. Dos enfoques son comunes en la literatura: estimación de máxima verosimilitud y estimación bayesiana. Sin embargo, a pesar de que las distribuciones locales en la práctica involucran sólo un pequeño número de variables, su dimensión por lo general no se ajusta al tamaño de la red bayesiana, la estimación de parámetros es problema en algunas situaciones, por ejemplo, es común tener tamaños de muestra mucho más pequeños que el número de variables incluidas en el modelo. Cabe mencionar que en el enfoque de estimación bayesiana se puede recurrir a MCMC para la estimación de las distribuciones posteriores, ya que éste permite calcular modelos con gran cantidad de parámetros desconocidos.

Máxima Verosimilitud

Para el caso de redes bayesianas discretas, es necesario estimar las probabilidades condicionadas a los padres de cada vértice. Éstas probabilidades son comúnmente presentadas en tablas. Así, en las redes bayesianas discretas los parámetros a estimar son las probabilidades condicionales en las distribuciones locales. Éstas se pueden estimar con las frecuencias empíricas correspondientes en el conjunto de datos.

Supongamos la red bayesiana del Ejemplo 2.1.1 donde se establece que las variables E y O

están relacionadas de tal forma que $E \rightarrow O$ (es decir E es padre de O). Los posibles valores de $E = \{prepa,\ uni\}$ y de $O = \{em,\ cp\}$, por lo que las frecuencias empíricas (de la base de datos) correspondientes son:

$$\begin{aligned} \widehat{P}(O = em|E = prepa) &= \frac{\widehat{P}(O = em, E = prepa)}{\widehat{P}(E = prepa)} \\ &= \frac{\text{número de observaciones donde } O = em \text{ y } E = prepa}{\text{número de observaciones donde } E = prepa}. \end{aligned}$$

De igual manera se procede para estimar combinaciones restantes, es decir, $\widehat{P}(O = em|E = uni)$, $\widehat{P}(O = cp|E = prepa)$ y $\widehat{P}(O = cp|E = uni)$. Lo anterior da lugar a la estimación frecuentista es decir vía *Máxima Verosimilitud*.

Estadística bayesiana

Siguiendo el enfoque bayesiano descrito en la ecuación (2.6), esto requeriría encontrar el valor de $\boldsymbol{\Theta}$ que maximiza $P(\boldsymbol{\Theta}|G, D)$ a través de sus componentes $P(\boldsymbol{\Theta}_{X_i}|X_i, pa(X_i))$.

Las distribuciones locales, en la práctica, involucran sólo un pequeño número de vértices, es decir, X_i y sus padres $pa(X_i)$. Además, su dimensión generalmente no se escala con el número de vértices en la red bayesiana (y, a menudo, se supone que está limitada por una constante al calcular la complejidad computacional de los algoritmos), evitando así el problema de dimensionalidad. Esto significa que cada distribución local tiene un número comparativamente pequeño de parámetros para estimar a partir de la muestra, y que las estimaciones son más precisas debido a la mejor relación entre el tamaño de $\boldsymbol{\Theta}_i$ y el tamaño de la muestra.

En el caso de redes bayesianas discretas, las probabilidades posteriores estimadas se calculan a partir de una distribución previa uniforme sobre cada tabla de probabilidad condicional. Para determinar cuánto peso se asigna a la distribución previa en comparación con los datos cuando se calcula la distribución posterior, se agrega un tamaño de muestra imaginario iss (también conocido como tamaño de muestra equivalente). Su valor se divide por el número de celdas en la tabla de probabilidad condicional (porque la distribución previa es plana) y

se usa para calcular la estimación posterior como una media ponderada con las frecuencias empíricas.

Consideremos nuevamente el Ejemplo 2.1.1 de red bayesiana. Supongamos que se tiene un tamaño de muestra n. Luego

$$\begin{aligned} \widehat{P}_{em,\ prepa} &= \frac{\text{número de observaciones en las que } O = em \text{ y } E = prepa}{n}, \\ \hat{P}_{prepa} &= \frac{\text{número de observaciones en las que } E = prepa}{n}, \end{aligned}$$

y denotamos las probabilidades previas correspondientes como

$$\pi_{em,\ prepa} = \frac{1}{nO \times nE} \qquad \text{y} \qquad \pi_{prepa} = \frac{nO}{nO \times nE}$$

donde nO es el número de niveles de O y nE el número de niveles de E, así tenemos que

$$\begin{aligned} \widehat{P}(O = em|E = prepa) &= \frac{iss}{n + iss}\pi_{em,\ prepa} + \frac{n}{n + iss}\widehat{P}_{emp,\ prepa} \\ \widehat{P}(E = prepa) &= \frac{iss}{n + iss}\pi_{prepa} + \frac{n}{n + iss}\widehat{P}_{prepa} \end{aligned}$$

y por lo tanto se tiene que

$$\widehat{P}(O = em|E = prepa) = \frac{\widehat{P}(O = em, E = prepa)}{\widehat{P}(E = prepa)}$$

El valor de iss generalmente se elige para que sea pequeño, generalmente entre 1 y 15 (ver Scutari & Denis (2014)), para permitir que la distribución previa sea fácilmente dominada por los datos. Estos valores tan pequeños dan como resultado probabilidades condicionales que son más suaves pero aún así cercanas a las frecuencias empíricas desde las que se calculan.

2.3. Datos incompletos

En la Sección 2.2.2 vimos cómo los parámetros de probabilidad en una red bayesiana pueden estimarse a partir de un conjunto de datos completos, es decir, un conjunto de datos en el que cada caso especifica un valor para cada una de las variables. En la práctica, sin embargo, frecuentemente nos enfrentamos a situaciones en las que los datos están incompletos. Por ejemplo, debido a lecturas defectuosas, algunos valores pueden haber sido eliminados intencionalmente o algunas variables simplemente no pueden ser observables (estas se denominan variables latentes). Si sólo en algunos de los casos de la base de datos contienen datos faltantes, entonces uno podría considerar simplemente desechar estos casos y estimar los parámetros utilizando la base de datos restante (completa). Sin embargo, para los estadísticos es obvio que este enfoque puede tener un serio inconveniente, además del riesgo de terminar con una base de datos muy pequeña, podemos sesgar involuntariamente las estimaciones de los parámetros.

Uno de los algoritmos más populares para realizar la estimación de parámetros en el caso de tener parámetros de estorbo o variables latentes, es el algoritmo de *Expectation-Maximization* (EM). El algoritmo EM es un algoritmo general para encontrar estimaciones de máxima verosimilitud para un conjunto de parámetros cuando uno se enfrenta a un conjunto de datos incompletos.

En referencia a la problemática de trabajar con datos faltantes se puede consultar Štajduhar et al. (2009). Es importante dejar en claro que en este trabajo solo se consideraron datos completos.

CAPÍTULO 3

Redes bayesianas para tiempos de falla

De acuerdo a Kraisangka & Druzdzel (2018) las redes bayesianas se han convertido en un enfoque alternativo para el análisis de supervivencia. Estas tienen múltiples aplicaciones y en contraste con otras metodologías de Machine Learning, como redes neuronales, permiten conocer la estructura de relación entre las variables. Kraisangka & Druzdzel proponen métodos de selección y ajuste de redes bayesianas en el contexto de supervivencia, sin embargo el modelo con mejor desempeño no considera la dependencia entre covariables. Con el fin de extender la idea de este artículo se propone hacer el aprendizaje automatizado a partir de los datos para la red bayesiana.

Recordemos que la mayoría de los métodos para aprender estructuras de red bayesiana a partir de datos (ver Sección 2.2.1) se basan en funciones de puntuación, en restricciones o híbridos. Los métodos basados en la puntuación buscan la estructura de modelo que mejor se adapte a los datos mediante la introducción de una función de puntuación que evalúa a cada modelo candidato con respecto a los datos. Los métodos basados en restricciones, por otro

lado, usan declaraciones de independencia condicionales (restricciones) que se determinan mediante pruebas estadísticas en los datos. En los últimos años han surgido varios trabajos de investigación que analizan el problema del uso de diferentes técnicas de Machine Learning para aprender de los datos de supervivencia censurados. Sólo unos pocos consideran usar redes bayesianas para modelar la supervivencia.

Una aplicación de redes bayesianas para el análisis de supervivencia encontrada en la literatura y relacionada con este trabajo de investigación es Štajduhar & Dalbelo-Bašić (2010), donde los autores, hacen aprendizaje de red bayesiana mediante dos procedimientos ya conocidos en la literatura, el algoritmo de puntuación Hill-Climbing y un algoritmo de independencia condicional basado en restricciones. El método propuesto se probó exhaustivamente en un estudio de simulación y en el conjunto de datos clínicos GBSG2. Éste método también se comparó con el aprendizaje de redes bayesianas al tratar datos censurados con la regresión de Cox. Las pruebas realizadas en el conjunto de datos GBSG2 sugieren que solo debe hacerse el aprendizaje de parámetros y no hacer aprendizaje de la estructura de red bayesiana, ya que al aprender la estructura de red bayesiana, los resultados empeoran ligeramente, con respecto al modelo de riesgos proporcionales de Cox.

Existen otros trabajos que han incluido redes bayesianas en el contexto de análisis de supervivencia. En Landoni et al. (2013) y Zangrillo et al. (2015) estudian si el tipo de anestésico puede influir en la supervivencia de los pacientes después de una cirugía cardíaca. Los autores usan una red bayesiana para comparar el efecto sobre la mortalidad de diferentes anestésicos. En Gerstung et al. (2009) los autores presentan un modelo de red bayesiana para modelar la progresión del cáncer. Los parámetros del modelo se estiman mediante un algoritmo de Expectación-Maximización y la estructura de la red bayesiana se obtiene mediante un procedimiento de recocido simulado. Štajduhar et al. (2009) analiza la influencia de la censura con una simulación en datos sintéticos muestreados de redes bayesianas para modelos de supervivencia. Para ésto utilizan dos métodos para aprender redes bayesianas, uno basado en

restricciones y otro basado en puntuación. En Donat et al. (2010) los autores proponen una red bayesiana dinámica, denominada modelo de duración gráfica (GDM) y tienen como objetivo representar una amplia gama de modelos de duración. Como aplicación de este trabajo se ilustra un estudio de análisis de supervivencia en el que el modelo propuesto se compara con modelos de cadenas de Markov.

Cabe resaltar que otros trabajos han buscado aplicar modelos de redes neuronales en problemas de análisis de supervivencia. Entre éstos destacan Burke et al. (1997) en donde los autores comparan la precisión de predicción de cáncer del sistema de estadificación TNM con la de los modelos de redes neuronales artificiales. El sistema de estadificación TNM se originó como una respuesta a la necesidad de un sistema de predicción de resultados del cáncer exacto, consistente y universal. En Bakker et al. (2004) muestran que el análisis tradicional de supervivencia de Cox puede mejorarse cuando se complementa con antecedentes razonables y se analiza con una red neuronal, esto al producir mejores resultados predictivos en la función de supervivencia. Eleuteri et al. (2003) presentan la arquitectura de una red neuronal orientada a la estimación de la probabilidad de supervivencia. La red crea una aproximación a la probabilidad de supervivencia de un sistema en un momento dado, condicionada a las características del sistema. A partir de los requisitos sobre la función de supervivencia definen una red neuronal que satisface esos requisitos con la elección adecuada de activaciones de unidades ocultas y restricciones en el espacio de peso. Los experimentos con datos de supervivencia sintéticos y reales demuestran que la red neuronal puede aproximar funciones de supervivencia complejas.

Aunque los modelos de machine learning son atractivos para modelar sistemas complejos, las redes bayesianas permiten modelar y conocer la relación entre las variables, lo que es muy atractivo para poder introducir información experta o extraer información sobre las relaciones entre variables.

En este capítulo se presenta un algoritmo de evolución diferencial para resolver el proble-

ma de aprendizaje de la estructura de una red bayesiana para modelar tiempos de falla o supervivencia. La evolución diferencial (ED) es ampliamente adoptada en problemas de optimización debido a su capacidad de auto-adaptación de la búsqueda al panorama de aptitud física en cuestión. Aunque originalmente se propuso ED para problemas continuos, en de artículos anteriores, Baioletti et al. (2018) introduce un marco algebraico que permite aplicar ED a problemas combinatorios en los que el espacio de búsqueda es un grupo finamente generado.

En la Sección 3.1 se describe el modelo propuesto en este trabajo de investigación. Ésta sección se divide en dos subsecciones, en donde respectivamente se expone la propuesta de selección de covariables de supervivencia y se describe el marco algebraico para redes bayesianas que permiten desarrollar la ED. Posteriormente en la Sección 3.2 se presentan los resultados experimentales obtenidos para datos sintéticos utilizando la metodología desarrollada y otras ya establecidas, con el fin de contrastarlas.

3.1. Descripción del modelo propuesto

El modelo propuesto tiene como objetivo estudiar tiempos de falla mediante redes bayesianas. La idea de trabajar tiempos de falla mediante redes bayesianas, para este trabajo de tesis, surgió con el trabajo desarrollado en Kraisangka & Druzdzel (2018) el cual es descrito en la Subsección 3.1.1. En este artículo se propone el modelo BN-Cox (interpretación de red bayesiana del modelo de riesgos proporcionales de Cox), donde resultó ser el de mayor precisión de un grupo de modelos. En particular, BN-Cox es una red bayesiana construida con conocimiento previo de las variables del modelo, es decir, fue construida de manera manual. El modelo BN-Cox resultó ser más preciso incluso que las redes bayesianas en donde estructura fue aprendida por los datos. En ese trabajo, el aprendizaje de la estructura es hecha en software *GeNIe* disponible en Druzdzel (1999).

GeNIe es una herramienta para el modelado de inteligencia artificial y el aprendizaje automático con redes bayesianas y otros tipos de modelos gráficos probabilísticos, que ha sido probado exhaustivamente en el campo desde 1998, ha recibido una amplia aceptacion tanto en la academia como en la industria. Es una interfaz gráfica intuitiva para *SMILE* (*Structural Modelling, Inference, and Learning Engine*), la cual permite la creación y el aprendizaje de modelos interactivos. Aunque *GeNIe* y *SMILE* son desarrollados por el Laboratorio de Sistemas de Decisión de Druzdzel. *GeNIe* está desarrollado para el entorno de Windows, también se puede usar en mac OS y Linux en Wine según Scutari & Denis (2014).

GeNIe se enfoca en la inferencia en redes bayesianas e implementa varios algoritmos exactos y aproximados como el Muestreo de Importancia Adaptativa (AIS-BN) de Cheng & Druzdzel (2000), que admiten tanto redes bayesianas discretas como continuas. En lo que respecta al aprendizaje de estructuras, *GeNIe* implementa los clasificadores Naive Bayes y Tree augmented naive Bayes, el algoritmo de PC y dos heurísticas de búsqueda codiciosas utilizando BDe como puntaje de red. Cabe mencionar que en *GeNIe*, ninguno de los algoritmos de aprendizaje de estructura (excepto el algoritmo Naive Bayes, que crea una estructura de modelo basada en una fuerte suposición de independencia) es capaz de aprender la estructura de un modelo cuando faltan valores en los registros.

Dado que el modelo BN-Cox de Kraisangka & Druzdzel se hace estudio de la estructura de red bayesiana mediante *GeNIe*, se cree que se puede hacer un mejor aprendizaje de estructura de red bayesiana a través de otro método. Así que formulamos la hipótesis:

Una estructura de red bayesiana aprendida por los datos es más precisa que una red bayesiana que su estructura fue aprendida manualmente.

Para probar la hipótesis anterior, se propone indagar en la estructura de red bayesiana mediante el trabajo propuesto por Baioletti et al. (2018) descrito en la Subsección 3.1.2, el cual

es un algoritmo basado en puntuación y propone un novedoso método para indagar la estructura de red bayesiana. El método consiste de utilizar un algoritmo evolutivo diferencial para redes bayesianas. Éste algoritmo hace el supuesto de que en la base de datos se cuenta con datos completos, es decir, no hay datos faltantes.

De lo anterior el modelo que se propone es utilizar redes bayesianas como interpretación del modelo de riesgos proporcionales de Cox, pero con una estructura de red bayesiana aprendida mediante el enfoque de evolución diferencial.

3.1.1. Una interpretación de red bayesiana del modelo de Cox

Kraisangka & Druzdzel se centran en hacer una interpretación de red bayesiana del modelo de riesgos proporcionales de Cox (CPH). Ellos proponen un método para codificar el conocimiento de los modelos de CPH existentes en el proceso de ingeniería de conocimiento para redes bayesianas. Los autores comparan la precisión del modelo propuesto resultante con el modelo CPH original, la estimación de Kaplan-Meier y las redes bayesianas aprendidas de los datos, incluidos Naive Bayes, árbol aumentado Naive Bayes, Noisy-Max y el aprendizaje de parámetros mediante el algoritmo EM. El modelo BN-Cox resultó como el más preciso de todos los enfoques de redes bayesianas y muy cercano al modelo CPH original.

BN-Cox

Kraisangka & Druzdzel proponen el uso de una red bayesiana añadiendo un tipo especial de variable. En general estas redes se denominan diagrama de influencia (también conocida como red de decisión). En general un diagrama de influencia respalda la representación y solución de problemas de decisión secuenciales con múltiples funciones de utilidad local bajo el supuesto de un recuerdo perfecto de todas las observaciones y decisiones tomadas en el pasado. Una red bayesiana es un caso particular de diagrama de influencia. Una red de decisión extiende una red bayesiana en el sentido de que interpretar la red bayesiana como una red probabilística para la toma de decisiones bajo incertidumbre pero se puede extender

a una red de decisión con la adición de nodos de utilidad Koller & Friedman (2009). Esto es, una red de decisión es una red bayesiana que tiene tres tipos de nodos:

- Nodos de variables aleatorias (X_C): nodos que representan eventos no controlado por el tomador de decisiones.
- Nodos de decisión (X_D): nodos que representan acciones bajo control directo del tomador de decisiones.
- Nodos de utilidad (X_U): nodos que representan la preferencia del tomador de decisiones. Estos nodos no pueden ser padres de nodos aleatorios o nodos de decisión.

Una red bayesiana sólo tiene nodos que representan variables aleatorias, al agregar nodos de decisión y de utilidad se forma un diagrama de influencia. Así un tomador de decisiones interesado en elegir las mejores acciones posibles puede especificar una estructura y distribuciones de probabilidad marginal o usar el modelo aprendido con el aprendizaje de estructura e inferencia. El tomador de decisiones luego atribuye funciones de utilidad a estados particulares de un nodo. El objetivo del análisis de decisiones es identificar las opciones de decisión que maximizan la utilidad esperada. Es importante notar que, aunque Kraisangka & Druzdzel añaden una variable de decisión su meta no es realizar análisis de decisiones, sino indicar la naturaleza no estocástica de esa variable.

Ahora se muestra cómo usar los parámetros de los modelos CPH existentes para crear redes bayesianas, propuesta hecha por Kraisangka & Druzdzel. Este enfoque es especialmente útil cuando hay muy poca o ninguna información disponible sobre la red bayesiana. Se supone que no se violan los supuestos del modelo de CPH y que los factores de riesgo o las variables aleatorias $\boldsymbol{X} = \{X_1, X_2, \ldots, X_n\}$ son variables discretas independientes del tiempo. Para crear una red bayesiana, primero crean su estructura designando las variables aleatorias que representan los factores de riesgo como padres $\boldsymbol{X}$ del nodo de supervivencia S. El número de estados de cada variable aleatoria es el mismo que en el modelo CPH, ésto ya que se trabaja con variables aleatorias discretas. Luego, se representa el tiempo explícitamente agregando

una variable indexada para el tiempo (T, variable determinista), capturando cada punto discreto en el tiempo que sea de interés, por ejemplo, cada día , cada dos semanas, etc. La Figura 3.1 muestra un ejemplo de dicho modelo (BN-Cox), que muestra la relación entre los factores de riesgo $\boldsymbol{X}$, la variable de tiempo T y el nodo de supervivencia S. Cabe mencionar que el variable T correspondiente al tiempo, es considerada como una variable de decisión, es decir, el modelo BN-Cox es un diagrama de influencia.

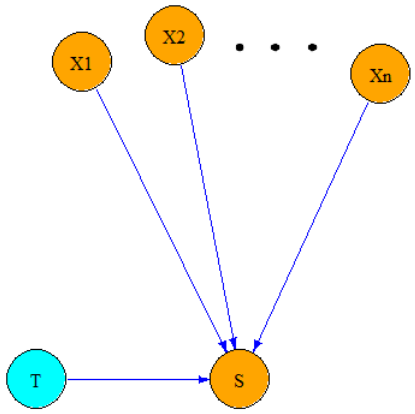


Figura 3.1: Modelo BN-Cox

En el siguiente paso, se crea la tabla de probabilidad condicional para el vértice de supervivencia S. Las probabilidades condicionales corresponden a las probabilidades de supervivencia en el modelo CPH, esto es, para cada instante de la variable T, se obtiene un conjunto de probabilidades de supervivencia, $S(t)$ del modelo CPH.

La probabilidad de supervivencia calculada para cada combinación de factores de riesgo corresponde a la probabilidad condicional de supervivencia. Por lo tanto, la probabilidad condicional que se codificará en las tabla de probabilidad condicional se estima por

$$P(s|\boldsymbol{X}, T = t) \quad = \quad S_0(t)^{\exp(\boldsymbol{\beta' X})}, \tag{3.1}$$

donde s corresponde al estado en el que se encuentra el nodo de supervivencia S, $\boldsymbol{X}$ son los factores de riesgo, T es el punto de tiempo y $\boldsymbol{\beta}$ los coeficientes del modelo CPH. Lo descrito

anteriormente permitió reproducir el modelo CPH mediante una red bayesiana, dando lugar al modelo propuesto (BN-Cox) por Kraisangka & Druzdzel.

Es importante mencionar que el artículo de Kraisangka & Druzdzel no menciona nada referente a la significancia de los parámetros en el modelo de regresión de Cox. Es decir, no se hace una selección de variables para la red bayesiana a partir de las variables significativas. Además de que no se verifican que la base de datos trabajados cumplan los supuestos del modelo de riesgos proporcionales de Cox.

3.1.2. Evolución diferencial para el aprendizaje de la estructura de redes bayesianas

En la literatura ya se tiene variantes de algoritmos genéticos propuestos como algoritmos para estudiar la estructura de una red bayesiana, por ejemplo Larrañaga & Poza (1994), Larrañaga et al. (1996) y Shetty et al. (2008). Los algoritmos genéticos, son algoritmos de búsqueda basados en la mecánica de la selección natural y la genética natural. Combinan la supervivencia del más apto entre estructuras con un intercambio de información estructurado pero aleatorizado para formar un algoritmo de búsqueda que evoluciona al óptimo con probabilidad 1.

Un algoritmo genético típico trabaja con poblaciones de individuos, cada uno de los cuales debe codificarse utilizando una función representativa y evaluarse utilizando una función de aptitud para medir la adaptabilidad de cada individuo. Estas dos funciones son los bloques de construcción básicos de un algoritmo genético. Para realizar realmente el algoritmo, se utilizan tres operadores genéticos para explorar el conjunto de soluciones: *reproducción, mutación y cruza*. El operador de reproducción promueve las mejores estructuras individuales para la próxima generación. Es decir, el individuo con la aptitud más alta en una población se reproducirá con una probabilidad más alta que el que tiene la mejor aptitud. El operador de mutación alterna una posición en la representación simbólica de las soluciones potenciales.

La mutación evita los óptimos locales al explorar nuevas soluciones al introducir una variación en la población. El operador de cruza intercambia material genético para generar nuevos individuos al seleccionar un punto donde se intercambian piezas de padres. Los parámetros principales, que influyen en el proceso de búsqueda del algoritmo genético, son la población inicial, el tamaño de la población, la mutación y los operadores de cruza.

Un algoritmo evolutivo diferencial es un tipo de algoritmo genético con un forma muy especifica de mutación, cruza y selección. En Price et al. (2006) se describe a fondo estos tipos de algoritmos diferenciales. Evolución diferencial es un algoritmo evolutivo simple y potente para optimizar funciones reales no lineales e incluso no diferenciables $f : \mathbb{R}^n \to \mathbb{R}$. Por lo tanto, ED evoluciona una población de N vectores de valores reales $x_1, \ldots, x_N \in \mathbb{R}^n$ mediante la aplicación iterativa de los tres operadores genéticos: mutación diferencial, cruza y selección. La mutación diferencial genera un mutante y_i para cada individuo objetivo x_i de la población. Aunque se han propuesto varios esquemas de mutación, el original se denota por `rand/1` y se calcula como:

$$y_i \quad = \quad x_{r_1} + F \cdot (x_{r_2} - x_{r_3}), \tag{3.2}$$

donde r_1, r_2, r_3 son tres números aleatorios en $\{1, \ldots, N\}$ mutuamente diferentes entre ellos y diferente a i, mientras que $F > 0$ es el parámetro del factor de escala. Para cada par formado por el individuo objetivo x_i y el mutante y_i, la cruza genera un nuevo individuo de prueba z_i mediante la recombinación de x_i y y_i. La variante más común es la cruza binomial que genera z_i de acuerdo con

$$z_i^{(j)} \quad = \quad \begin{cases} y_i^{(j)}, & \text{si} \quad u_j \leq CR \text{ o } v = j \\ \\ x_i^{(j)}, & \text{en otro caso} \end{cases} \tag{3.3}$$

donde el superíndice (j) denota la posición j, CR es la probabilidad de cruza, otro parámetro de ED, $u_j \in [0, 1]$ es un número aleatorio generado para j, y $v \in \{1, \ldots, n\}$ se genera aleatoriamente para garantizar que al menos un componente se hereda del mutante y_i. Existen otros

esquemas de cruza disponibles en la literatura. Finalmente, el operador de selección más utilizado compara cada x_i individual objetivo con los ensayos correspondientes z_i y selecciona el mejor entre ellos para ingresar en la población de la siguiente generación.

Para proponer una versión de algoritmo evolutivo Baioletti et al. (2018) hace uso de la teoría de grupos. En álgebra abstracta, un grupo es una estructura algebraica formada por un conjunto no vacío dotado de una operación interna que combina cualquier par de elementos para componer otro elemento, dentro del mismo conjunto y que satisface las propiedades asociativa, existencia de elemento neutro y simétrico.

Definición 3.1.1. *Un grupo G es un par $(X, \star)$ donde $X \neq \emptyset$ es un conjunto y $\star$ una operación interna entre elementos de X. Tal que $(X, \star)$ debe cumplir con*

- *La operación $\star$ es asociativa:* $\forall x, y, z \in X \quad x \star (y \star z) = (x \star y) \star z$.
- *Existencia del elemento neutro:* $\exists! e \in X : \ e \star a = a, \forall a \in G$.
- *Existencia del elemento inverso:* $\forall a \in X, \ \exists a^{-1} : \ a \star a^{-1} = e$.

La metodología propuesta por Baioletti et al. se puede aplicar a todos los problemas combinatorios cuyo espacio de búsqueda X forma un grupo finamente generado con respecto a una composición interna $\star$ y un conjunto de generadores $H \subseteq X$.

Definición 3.1.2. *Un grupo $(X, \star)$ es finitamente generado si existe un subconjunto finito $H \subseteq X$, de manera que cualquier $x \in X$ se puede descomponer como $x = h_1 \star h_2 \star \cdots \star h_l$ con $h_1, h_2, \ldots, h_l \in H$.*

H es llamado conjunto generador. Se denota por $|x|$ la longitud de una descomposición mínima de x en términos de H.

Un grafo de Cayley es un grafo que muestra la estructura de un grupo. El grafo de Cayley de un grupo finitamente generado es el dígrafo etiquetado cuyos vértices son las soluciones en X; existe un arco de x a y *etiquetado* por $h \in H$ si y sólo si $y = x \star h$. Además, para todos los

$x \in X$, cada trayecto desde el elemento neutro e hasta x corresponde a una descomposición mínima de x, es decir, elementos en el grupo generador capaces de expresar un elemento del grupo, por ejemplo, supongamos que para expresar x, una forma es a través de los arcos $(h_1, h_2, \ldots, h_l)$ en el conjunto generador, entonces a es de la forma $x = h_1 \star h_2 \star \cdots \star h_l$.

El grafo de Cayley tiene una importante interpretación geométrica. De hecho, cualquier solución $x \in X$ puede verse tanto como un punto, es decir, como un vértice en el grafo, y también como un vector porque su descomposición es una secuencia de elementos que lo generan, es decir, arcos de una trayectoria en el grafo de Cayley. Esta interpretación dicotómica permite definir las operaciones $\oplus, \ominus, \odot$ en X de tal manera que simulan las operaciones análogas del espacio euclidiano. Estas operaciones permitirán definir operar redes bayesianas para definir el algoritmo evolutivo adaptado a redes bayesianas.

La suma denotada como $x \oplus y$, se define como la aplicación del vector $y \in X$, descompuesta como $(h_1, h_2, \ldots, h_l)$, al punto $x \in X$. Se puede mostrar que

$$x \oplus y \quad = \quad x \star y. \tag{3.4}$$

Dado $x, y \in X$ considerados como puntos, su diferencia $y \ominus x$ es el vector $(h_1, h_2, \ldots, h_l)$ que son las etiquetas de una ruta que va desde x hasta y. Se prueba que

$$y \ominus x \quad = \quad x^{-1} \star y, \tag{3.5}$$

donde x^{-1} es un el elemento en X tal que $x \star x^{-1} = e$. Dado un $a \in [0, 1]$ y $x \in X$, el resultado de la multiplicación escalar de x por el escalar a, denotado por $a \odot x$, se define como

$$a \odot x \quad = \quad h_1 \star h_2 \star \cdots \star h_k, \tag{3.6}$$

donde $(h_1, h_2, \ldots, h_l)$ es una descomposición mínima de x y $k = \lceil a \cdot |x| \rceil$. La operación $\odot$, contrariamente a $\oplus$ y $\ominus$, depende de la descomposición mínima particular elegida para x. En general puede haber múltiples descomposiciones mínimas, por lo tanto, $\odot$ no se define de

forma única. Sin embargo, dado que se esta diseñando un algoritmo evolutivo, se considera una descomposición aleatoria mínima de x cuando se calcula $a \odot x$. Para el caso de redes bayesianas, esta descomposición está dada en términos del conjunto A del Teorema 3.1.3.

Representación dual de las redes bayesianas

En este apartado, se muestra la representación de las estructuras de una red bayesiana, es decir, los GAD y su grupo asociado. Un GAD, G, de n vértices puede ser representado por un par $(\pi, \mathbf{b})$, donde $\pi \in S_n$, $\mathbf{b} \in \mathbb{B}^m$ con $m = \binom{n}{2}$, es decir, π es una permutación en S_n (es el conjunto de todas las permutaciones con n elementos) y $\mathbb{B} \in \{0, 1\}$. Entonces π es una permutación de los vértices y $\mathbf{b}$ es un vector de longitud m de 1's y 0's.

El vector de bits $\mathbf{b}$ representan el esqueleto de G, esto es, al definir $C = \{(j, k) : 1 \leq j < k \leq n\}$ el conjunto de pares ordenados, si el i-ésimo par de C es (j, k), entonces existe en G una arista de la variable X_j a X_k, si y sólo si $b_i = 1$. Por otra parte, la permutación π determina la dirección de los arcos, si $b_i = 1$, entonces la arista va de X_j a X_k si j aparece antes de k en π, es decir, $\pi^{-1}(j) < \pi^{-1}(k)$, de lo contrario la arista va en la dirección opuesta. Dicho en otras palabras, π es un orden topológico de las variables $X_1, \ldots, X_n$.

Proposición 3.1.1. *Cualquier par ordenado* $(\pi, \mathbf{b})$ *representa un GAD.*

Demostración:

Supongamos que $(\pi, \mathbf{b})$ *no es un GAD, de lo anterior es claro que* $(\pi, \mathbf{b})$ *originan un grafo dirigido, por lo que la única forma de que* $(\pi, \mathbf{b})$ *no se un GAD es que al menos existe un ciclo. Sea* $x_{i_1} \to x_{i_2} \to \cdots \to x_{i_n} \to x_{i_1}$ *un ciclo en el grafo dirigido* $(\pi, \mathbf{b})$*, luego* $\pi^{-1}(i_1) < \pi^{-1}(i_2) < \ldots < \pi^{-1}(i_n) < \pi^{-1}(i_1)$*, de lo cual se concluye que* $\pi^{-1}(i_1) < \pi^{-1}(i_1)$ *lo cual es una contradicción ya que contradice la relación de orden de los números naturales. Por tanto* $(\pi, \mathbf{b})$ *es un GAD.*

■

Una de las propiedades más importantes de esta representación es la Proposición 3.1.1. Este hecho es una ventaja aparente de esta representación con respecto a otras formas donde se de-

be usar la restricción para seleccionar qué combinaciones corresponden a un GAD. Por otro lado hay que notar que un GAD puede tener más de un orden topológico, esta representación es, en general, una representación de muchos a uno, es decir, puede haber varios pares $(\pi, \mathbf{b})$ que representan el mismo GAD.

El conjunto de todos los pares $(\pi, \mathbf{b})$, tal que $\pi \in S_n$ y $\mathbf{b} \in \mathbb{B}^m$ es el producto cartesiano $\mathcal{B} = S_n \times \mathbb{B}^m$. Es importante destacar que $\mathcal{B}$ puede ser dotado de la operación binaria $*$ definida como

$$(\pi_1, \mathbf{b}_1) * (\pi_2, \mathbf{b}_2) \quad = \quad (\pi_1 \circ \pi_2, \mathbf{b}_1 \veebar \mathbf{b}_2), \tag{3.7}$$

donde $\circ$ es la composición de permutaciones y $\veebar$ es el operador lógico XOR (O-exclusivo). La proposición 3.1.2 muestra que $\mathcal{B}$ es un grupo con respecto a $*$, es decir, el grupo de productos de S_n y $\mathbb{B}^m$.

Proposición 3.1.2. *El par $(\mathcal{B}, *)$ es un grupo, donde $\mathcal{B} = S_n \times \mathbb{B}^m$.*

Demostración:

De la definición de grupo, es necesario mostrar la asociatividad con respecto a $*$, la existencia del elemento neutro y del elemento inverso.

La operación $*$ es asociativa ya que para $(\pi_1, \mathbf{b}_1), (\pi_2, \mathbf{b}_2), (\pi_3, \mathbf{b}_3) \in \mathcal{B}$ se tiene que

$$\begin{aligned}
(\pi_1, \mathbf{b}_1) * ((\pi_2, \mathbf{b}_2) * (\pi_3, \mathbf{b}_3)) &= (\pi_1, \mathbf{b}_1) * ((\pi_2 \circ \pi_3, \mathbf{b}_2 \veebar \mathbf{b}_3)) \\
&= (\pi_1 \circ (\pi_2 \circ \pi_3), \mathbf{b}_1 \veebar (\mathbf{b}_2 \veebar \mathbf{b}_3)) \\
&= ((\pi_1 \circ \pi_2) \circ \pi_3, (\mathbf{b}_1 \veebar \mathbf{b}_2) \veebar \mathbf{b}_3) \\
&= ((\pi_1, \mathbf{b}_1) * (\pi_2, \mathbf{b}_2)) * (\pi_3, \mathbf{b}_3).
\end{aligned}$$

El elemento neutro es $(\iota, \mathbf{0})$, donde ι es la permutación identidad y el vector $\mathbf{0}$ corresponde a la identidad del operador XOR, ya que $(\iota, \mathbf{0}) * (\pi, \mathbf{b}) = (\pi, \mathbf{b})$ para $(\pi, \mathbf{b}) \in \mathcal{B}$.

El elemento inverso de $(\pi, \mathbf{b})$ es $(\pi^{-1}, \mathbf{b})$, ya que $(\pi, \mathbf{b}) * (\pi^{-1}, \mathbf{b}) = (\iota, \mathbf{0})$. Por tanto se concluye que $(\mathcal{B}, *)$ es un grupo.

■

La suma y la resta en $\mathcal{B}$ ahora se pueden definir como en las ecuaciones (3.4) y (3.5), utilizando la operación $*$ y su operador inverso relacionado. Para definir la multiplicación de un par $(\pi, \mathbf{b})$ por un escalar $a \in [0, 1]$, se debe elegir un grupo generador para $\mathcal{B}$.

Un conjunto generador de $\mathcal{B}$ es $A = ST \cup U$ (ver Proposición 3.1.3) donde ST genera el conjunto de todas las permutaciones S_n y U genera los vectores de bits de longitud m. Usando al conjunto A como grupo generador, es fácil probar que $|(\pi, \mathbf{b})| = |\pi| + |\mathbf{b}|$ y la cardinalidad de A es $|A| = n - 1 + m$.

Proposición 3.1.3. *El grupo $\mathcal{B} = S_n \times \mathbb{B}^m$ es finitamente generado por el conjunto A con respecto a la operación $*$ definida en la Ecuación (3.7)*

$$A = ST \cup U$$

donde $ST = \{(\sigma_i, \mathbf{0}) : i = 1, \ldots, n-1\}$ y $U = \{(\iota, \mathbf{u}_j) : j = 1, \ldots, m\}$ tal que σ_i corresponde a un intercambio adyacente entre las posiciones i y $i+1$, es decir; $\sigma_i(i) = i+1$, $\sigma_i(i+1) = i$ y $\sigma_i(j) = j$ para todo $j \in \{1, 2, \ldots, n\} \backslash \{i, i+1\}$ y $\mathbf{u}_j$ es un vector de ceros excepto en la posición j, donde hay un uno.

Demostración:

Sea $r \in \mathcal{B}$, luego r es tal que $r = (\pi, \mathbf{b})$ con π una permutación en S_n, $\mathbf{b}$ un vector de bits en $\mathbb{B}^m$ de dimensión $m = \binom{n}{2}$. Se define $ST' = \{\sigma_i : i = 1, \ldots, n-1\}$ tal que σ_i corresponde a un intercambio adyacente entre las posiciones i y $i+1$. Dado que n es finito y $\pi \in S_n$ existe un entero l tal que $\pi = \pi_{k_l} \circ \cdots \circ \pi_{k_2} \circ \pi_{k_1}$ con $\pi_{k_1}, \ldots, \pi_{k_l} \in ST'$, donde $\pi_{k_i} \neq \pi_{k_j}$ para $i \neq j$.

Análogamente, se define $U' = \{u_j : j = 1, \ldots, m\}$ vector de ceros excepto en la posición j, donde hay un uno. Ya que n es finito, m también es finito. Luego si $\mathbf{b} \in \mathbb{B}^m$ que es cero excepto en las las posiciones $w_1, w_2, \ldots, w_{l'}$, es claro que tal que el vector se puede obtener como $\mathbf{b} = \mathbf{b}_{w_1} \veebar \mathbf{b}_{w_2} \veebar \cdots \veebar \mathbf{b}_{w_{l'}}$ con $\mathbf{b}_{w_1}, \mathbf{b}_{w_2}, \ldots, \mathbf{b}_{w_{l'}} \in U'$, donde $\mathbf{b}_{w_i}$ es vector de ceros y un uno en la posición w_j.

De lo anterior, y recordando que ι corresponde a la permutación identidad en S_n y $\mathbf{0}$ es la correspondiente identidad en $\mathbb{B}^m$ con respecto al operador lógico $\veebar$ se tiene que

$$(\pi, \mathbf{b}) = (\pi_{k_1}, \mathbf{0}) * (\pi_{k_2}, \mathbf{0}) * \cdots * (\pi_{k_l}, \mathbf{0}) * (\iota, \mathbf{b}_{w_1}) * (\iota, \mathbf{b}_{w_2}) * \cdots * (\iota, \mathbf{b}_{w_{l'}}).$$

Dado que $r = (\pi, \mathbf{b})$ fue un elemento cualquiera en $\mathcal{B}$ se concluye que A en efecto genera a $\mathcal{B}$. ■

Un algoritmo de descomposición para A que produce una descomposición mínima aleatoria de $(\pi, \mathbf{b}) \in \mathcal{B}$ es el mostrado en el Algoritmo 1.

Algoritmo 1 Pseudocódigo de algoritmo de descomposición mínima para A

Entrada: descomposición mínima aleatoria $(\sigma_{h_1}, \ldots, \sigma_{h_L})$ de π y una descomposición mínima aleatoria $(u_{k_1}, \ldots, u_{k_M})$ de $\mathbf{b}$

1: crear una secuencia vacia r de tamaño $L + M$.
2: elegir L indices aleatorios $1 \leq j_1 < \ldots < j_L \leq L + M$ de r.
3: asignar $r_{j_v} \leftarrow (\sigma_{h_v}, \mathbf{0})$ para $v = 1, \ldots, L$.
4: completar las M posiciones restantes de r con (ι, u_{k_v}) para $v = 1, \ldots, M$.
5: **regresar** descomposición mínima de $(\pi, \mathbf{b})$

Una importante cualidad del grupo $(\mathcal{B}, *)$ es que cualquier elemento puede ser generado pero también que es cerrado bajo las operaciones del algoritmo ED. En relación a ésto se tiene la siguiente proposición.

Proposición 3.1.4. *El grupo $\mathcal{B}$ es cerrado con respecto a la operación $*$.*

Demostración:

Es claro que $\mathcal{B}$ es cerrado bajo $*$. En efecto, es fácil notar que para $r_1, r_2 \in \mathcal{B}$ se cumple que $r_1 * r_2 = (\pi_1, \mathbf{b}_1) * (\pi_2, \mathbf{b}_2) = (\pi_1 \circ \pi_2, \mathbf{b}_1 \veebar \mathbf{b}_2)$ y por tanto, dado que $\pi_1 \circ \pi_2$ sigue sien una permutación pues $\pi_1 \circ \pi_2 \in S_n$ y $\mathbf{b}_1 \veebar \mathbf{b}_2$ sigue siendo un vector de bits en $\mathbb{B}^m$, se concluye que $r_1 * r_2 \in \mathcal{B}$. ■

Ejemplo 3.1.1. *Veamos un ejemplo de la operación $\oplus$ al considerar las redes de la Figura 3.2.*

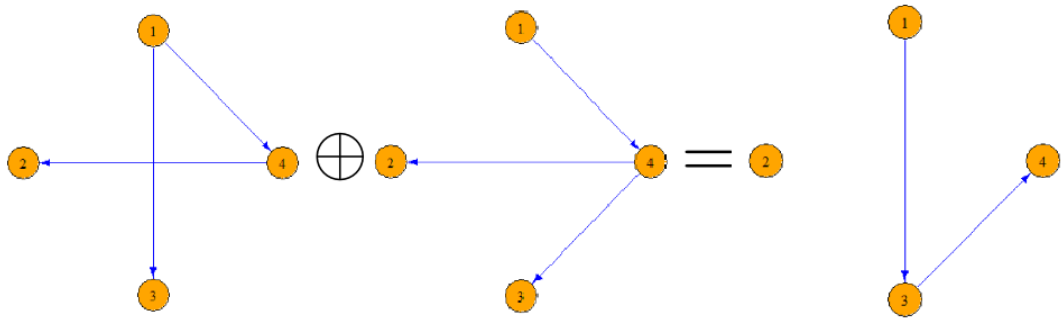


Figura 3.2: Operación $\oplus$ para dos GAD

La operación $\oplus$ hecha en la Figura 3.2, es equivalente a hacer en $\mathcal{B}$:

$$\left(\begin{pmatrix}1&2&3&4\\1&4&2&3\end{pmatrix},\quad (0,1,1,0,1,0)\right) \oplus \left(\begin{pmatrix}1&2&3&4\\1&4&3&2\end{pmatrix},\quad (0,0,1,0,1,1)\right)$$
$$= \left(\begin{pmatrix}1&2&3&4\\1&3&2&4\end{pmatrix},\quad (0,1,0,0,0,1)\right)$$

Ahora, un ejemplo de la operación $\ominus$ al considerar las redes de la Figura 3.3

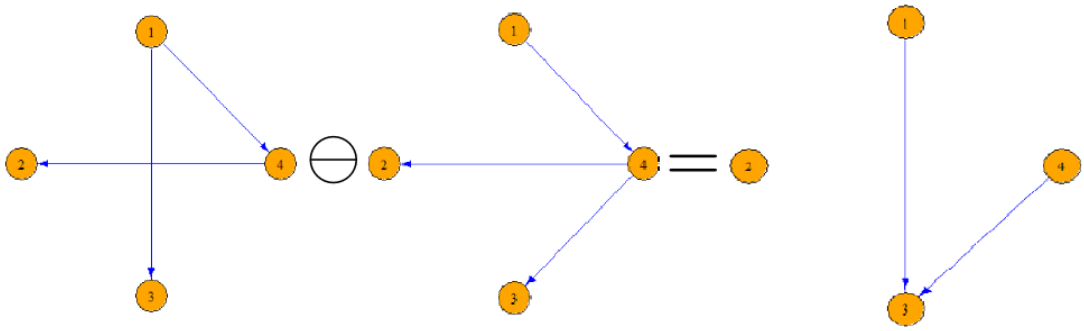


Figura 3.3: Operación $\ominus$ para dos GAD

La operación $\ominus$ hecha en la Figura 3.2, es equivalente a hacer en $\mathcal{B}$:

$$\left(\begin{pmatrix}1&2&3&4\\1&4&2&3\end{pmatrix},\quad (0,1,1,0,1,0)\right) \ominus \left(\begin{pmatrix}1&2&3&4\\1&4&3&2\end{pmatrix},\quad (0,0,1,0,1,1)\right)$$
$$= \left(\begin{pmatrix}1&2&3&4\\1&2&4&3\end{pmatrix},\quad (0,1,0,0,0,1)\right)$$

Finalmente un ejemplo de la operación $\odot$ al considerar la red de la Figura 3.4

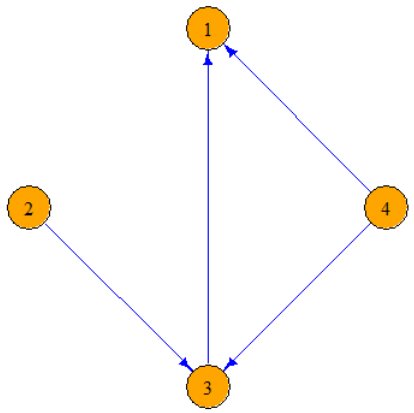


Figura 3.4: Grafo dirigido acíclico para la red $R1$

haciendo la descomposición mínima de $R1 = \left(\begin{pmatrix} 1 & 2 & 3 & 4 \\ 2 & 4 & 3 & 1 \end{pmatrix}, (0,1,1,1,0,1) \right)$ *se obtiene:*

$$
\begin{aligned}
R1 \;=\; & \left(\begin{pmatrix} 1 & 2 & 3 & 4 \\ 1 & 2 & 3 & 4 \end{pmatrix}, (0,1,0,0,0,0) \right) * \left(\begin{pmatrix} 1 & 2 & 3 & 4 \\ 1 & 2 & 3 & 4 \end{pmatrix}, (0,0,1,0,0,0) \right) \\
* & \left(\begin{pmatrix} 1 & 2 & 3 & 4 \\ 2 & 1 & 3 & 4 \end{pmatrix}, (0,0,0,0,0,0) \right) * \left(\begin{pmatrix} 1 & 2 & 3 & 4 \\ 1 & 3 & 2 & 4 \end{pmatrix}, (0,0,0,0,0,0) \right) \\
* & \left(\begin{pmatrix} 1 & 2 & 3 & 4 \\ 1 & 2 & 4 & 3 \end{pmatrix}, (0,0,0,0,0,0) \right) * \left(\begin{pmatrix} 1 & 2 & 3 & 4 \\ 1 & 2 & 3 & 4 \end{pmatrix}, (0,0,0,1,0,0) \right) \\
* & \left(\begin{pmatrix} 1 & 2 & 3 & 4 \\ 1 & 2 & 3 & 4 \end{pmatrix}, (0,0,0,0,0,1) \right) * \left(\begin{pmatrix} 1 & 2 & 3 & 4 \\ 1 & 3 & 2 & 4 \end{pmatrix}, (0,1,0,0,0,0) \right)
\end{aligned}
$$

Supongamos $a = 0.3$*, notar que la longitud de descomposición mínima de* $R1$ *es 8, es decir*

$|R1| = 8$, *por tanto* $3 = \lceil 0.3 \cdot 8 \rceil$, *es decir* $a \odot R1$ *es tal que:*

$$
\begin{aligned}
a \odot R1 &= \left(\begin{pmatrix} 1 & 2 & 3 & 4 \\ 1 & 2 & 3 & 4 \end{pmatrix}, (0,1,0,0,0,0) \right) * \left(\begin{pmatrix} 1 & 2 & 3 & 4 \\ 1 & 2 & 3 & 4 \end{pmatrix}, (0,0,1,0,0,0) \right) \\
&* \left(\begin{pmatrix} 1 & 2 & 3 & 4 \\ 2 & 1 & 3 & 4 \end{pmatrix}, (0,0,0,0,0,0) \right) \\
&= \left(\begin{pmatrix} 1 & 2 & 3 & 4 \\ 2 & 1 & 3 & 4 \end{pmatrix}, (0,1,1,0,0,0) \right)
\end{aligned}
$$

La Figura 3.5 muestra el grafo correspondiente al operar $a \odot R1$

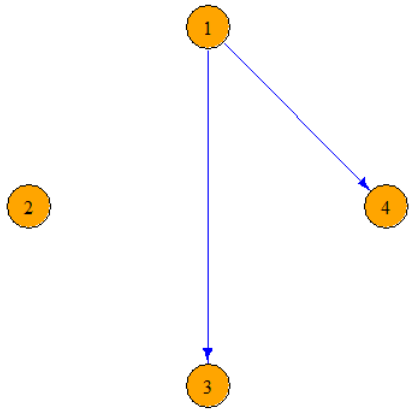


Figura 3.5: Grafo dirigido acíclico para la red $a \odot R1$ con $a = 0.3$

El algoritmo basado en la Evolución Diferencial para el aprendizaje de redes Bayesianas (DEBN, por su nombre en inglés *Differential Evolution Bayesian Network*), tiene la misma estructura de un algoritmo ED clásico: su pseudocódigo se representa en el Algoritmo 2. Cualquier individuo x_i de la población se representa por medio de la representación dual, es decir, $x_i = (\pi_i, \mathbf{b}_i)$, donde $\pi_i \in S_n$, $\mathbf{b}_i \in \mathbb{B}^m$, y $m = \binom{n}{2}$. Los individuos se evalúan mediante una función de puntuación de la red bayesiana seleccionada por el usuario. En este trabajo, se ha considerado K2, BDe y log verosimilitud. Cada individuo $x_i = (\pi_i, \mathbf{b}_i)$ se inicia aleatoriamente seleccionando una permutación π_i uniformemente al azar en S_n, mientras que cada bit $\mathbf{b}_i$ contiene un 1 con probabilidad $2/(n-1)$, esto ya que Baioletti et al. sugiere que el

Algoritmo 2 Pseudocódigo DEBN

1: Iniciar y evaluar la población $x_1, x_2, \ldots, x_N$
2: **mientras** no termine el criterio **hacer**
3: **para** $i \leftarrow 0$ hasta N **hacer**
4: simular F de $U(0, 1)$
5: simular pm de $U(0.1, 0.3)$
6: simular CR de $U(0, 1)$
7: $y_i \leftarrow$ MutaciónDiferencial(x_i, F, pm)
8: $z_i \leftarrow$ Cruza(x_i, y_i, CR)
9: Evaluar(z_i)
10: **fin para**
11: **para** $i \leftarrow 0$ hasta N **hacer**
12: $x_i \leftarrow$ Selección(x_i, z_i)
13: **fin para**
14: **fin mientras**
15: **regresar** la mejor estructura de red bayesiana.

número promedio de aristas en el GAD sea igual al número de vértices n.

La mutación diferencial para los GADs utiliza las operaciones algebraicas $\oplus, \ominus, \odot$, de $\mathcal{B}$. Además, para mitigar el fenómeno de pérdida de diversidad, típico en los espacios combinatorios, Baioletti et al. propone introducir un término aleatorio t como sigue:

$$y_i = (x_{r_1} \oplus t) \oplus F \odot (x_{r_2} \ominus x_{r_3}),$$

donde, x_{r_1}, x_{r_2} y x_{r_3} son tres individuos de la población aleatorias, y diferentes entre sí y respecto a x_i, mientras que $F \in [0, 1]$ es el parámetro del factor de escala. Además, $t \in \mathcal{B}$ se genera aleatoriamente por medio de la probabilidad de pre-mutación $pm \in (0, 1)$ de manera que $|t| = k$ con probabilidad pm^k. Operativamente, t se inicializa en el elemento neutro $(\iota, \mathbf{0})$, luego, durante un ciclo, se genera un número aleatorio $r \in [0, 1]$ y, si $r < pm$, un elemento en A es seleccionado al azar y se opera con t. Tan pronto como $r \geq pm$, el ciclo se detiene y se devuelve t, para dejar en claro como se selecciona t ver el Algoritmo 3.

Se aplican dos operadores de cruza por separado, uno a la permutación y otro para la parte

Algoritmo 3 Pseudocódigo DEBN

Entrada: $pm \in [0.1, 0.3]$

1: $t_0 = (\iota, \mathbf{0})$
2: **mientras** $r < pm_p$ **hacer**
3: $\quad t = t * t_0$
4: $\quad t_0 = t$
5: **fin mientras**
6: **regresar** t.

binaria. Así al cruzar el individuo $x_i = (\pi_i, \mathbf{b}_i)$ con el individuo $y_i = (\pi_i^{'}, \mathbf{b}_i^{'})$ se obtiene un nuevo individuo resultante de la cruza, $z_i = (\pi_i^{''}, \mathbf{b}_i^{''})$. Las cruzas utilizadas son las siguientes:

$$\pi^{''} = CYC(\pi_i, \pi_i^{'}),$$
$$\mathbf{b}_i^{''} = BIN(\mathbf{b}_i, \mathbf{b}_i^{'}, CR),$$

donde CYC es la cruza cíclica propuesta por Larranaga et al. (1999), y BIN es la cruza binomial habitual de ED, como se define en la Ecuación (3.3). La generación se concluye aplicando el esquema de selección 1 a 1 de la ED clásica, es decir, se compara x_i con z_i y se selecciona el mejor de ellos. Este algoritmo fue equipado por Baioletti et al. con un procedimiento autoadaptativo, inspirado en el conocido método jDE de Brest et al. (2006), que permite regular los tres parámetros pm, F y CR. Cada individuo de la población mantiene sus propios valores de parámetros, esto es, se tiene que los parámetros son seleccionados como variables aleatorias uniformes en el intervalo $[0.1, 1]$ para F, $[0, 1]$ para CR, y $[0.1, 0.3]$ para pm.

El operador de cruza cíclico fue propuesto por Oliver et al. (1987). Intenta crear una descendencia de dos individuos donde cada posición está ocupada por un elemento correspondiente de uno de los individuos, es decir, el descendiente es resultado de comparar componente a componente dos individuos. El Algoritmo 4 muestra el pseudocódigo que describe el cruce cíclico CYC. Para facilitar la compresión del Algoritmo 4, en el Ejemplo 3.1.2 se observa como se hace la cruza cíclica entre dos permutaciones.

Ejemplo 3.1.2. *Cruce cíclico CYC hace la cruza de individuos de la siguiente manera. Supongamos que π_1 y π_2 son dos individuos a cruzar*

$$\pi_1 = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \end{pmatrix} \qquad \pi_2 = \begin{pmatrix} 2 & 4 & 6 & 8 & 7 & 5 & 3 & 1 \end{pmatrix}$$

Se elige el primer elemento de la descendencia igual al primer elemento del primer individuo o el primer elemento del segundo individuo. Por lo tanto, el primer elemento de la descendencia debe ser un 1 o un 2. Suponga que se elige el 1

$$r = \begin{pmatrix} 1 & * & * & * & * & * & * & * \end{pmatrix}$$

Ahora, considere el último elemento de la descendencia. Dado que este elemento debe elegirse de uno de los individuos, sólo puede ser un 8 o un 1. Sin embargo, si se seleccionara un 1, la descendencia tendría dos 1's. Por lo tanto, se elige un 8

$$r = \begin{pmatrix} 1 & * & * & * & * & * & * & 8 \end{pmatrix}$$

Algoritmo 4 Pseudocódigo CYC

Entrada: $\pi_1, \pi_2 \in S_n$

1: $\boldsymbol{d} =$ vector descendiente inicializado con 0's (de longitud n)
2: $j = 1$
3: **mientras** $\boldsymbol{d}$ tiene algún elemento 0 **hacer**
4: flag=TRUE
5: $i = 0$
6: **si** j es impar **entonces**
7: $\boldsymbol{a} = \pi_1$ y $\boldsymbol{b} = \pi_2$
8: **si no**
9: $\boldsymbol{a} = \pi_2$ y $\boldsymbol{b} = \pi_1$
10: **fin si**
11: **mientras** flag=TRUE **hacer**
12: **si** $i = 0$ **entonces**
13: Sea w el primer elemento de $\boldsymbol{d}$ igual a 0
14: $d_w = a_w$
15: $i = 1$
16: **si no**
17: $k =$ el lugar donde el vector $\boldsymbol{b}$ es igual al último valor asignado a un elem. de d
18: **si** $d_k = 0$ **entonces**
19: $d_k = a_k$
20: **si no**
21: flag=FALSE
22: j=j+1
23: **fin si**
24: **fin si**
25: **fin mientras**
26: **fin mientras**
27: **regresar** descendiente d.

Análogamente, se encuentra que el cuarto y el segundo elemento de la descendencia también deben seleccionarse del primer individuo, lo que da como resultado

$$r = \begin{pmatrix} 1 & 2 & * & 4 & * & * & * & 8 \end{pmatrix}$$

Las posiciones de los elementos elegidos hasta ahora se dice que son un ciclo. Considere ahora el tercer elemento de la descendencia. Este elemento se puede elegir entre cualquiera de los dos individuos. Suponga que se selecciona que sea del segundo individuo. Esto implica que los elementos quinto, sexto y séptimo de la descendencia también deben elegirse del segundo individuo, ya que forman otro ciclo. Por lo tanto, la cruza cíclica de los individuos resulta ser la siguiente descendencia

$$r = \begin{pmatrix} 1 & 2 & 6 & 4 & 7 & 5 & 3 & 8 \end{pmatrix}$$

Por otro lado, el Ejemplo 3.1.3 muestra una la aplicación de algoritmo de evolución diferencial para la construcción de redes bayesianas. En este ejemplo se hace uso de la base de datos *asia* del paquete *bnlearn* de R. Esta es una base de datos sintéticos de Lauritzen & Spiegelhalter (1988) acerca de enfermedades pulmonares (tuberculosis, cáncer de pulmón or bronquitis) y visitas a Asia.

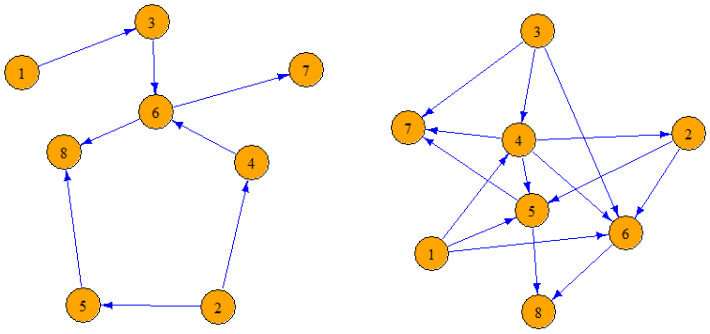
Ejemplo 3.1.3. *La base de datos asia cuenta con las siguientes variables:*

- *D (disnea). Dos factores de nivel, si y no.*
- *T (tuberculosis). Dos factores de nivel, si y no.*
- *L (cáncer de pulmón), dos factores de nivel with levels yes and no.*
- *B (bronquitis). Dos factores de nivel, si y no.*
- *A (visita a Asia). Dos factores de nivel, si y no.*
- *S (fumador). Dos factores de nivel, si y no.*
- *X (radiografía de pecho). Dos factores de nivel, si y no.*

- *E (tuberculosis vs cáncer de pulmón/bronquitis). Dos factores de nivel, si y no.*

Primero se hizo la codificación A=1, S=2, T=3, L=4, B=5, E=6, X=7 y D=8, ya que en el algoritmo evolutivo diferencial en el grupo $\mathcal{B}$ considera variables aleatorias los vértices de una permutación, es decir un GAD G es de la forma $G = (\pi, b)$ donde $\pi \in S_8$ y $b \in \mathbb{B}^{28}$ ya que hay $n = 8$ variables y $m = \binom{n}{2} = \binom{8}{2} = 28$. En este ejemplo se considera la puntuaciones $K2$ descrita en la Sección 2.2.1.

Al contar con una función de puntuación la cual se optimizará mediante el algoritmo de evolución diferencial, se simulan datos de un GAD conocido para evaluar el desempeño del algoritmo propuesto. Así, se generaron 1000 observaciones del GAD de la Figura 3.6 (a). Luego, al implementar el algoritmo evolutivo diferencial y generar 50 GAD aleatorios se obtiene la estructura del GAD mostrado en la Figura 3.6 (b).



(a) Grafo dirigido acíclico conocido con puntuación $K2 = -11110.15$.

(b) GAD resultante de DE con puntuación $K2 = -11151.5$.

Figura 3.6: Red bayesiana 1.

En la Figura 3.7 se muestra la puntuación K2 para la población de 50 GADs así como la puntuación para DAGs mutados del Ejemplo 3.1.3.

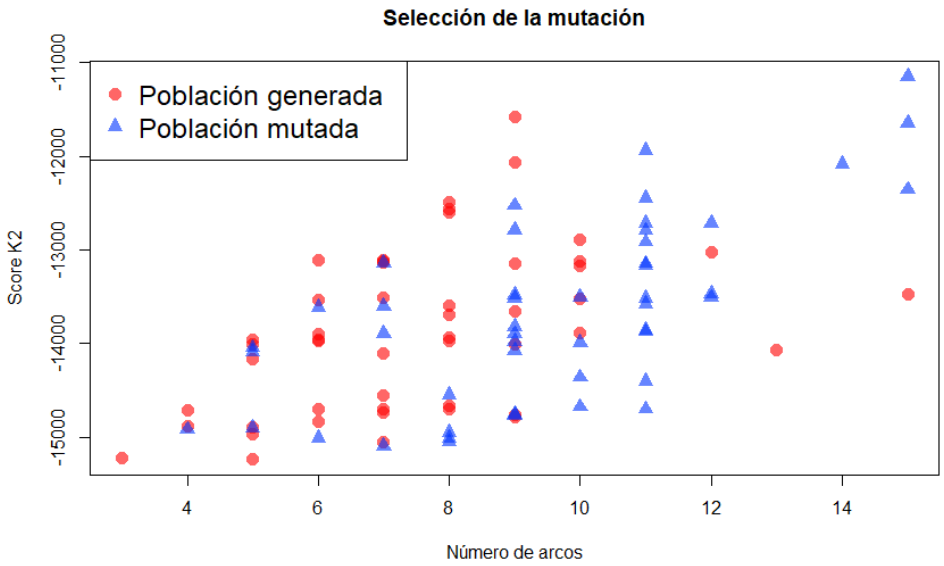


Figura 3.7: Puntuación K2 de los DAGs generados con el algoritmo DEBN.

3.2. Experimentos computacionales

En esta sección se generaron 5 redes bayesianas, para cada red bayesiana se fijó con un GAD y su respectiva probabilidad condicional dependiendo el nodo. Para cada red bayesiana se simularon 1000 datos de dicha red para el aprendizaje. Estas redes bayesianas no consideran un nodo que represente el tiempo, sino que los valores de cada variable se consideran ya dependiente del tiempo. Es decir, a diferencia del modelo BN-Cox consideramos que la variable tiempo absoluto no debe tener incidencia sobre los individuos, ya que estos pueden tener edades diferentes y por lo tanto su exposición al riesgo debería estar más bien relacionada a los valores propios y no de una variable absoluta.

Con el fin de modelar redes bayesianas para modelos de supervivencia, hicimos una modificación al algoritmo DEBN, denotado como DEBN Rest, donde al vértice que se considera como la variable de supervivencia se restringió a ser una hoja. Esto es, que cualquiera de las variables pueden apuntar a ésta pero esta variable no puede ser padre de ninguna de las cova-

riables. Esto se realiza fácilmente utilizando la representación del DAG en el ED, ya que la permutación π determina la dirección en un GAD $r = (\pi, \mathbf{b})$ cualquiera en $\mathcal{B}$, así que se fijó el último lugar de π a la etiqueta del último vértice y solo se permitió que la imagen inversa de la permutación $\pi \in S_n$ intercambie los elementos $n-1$ de la permutación π.

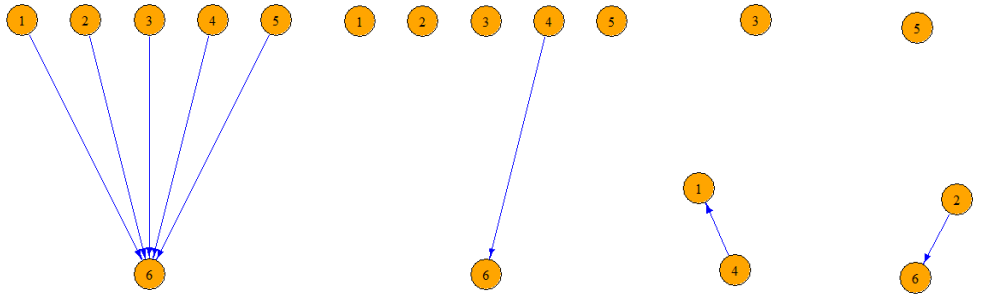
De lo anterior, las variantes de los modelos que se utilizó fueron las combinaciones posibles entre los algoritmo DEBN, DEBN Rest, Hill-Climbing (hc), Tabu Search, Max-Min Hill-Climbing (mmhc), Restricted Maximization (rm2 tabu) y las funciones de puntuación K2 (k2), BDe (bde) y log-likelihood (logVer).

Además para cada GAD de la red bayesiana simulada se construyó una tabla que especifica la puntuación obtenida según el tipo de GAD y la función de puntuación en turno. Esta tabla contiene también las métricas de sensibilidad, especificidad y el tiempo en segundos que tardo el algoritmo corriendo. La sensibilidad es la proporción de arcos que fueron asignados en el GAD aprendido y coinciden con el GAD real. Por otra parte, la especificidad es la proporción de no-arcos que fueron asignados en el GAD aprendido y coinciden con el GAD real.

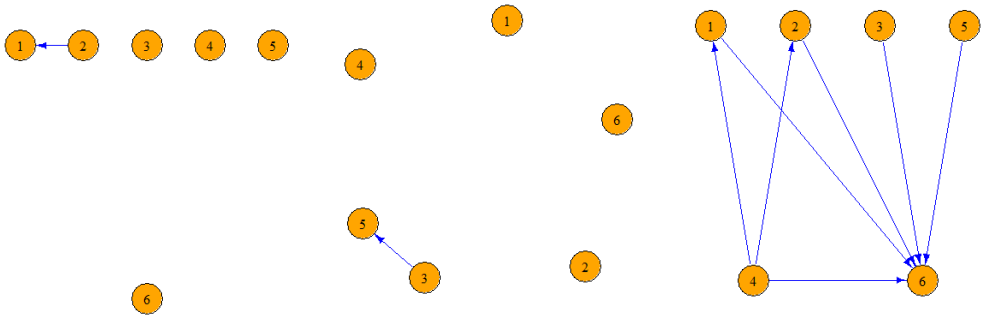
También se realizaron gráficas que permitieran comparar tanto la puntuación obtenida para cada estructura de red bayesiana aprendida por las diferentes variantes como para comparar el tiempo de ejecución de cada variante de estructura aprendida.

Es de suma importancia recordar que la programación fue implementada en R (R Core Team) y el código puede consultarse en https://github.com/gustavoberzada/Codigo_Tesis_Gustavo.

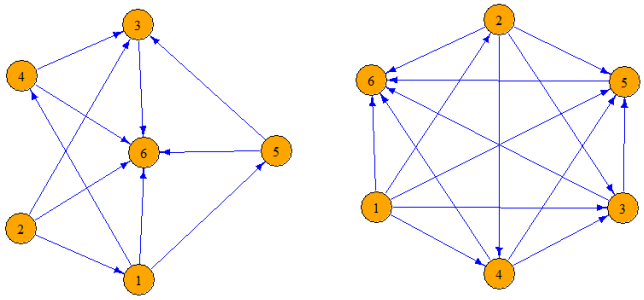
La Figura 3.8 (a) muestra el GAD real (red bayesiana co 6 vértices) del cual se simularon los datos, así como las estructuras más representativas obtenidos al hacer el aprendizaje correspondiente al GAD 1.



(a) Real. (b) DEBN k2. (c) DEBN k2 Rest.



(d) DEBN Bde. (e) DEBN Bde Rest. (f) DEBN logVer.



(g) DEBN logVer Rest. (h) HC logVer.

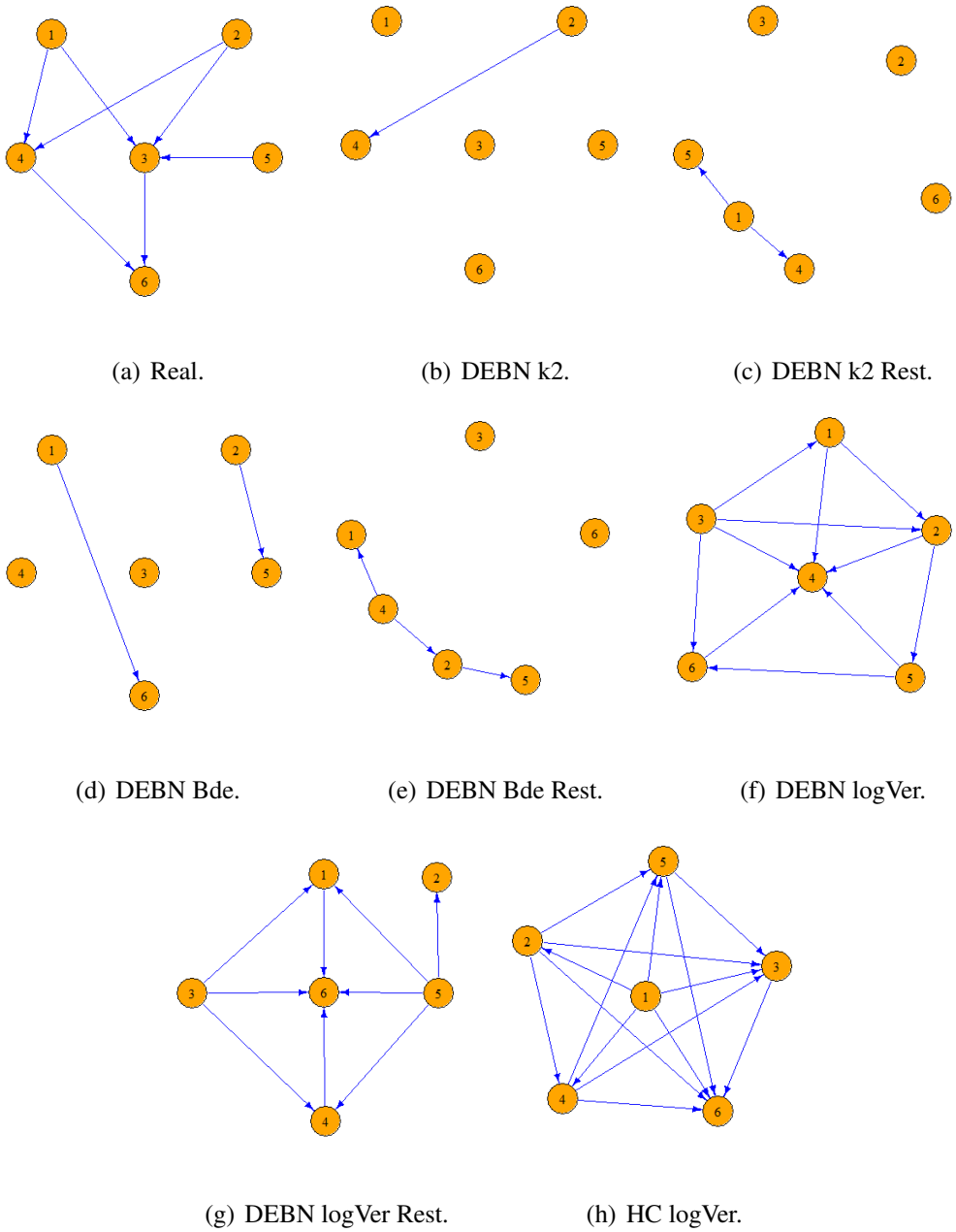
Figura 3.8: GAD 1.

Tipos de puntuación	Puntuación	Sensibilidad	Especificidad	Tiempo (s)
k2 Real	-5019.169	-	-	-
k2 DEBN	-4971.932	1/5	25/25	1.032
k2 DEBN Rest	-4975.622	1/5	24/25	1.76
k2 hc	-4969.787	0/5	25/25	0.005
k2 tabu	-4969.787	0/5	25/25	0.034
k2 mmhc	-4969.787	0/5	25/25	0.016
k2 rm2 tabu	-4969.787	0/5	25/25	0.001
bde Real	-5678.932	-	-	-
bde DEBN	-4986.615	0/5	24/25	1.218
bde DEBN Rest	-4992.561	0/5	24/25	1.84
bde hc	-4976.871	0/5	25/25	0.009
bde tabu	-4976.871	0/5	25/25	0.028
bde mmhc	-4976.871	0/5	25/25	0.012
bde rm2 tabu	-4976.871	0/5	25/25	0.007
logVer Real	-4803.378	-	-	-
logVer DEBN	-4848.879	5/5	23/25	1.04
logVer DEBN Rest	-4793.619	5/5	19/25	1.04
logVer hc	-4779.798	5/5	15/25	0.035
logVer tabu	-4779.798	5/5	15/25	0.024
logVer mmhc	-4935.693	0/5	25/25	0.003
logVer rm2 tabu	-4935.693	0/5	25/25	0.002

Tabla 3.1: GAD 1

La Tabla 3.1 muestra el desempeño de los GADs obtenidos para la estructura de red bayesiana 1, donde el GAD con mayor sensibilidad y especificidad es el aprendido mediante el algoritmo DEBN y la función de puntuación log-likelihood.

La Figura 3.9 (a) muestra el GAD real (red bayesiana co 6 vértices) del cual se simularon los datos, así como las estructuras más representativas obtenidos al hacer el aprendizaje correspondiente al GAD 2.



(a) Real. (b) DEBN k2. (c) DEBN k2 Rest.

(d) DEBN Bde. (e) DEBN Bde Rest. (f) DEBN logVer.

(g) DEBN logVer Rest. (h) HC logVer.

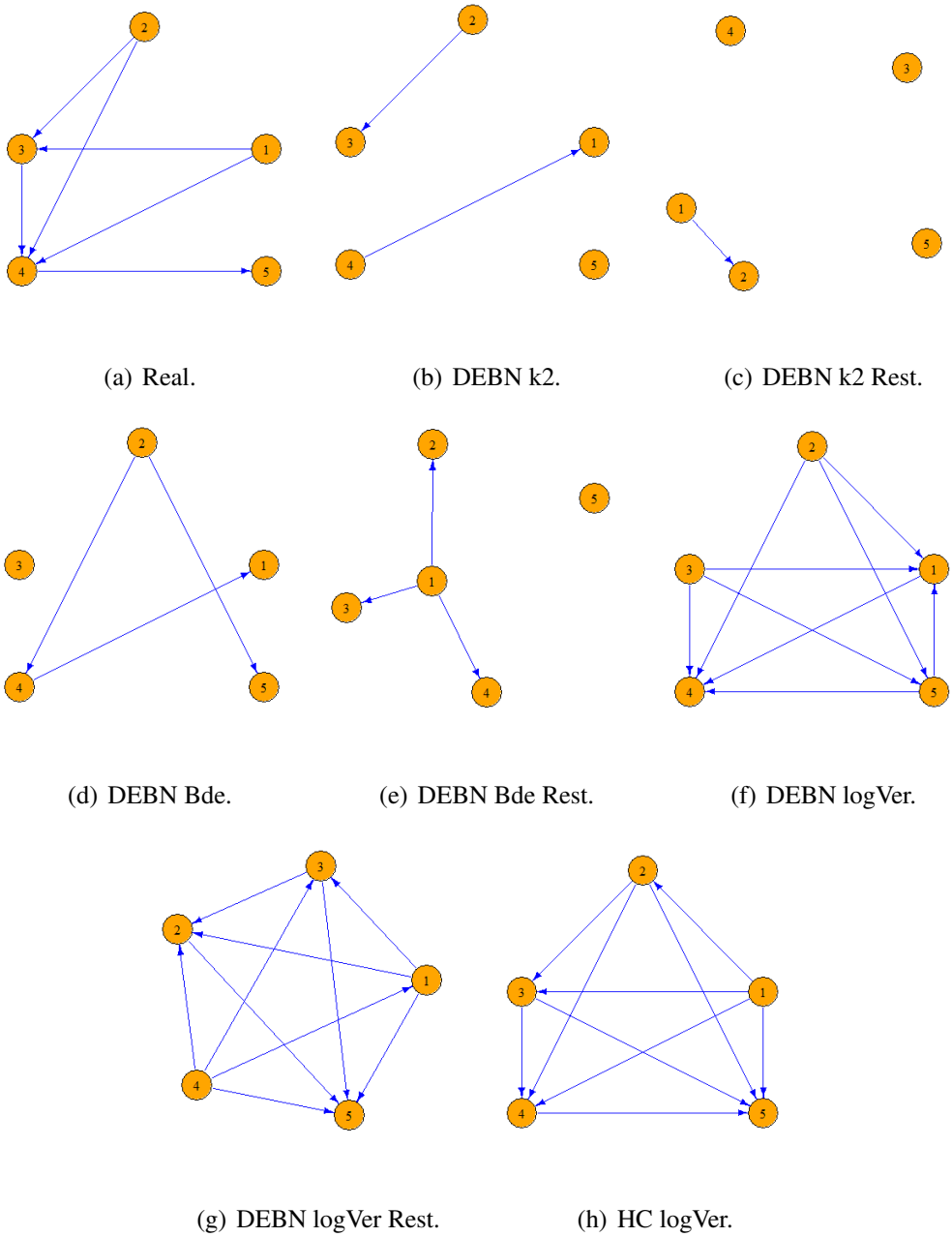
Figura 3.9: GAD 2.

Tipos de puntuación	Puntuación	Sensibilidad	Especificidad	Tiempo (s)
k2 Real	-5170.832	-	-	-
k2 DEBN	-5126.051	1/7	23/23	1.095
k2 DEBN Rest	-5130.552	1/7	22/23	1.66
k2 hc	-5123.243	0/7	23/23	0.006
k2 tabu	-5123.243	0/7	23/23	0.01
k2 mmhc	-5123.243	0/7	23/23	0.006
k2 rm2 tabu	-5123.243	0/7	23/23	0.005
bde Real	-5293.971	-	-	-
bde DEBN	-5142.537	1/7	21/23	1.234
bde DEBN Rest	-4992.561	0/5	20/23	2.07
bde hc	-5130.277	0/7	23/23	0.007
bde tabu	-5130.277	0/7	23/23	0.016
bde mmhc	-5130.277	0/7	23/23	0.008
bde rm2 tabu	-5130.277	0/7	23/23	0.006
logVer Real	-5066.337	-	-	-
logVer DEBN	-4931.222	3/7	15/23	1.092
logVer DEBN Rest	-4793.619	2/7	16/23	1.04
logVer hc	-4921.337	7/7	15/23	0.021
logVer tabu	-4921.337	7/7	15/23	0.02
logVer mmhc	-5081.368	0/7	22/23	0.003
logVer rm2 tabu	-5081.368	0/7	22/23	0.002

Tabla 3.2: GAD 2

La Tabla 3.2 muestra el desempeño de los GADs obtenidos para la estructura de red bayesiana 2, donde los GADs con mayor sensibilidad y especificidad son aprendidos mediante los algoritmos Hill-Climbing y Tabu Search y la función de puntuación log-likelihood para ambos casos.

La Figura 3.10 (a) muestra el GAD real (red bayesiana con 5 vértices) del cual se simularon los datos, así como las estructuras más representativas obtenidos al hacer el aprendizaje correspondiente al GAD 3.



(a) Real. (b) DEBN k2. (c) DEBN k2 Rest.

(d) DEBN Bde. (e) DEBN Bde Rest. (f) DEBN logVer.

(g) DEBN logVer Rest. (h) HC logVer.

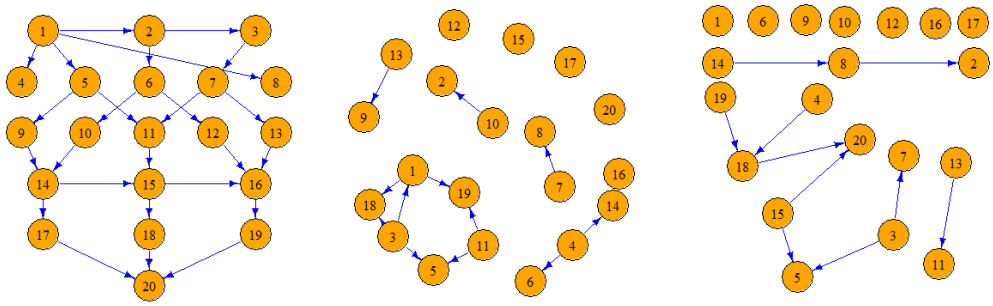
Figura 3.10: GAD 3.

Tipos de puntuación	Puntuación	Sensibilidad	Especificidad	Tiempo (s)
k2 Real	-4533.472	-	-	-
k2 DEBN	-4490.557	1/6	13/14	0.086
k2 DEBN Rest	-4490.557	0/6	13/14	1.27
k2 hc	-4488.014	0/6	14/14	0.004
k2 tabu	-4488.014	0/6	14/14	0.019
k2 mmhc	-4488.014	0/6	14/14	0.005
k2 rm2 tabu	-4488.014	0/6	14/14	0.002
bde Real	-4626.819	-	-	-
bde DEBN	-4504.299	0/6	14/14	1.071
bde DEBN Rest	-4509.152	2/6	13/14	1.27
bde hc	-4493.056	0/6	14/14	0.006
bde tabu	-4493.056	0/6	14/14	0.014
bde mmhc	-4493.056	0/6	14/14	0.007
bde rm2 tabu	-4493.056	0/6	14/14	0.005
logVer Real	-4444.53	-	-	-
logVer DEBN	-4405.664	3/6	8/14	0.897
logVer DEBN Rest	-4402.791	2/6	6/14	1.20
logVer hc	-4402.791	6/6	10/14	0.023
logVer tabu	-4402.791	6/6	10/14	0.012
logVer mmhc	-4459.806	0/6	14/14	0.001
logVer rm2 tabu	-4459.806	0/6	14/14	0.004

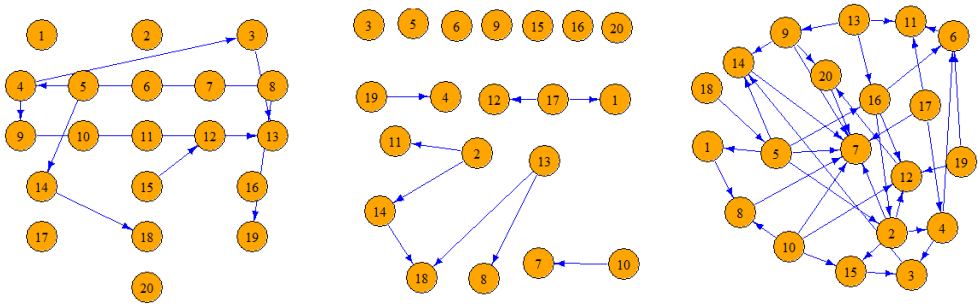
Tabla 3.3: GAD 3.

La Tabla 3.3 muestra el desempeño de los GADs obtenidos para la estructura de red bayesiana 3, donde los GADs con mayor sensibilidad y especificidad son aprendidos mediante los algoritmos Hill-Climbing y Tabu Search y la función de puntuación log-likelihood para ambos casos.

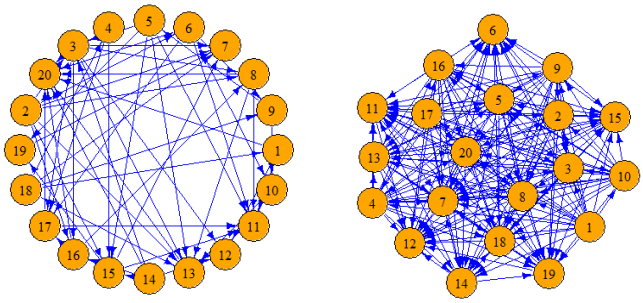
La Figura 3.11 (a) muestra el GAD real (red bayesiana co 20 vértices) del cual se simularon los datos, así como las estructuras más representativas obtenidos al hacer el aprendizaje correspondiente al GAD 4.



(a) Real. (b) DEBN k2. (c) DEBN k2 Rest.



(d) DEBN Bde. (e) DEBN Bde Rest. (f) DEBN logVer.



(g) DEBN logVer Rest. (h) HC logVer.

Figura 3.11: GAD 4.

Tipos de puntuación	Puntuación	Sensibilidad	Especificidad	Tiempo (s)
k2 Real	-12587.17	-	-	-
k2 DEBN	-12543.24	0/26	342/354	4.051
k2 DEBN Rest	-12546.58	2/26	346/354	6.56
k2 hc	-12518.54	2/26.	350/354	0.022
k2 tabu	-12517.24	2/26.	348/354	0.044
k2 mmhc	-12518.6	2/26.	351/354	0.031
k2 rm2 tabu	-12518.6	2/26.	381/354	0.021
bde Real	-12661.81	-	-	-
bde DEBN	-12574.66	0/26	345/354	4.535
bde DEBN Rest	-12563.35	0/26	345/354	6.68
bde hc	-12531.06	1/26.	354/354	0.006
bde tabu	-12531.06	1/26.	354/354	0.002
bde mmhc	-12531.06	1/26.	354/354	0.018
bde rm2 tabu	-12531.06	1/26.	354/354	0.016
logVer Real	-12437.41	-	-	-
logVer DEBN	-12258.7	3/26.	319/354.	3.758
logVer DEBN Rest	-12227.31	7/26	306/354	6.31
logVer hc	-6902.21	12/26.	178/354.	12.769
logVer tabu	-6902.21	12/26.	178/354.	6.279
logVer mmhc	-12438.19	2/26.	350/354.	0.019
logVer rm2 tabu	-12437.22	1/26.	349/354.	0.189

Tabla 3.4: GAD 4

La Tabla 3.4 muestra el desempeño de los GADs obtenidos para la estructura de red bayesiana 4, donde los GADs con mayor sensibilidad y especificidad son aprendidos mediante los algoritmos Hill-Climbing y Tabu Search y la función de puntuación log-likelihood para ambos casos.

La Figura 3.12 muestra el GAD real (red bayesiana co 100 vértices) del cual se simularon los datos para el GAD 5. Por simplicidad, en este caso, no se presentan redes bayesianas originadas con los aprendizajes pero la Tabla 3.5 muestra el desempeño de los algoritmos empleados para la tarea de investigar la estructura de la red bayesiana de la Figura 3.12.

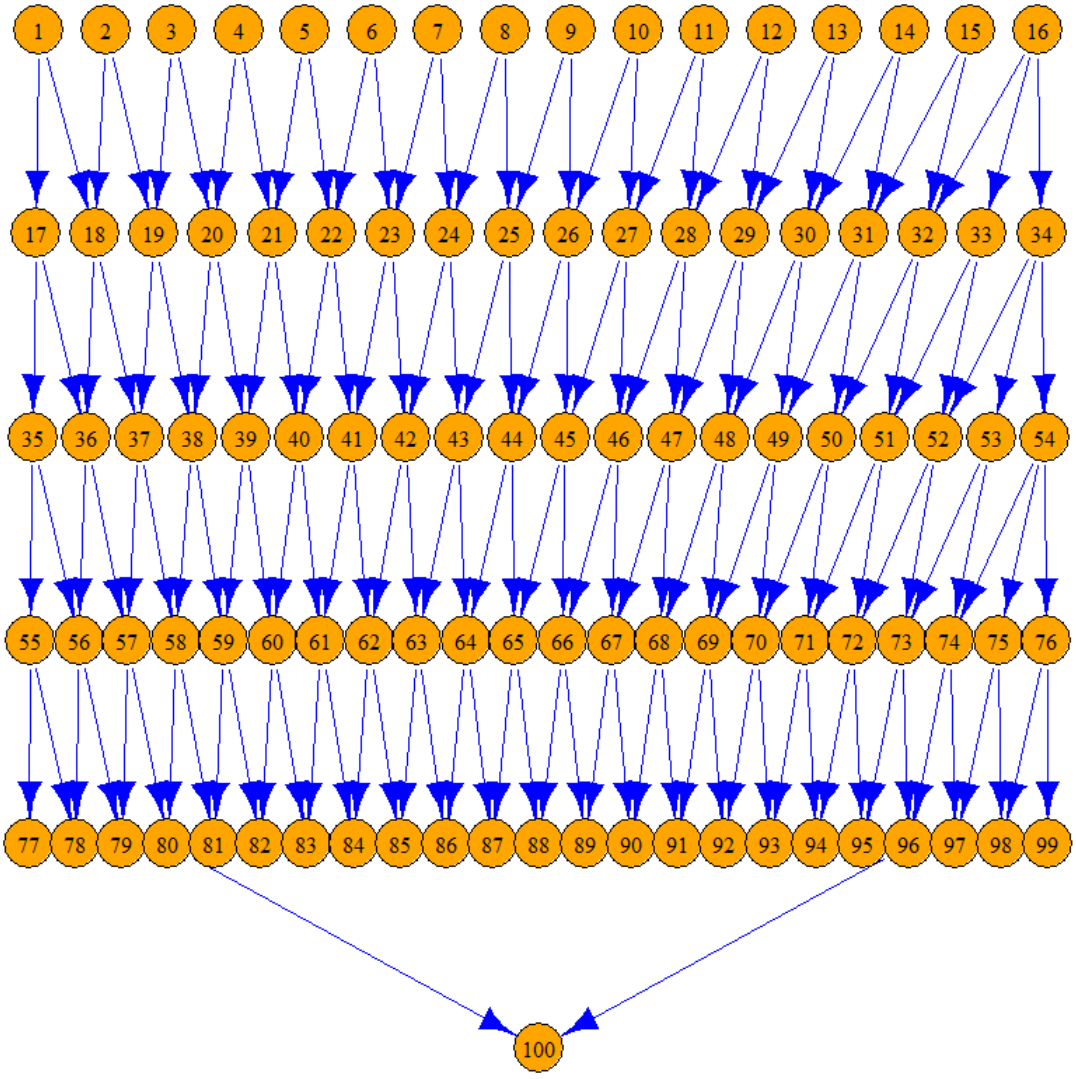


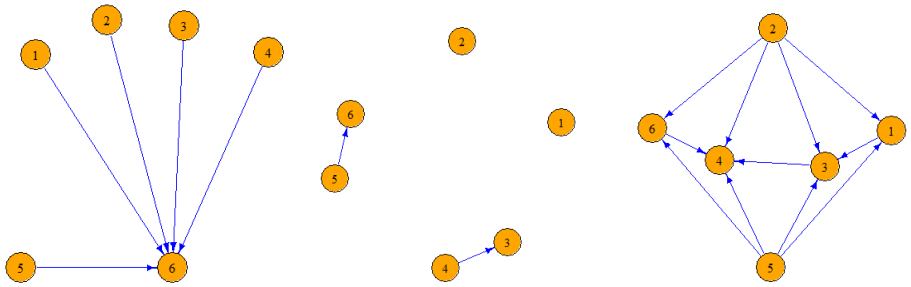
Figura 3.12: GAD 5.

Tipos de puntuación	Puntuación	Sensibilidad	Especificidad	Tiempo (s)
k2 Real	-59497.07	-	-	-
k2 DEBN	-59307.1	2/157	9666/9743	123.96
k2 DEBN Rest	-59284	2/157	9672/9743	126.04
k2 hc	-58870.94	4/157	9563/9743	1.020
k2 tabu	-58870.94	4/157	9563/9743	0.806
k2 mmhc	-59017.54	2/157	9691/9743	0.49
k2 rm2 tabu	-59011.11	1/157	9688/9743	0.189
bde Real	-59907.31	-	-	-
bde DEBN	-59521.11	0/157	9672/9743	130.2
bde DEBN Rest	-59522.01	2/157	9667/9743	131.52
bde hc	-59094.18	1/157	9702/9743	0.169
bde tabu	-59094.18	1/157	9702/9743	0.227
bde mmhc	-59100.93	1/157	9713/9743	0.366
bde rm2 tabu	-59100.7	1/157	9711/9743	0.153
logVer Real	-58647.32	-	-	-
logVer DEBN	-57579.13	3/157	9447/9743	130.6
logVer DEBN Rest	-57398.77	6/157	9464/9743	127.206
logVer hc	-6907.755	27/157	8244/9743	435.48
logVer tabu	-6907.755	27/157	8244/9743	457.86
logVer mmhc	-58514.49	2/157	9689/9743	0.36
logVer rm2 tabu	-58500.36	2/157	9685/9743	0.242

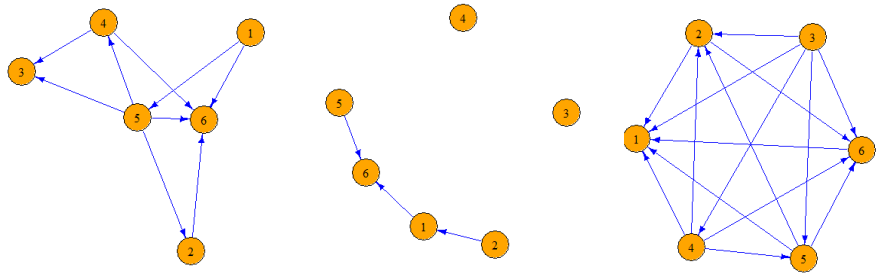
Tabla 3.5: Red bayesiana 5.

La Tabla 3.5 muestra el desempeño de los GADs obtenidos para la estructura de red bayesiana 5, donde los GADs con mayor sensibilidad y especificidad son aprendidos mediante los algoritmos Hill-Climbing y Tabu Search y la función de puntuación log-likelihood para ambos casos.

La Figura 3.13 (a) muestra el modelo BN-Cox usado por Kraisangka & Druzdzel para la base de datos *Recidivism*, conjunto de datos obtenidos de un estudio experimental de 432 prisioneros varones, que estuvieron bajo observación durante un año después de ser liberados de la prisión. El evento de interés en este análisis es volver a ser arrestado. La Figura 3.13 muestra el aprendizaje de la estructura para la base de datos *Recidivism* al considerar los riesgos: estatus financiero (no = 0, sí = 1), la raza del prisionero (otro = 0, negro = 1), experiencia laboral previa (sí = 0, no = 1), y condenas anteriores (cinco o menos = 0, más de cinco = 1). La variable de tiempo en este conjunto de datos es la semana, que es la semana en que un prisionero fue arrestado nuevamente durante el período de observación de un año (52 semanas). La variable de supervivencia está deteniendo la erradicación del estado de reclusión de un preso (detenido de nuevo = 1, no detenido de nuevo = 0).



(a) BN-Cox. (b) DEBN k2 Rest. (c) DEBN logVer.



(d) DEBN logVer Rest. (e) DEBN Bde Rest. (f) TABU logVer.

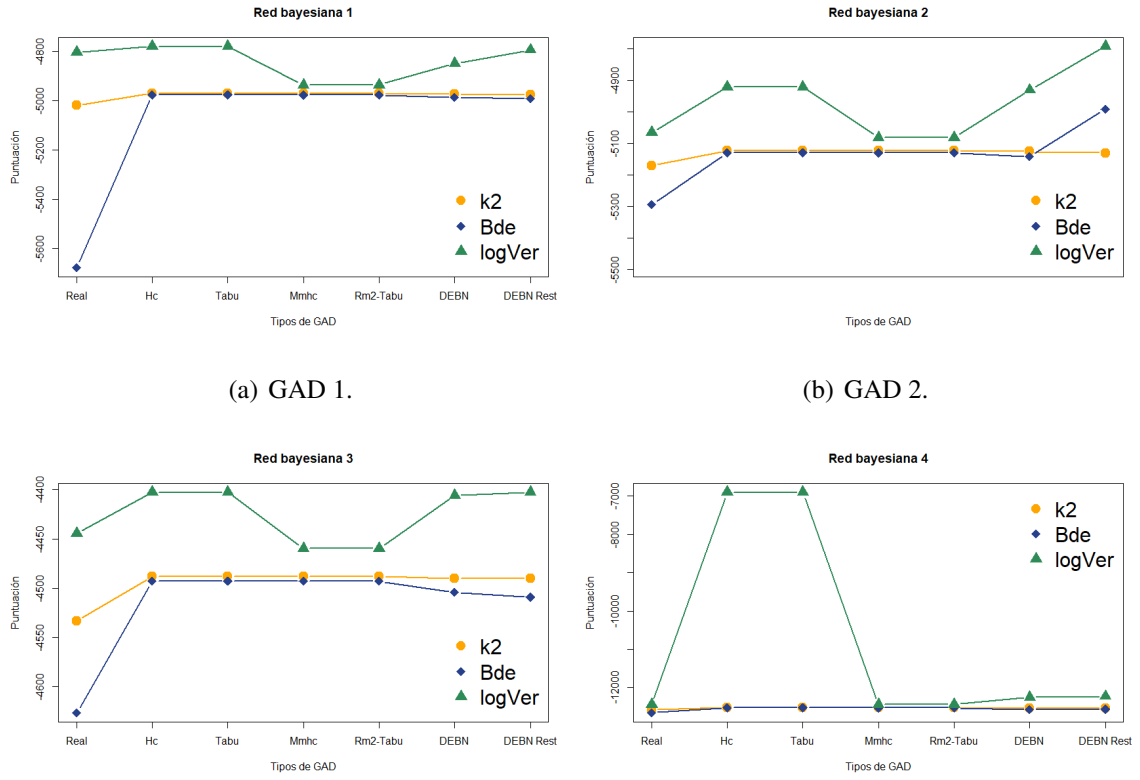
Figura 3.13: GAD BN-Cox para *Recidivism*.

Tipos de puntuación	Puntuación	Sensibilidad	Especificidad	Tiempo (s)
k2 BN-Cox	-1798.562	-	-	-
k2 DEBN	-1749.615	1/5	19/25	1.826
k2 DEBN Rest	-1746.516	1/5	24/25	1.53
k2 hc	-1745.13	1/5	23/25	0.003
k2 tabu	-1742.087	1/5	20/25	0.034
k2 mmhc	-1746.516	1/5	24/25	0.033
k2 rm2 tabu	-1746.516	1/5	24/25	0.006
bde BN-Cox	-1865.484	-	-	-
bde DEBN	-1809.811	0/5	22/25	1.417
bde DEBN Rest	-1827.529	2/5	24/25	1.68
bde hc	-1803.896	1/5	23/25	0.007
bde tabu	-1803.896	0/5	22/25	0.018
bde mmhc	-1804.643	1/5	24/25	0.007
bde rm2 tabu	-1804.643	1/5	24/25	0.004
logVer BN-Cox	-1592.284	-	-	-
logVer DEBN	-1444.777	2/5	16/25	1.524
logVer DEBN Rest	-1445.367	4/5	20/25	1.54
logVer hc	-1411.891	5/5	15/25	0.005
logVer tabu	-1411.891	4/5	14/25	0.112
logVer mmhc	-1583.833	1/5	24/25	0.006
logVer rm2 tabu	-1583.833	1/5	24/25	0.007

Tabla 3.6: GAD BN-Cox

La Tabla 3.6 muestra el desempeño de los GADs obtenidos para la estructura de red bayesiana BN-Cox, donde los GADs con mayor sensibilidad y especificidad son aprendidos mediante los algoritmos Hill-Climbing y Tabu Search y la función de puntuación log-likelihood para ambos casos.

La Figura 3.14, ilustran la puntuación obtenida por los diversos GADs obtenidos al hacer el aprendizaje de la estructura de la red bayesiana 1 (Figura 3.14 (a)), 2 (Figura 3.14 (b)), 3 (Figura 3.14 (c)), y 4 (Figura 3.14 (d)).



(a) GAD 1.

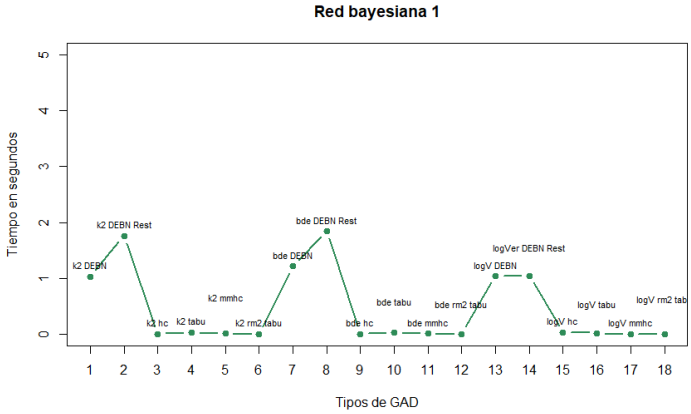
(b) GAD 2.

(c) GAD 3.

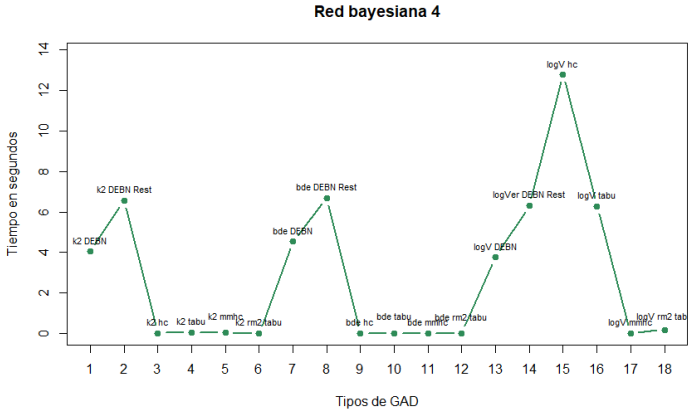
(d) GAD 4.

Figura 3.14: Puntuación

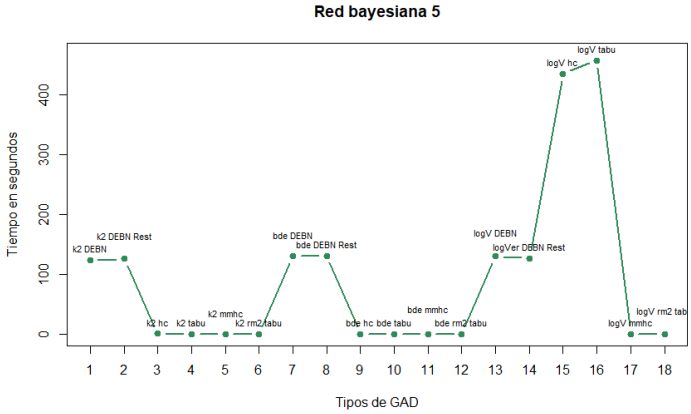
Por otro lado la Figura 3.15, muestran el tiempo en segundos de la ejecución de los algoritmos utilizados para hacer el aprendizaje de la estructura para la red bayesiana 1 (Figura 3.15 (a)), 4 (Figura 3.15 (b)) y 5 (Figura 3.15 (c)).



(a) GAD 1.



(b) GAD 4.



(c) GAD 5.

Figura 3.15: Tiempos

CAPÍTULO 4

Conclusiones y trabajo futuro

Este trabajo de investigación presenta las redes bayesianas como una herramienta alternativa para el estudio de tiempos de falla. Éstas se distinguen de los modelos de regresión estándar al tratar de manifestar no sólo la correlación entre las covariables y la variable de respuesta, sino también la correlación entre las diferentes covariables.

Con el fin de extender e intentar mejorar la idea de interpretar el modelo de riesgos proporcionales de Cox presentado por Kraisangka & Druzdzel, se propone un algoritmo genético para aprender la estructura de una red bayesiana a través de los datos.

El aprendizaje de la estructura de red bayesiana a partir de los datos puede verse como un problema de optimización en el que se debe encontrar una red bayesiana que mejor represente la distribución de probabilidad que ha generado los datos en una base de datos dada. Al tener enfrente un problema de optimización, ya varios algoritmos genéticos han sido propuestos para el aprendizaje de la estructura de red bayesiana, en particular, este trabajo presenta un algoritmo diferencial evolutivo para el estudio de la estructura de red bayesiana.

Cabe mencionar que se hizo una modificación al algoritmo diferencial evolutivo, donde el vértice que se considera como la variable de supervivencia se restringe a que si hay un arco conectado a este nodo de supervivencia, el arco siempre apunta al nodo de supervivencia. En este sentido la variable de supervivencia es siempre una "hoja" dentro de la red bayesiana. Esto es posible ya que la permutación determina la dirección en un grafo acíclico dirigido cualquiera, por lo que se restringieron las permutaciones iniciales usadas en el algoritmo diferencial evolutivo. Ésta restricción prevalece en la generación de las nuevas redes bayesianas a través de las operaciones definidas.

Se probaron tres diferentes funciones a optimizar, ademas se consideró otros cuatro algoritmos de aprendizaje de estructura de red bayesiana ya implementados en el software R. Al concluir este estudio, se realizan las siguientes observaciones

- El aprendizaje de red bayesiana a través de los datos que no es una tarea fácil y ésto es una limitante para hacer el estudio de tiempos de falla a través de redes bayesianas. Aunque en Kraisangka & Druzdzel proponen usar el modelo BN-Cox como una alternativa para el aprendizaje de la red y sus parámetros; restringen ampliamente la estructura de las redes bayesianas resultantes.

- El algoritmo Hill-Climbing tiene mayor desempeño al momento de encontrar la estructura de red bayesiana cuando se tiene una gran cantidad de vértices, sin embargo, este asigna una gran cantidad de arcos en la estructura de la red e incrementa el tiempo de ejecución cuando aumenta el número de variables.

- En los experimentos computacionales, solo se corrió una vez el algoritmo diferencial evolutivo por cada tipo de GAD, sin embargo, éste algoritmo puede obtener un mejor desempeño que el mostrado, bajo un costo computacional.

- Utilizando inferencia bayesiana en el aprendizaje de la red bayesiana el conocimiento previo por expertos puede incorporarse para potencialmente reducir los tiempos de

ajuste.

Trabajo futuro

Algunas lineas de trabajo futuro son

- Estudiar las redes bayesianas mixtas ya que consideran tanto variables aleatorias discretas como continuas.
- Incorporar técnicas de Markov chain Monte Carlo (MCMC) para realizar inferencias bayesianas en configuraciones más generales, como la mixta.
- Considerar que la estructura de relación entre variables puede cambiar a lo largo del tiempo y hacer uso de redes bayesianas dinámicas.

Referencias

Aalen, O. O., Borgan, Ø., & Gjessing, H. K. (2008). Nonparametric analysis of survival and event history data. *Survival and Event History Analysis: A Process Point of View*, 69–130.

Baioletti, M., Milani, A., & Santucci, V. (2018). Learning bayesian networks with algebraic differential evolution. In *International Conference on Parallel Problem Solving from Nature*, (pp. 436–448). Springer.

Bakker, B., Heskes, T., Neijt, J., & Kappen, B. (2004). Improving cox survival analysis with a neural-bayesian approach. *Statistics in medicine*, *23*(19), 2989–3012.

Bandyopadhyay, S., Wolfson, J., Vock, D. M., Vazquez-Benitez, G., Adomavicius, G., Elidrisi, M., Johnson, P. E., & O'Connor, P. J. (2015). Data mining for censored time-to-event data: a bayesian network model for predicting cardiovascular risk from electronic health record data. *Data Mining and Knowledge Discovery*, *29*(4), 1033–1069.

Berzuini, C., Bellazzi, R., Quaglini, S., & Spiegelhalter, D. J. (1992). Bayesian networks for patient monitoring. *Artificial intelligence in medicine*, *4*(3), 243–260.

Brest, J., Greiner, S., Boskovic, B., Mernik, M., & Zumer, V. (2006). Self-adapting control

parameters in differential evolution: A comparative study on numerical benchmark problems. *IEEE transactions on evolutionary computation*, *10*(6), 646–657.

Burke, H. B., Goodman, P. H., Rosen, D. B., Henson, D. E., Weinstein, J. N., Harrell Jr, F. E., Marks, J. R., Winchester, D. P., & Bostwick, D. G. (1997). Artificial neural networks improve the accuracy of cancer survival prediction. *Cancer*, *79*(4), 857–862.

Cheng, J. & Druzdzel, M. J. (2000). Ais-bn: An adaptive importance sampling algorithm for evidential reasoning in large bayesian networks. *Journal of Artificial Intelligence Research*, *13*, 155–188.

Collett, D. (2015). *Modelling survival data in medical research*. Chapman and Hall/CRC.

Cox, D. R. (1975). Partial likelihood. *Biometrika*, *62*(2), 269–276.

Dechter, R. & Pearl, J. (1988). Network-based heuristics for constraint-satisfaction problems. In *Search in Artificial Intelligence* (pp. 370–425). Springer.

Donat, R., Leray, P., Bouillaut, L., & Aknin, P. (2010). A dynamic bayesian network to represent discrete duration models. *Neurocomputing*, *73*(4-6), 570–577.

Druzdzel, M. J. (1999). Smile: Structural modeling, inference, and learning engine and genie: a development environment for graphical decision-theoretic models. In *Aaai/Iaai*, (pp. 902–903).

Eleuteri, A., Tagliaferri, R., Milano, L., De Placido, S., & De Laurentiis, M. (2003). A novel neural network-based survival analysis model. *Neural Networks*, *16*(5-6), 855–864.

Gerstung, M., Baudis, M., Moch, H., & Beerenwinkel, N. (2009). Quantifying cancer progression with conjunctive bayesian networks. *Bioinformatics*, *25*(21), 2809–2815.

Heckerman, D., Geiger, D., & Chickering, D. M. (1995). Learning bayesian networks: The combination of knowledge and statistical data. *Machine learning*, *20*(3), 197–243.

Jones, B., Jenkinson, I., Yang, Z., & Wang, J. (2010). The use of bayesian network modelling for maintenance planning in a manufacturing industry. *Reliability Engineering & System Safety*, *95*(3), 267–277.

Kjaerulff, U. B. & Madsen, A. L. (2008). Bayesian networks and influence diagrams. *Springer Science+ Business Media*, *200*, 114.

Klein, J. P. & Moeschberger, M. L. (1997). Survival analysis: techniques for censored and truncated data.

Koller, D. & Friedman, N. (2009). *Probabilistic graphical models: principles and techniques*. MIT press.

Kraisangka, J. & Druzdzel, M. J. (2018). A bayesian network interpretation of the cox's proportional hazard model. *International Journal of Approximate Reasoning*, *103*, 195–211.

Landoni, G., Greco, T., Biondi-Zoccai, G., Nigro Neto, C., Febres, D., Pintaudi, M., Pasin, L., Cabrini, L., Finco, G., & Zangrillo, A. (2013). Anaesthetic drugs and survival: a bayesian network meta-analysis of randomized trials in cardiac surgery. *British journal of anaesthesia*, *111*(6), 886–896.

Larranaga, P., Kuijpers, C. M. H., Murga, R. H., Inza, I., & Dizdarevic, S. (1999). Genetic algorithms for the travelling salesman problem: A review of representations and operators. *Artificial Intelligence Review*, *13*(2), 129–170.

Larrañaga, P., Murga, R., Poza, M., & Kuijpers, C. (1996). Structure learning of bayesian networks by hybrid genetic algorithms. In *Learning from Data* (pp. 165–174). Springer.

Larrañaga, P. & Poza, M. (1994). Structure learning of bayesian networks by genetic algorithms. In *New Approaches in Classification and Data Analysis* (pp. 300–307). Springer.

Lauritzen, S. L. & Spiegelhalter, D. J. (1988). Local computations with probabilities on graphical structures and their application to expert systems. *Journal of the Royal Statistical Society: Series B (Methodological)*, *50*(2), 157–194.

Lawless, J. F. (2011). *Statistical models and methods for lifetime data*, volume 362. John Wiley & Sons.

Meeker, W. Q., Escobar, L. A., & Lu, C. J. (1998). Accelerated degradation tests: modeling and analysis. *Technometrics*, *40*(2), 89–99.

Mittal, A. (2007). *Bayesian network technologies: applications and graphical models: applications and graphical models*. IGI Global.

Nielsen, T. D. & Jensen, F. V. (2009). *Bayesian networks and decision graphs*. Springer Science & Business Media.

Oliver, I., Smith, D., & Holland, J. (1987). A study of permutation crossover operators on the tsp, genetic algorithms and their applications. In *Proceedings of the Second International Conference on Genetic Algorithms*, (pp. 224–230).

Price, K., Storn, R. M., & Lampinen, J. A. (2006). *Differential evolution: a practical approach to global optimization*. Springer Science & Business Media.

R Core Team (2013). *R: A Language and Environment for Statistical Computing*. Vienna, Austria: R Foundation for Statistical Computing.

Schwarz, G. et al. (1978). Estimating the dimension of a model. *The annals of statistics*, *6*(2), 461–464.

Scutari, M. (2009). Learning bayesian networks with the bnlearn r package. *arXiv preprint arXiv:0908.3817*.

Scutari, M. & Denis, J.-B. (2014). *Bayesian networks: with examples in R*. Chapman and Hall/CRC.

Shetty, S., Song, M., & Alam, M. (2008). Data mining of bayesian network structure using a semantic genetic algorithm-based approach. In *Data Warehousing and Mining: Concepts, Methodologies, Tools, and Applications* (pp. 1081–1090). IGI Global.

Štajduhar, I. & Dalbelo-Bašić, B. (2010). Learning bayesian networks from survival data using weighting censored instances. *Journal of biomedical informatics*, *43*(4), 613–622.

Štajduhar, I., Dalbelo-Bašić, B., & Bogunović, N. (2009). Impact of censoring on learning bayesian networks in survival modelling. *Artificial intelligence in medicine*, *47*(3), 199–217.

Zangrillo, A., Musu, M., Greco, T., Di Prima, A. L., Matteazzi, A., Testa, V., Nardelli, P., Febres, D., Monaco, F., Calabrò, M. G., et al. (2015). Additive effect on survival of anaesthetic cardiac protection and remote ischemic preconditioning in cardiac surgery: a bayesian network meta-analysis of randomized trials. *PLoS One*, *10*(7), e0134264.