1

APRENDIZAJE DE UNA RED BAYESIANA ÓPTIMA Y SUS PARÁMETROS A PARTIR DE DATOS DE EXPRESIÓN GÉNICA

1.1 INTRODUCCIÓN

La identificación y el análisis de genes de interes biológico y sus interaciones son claves en diseño de fármacos, mejoramiento fisiológico de plantas e investigación de enfermedades como la oncogénesis, entre otros. Es posible analizar el comportamiento colectivo de grupos de genes usando modelos que representen la interacción entre ellos. Algunos modelos son las redes de co-expresión génica y las redes regulatorias de genes, donde se observan patrones de expresión que siguen los grupos de genes bajo ciertas condiciones biológicas y el impacto del comportamiento de un grupo sobre otros. Estas redes han sido representadas con modelos de redes Bayesianas. Su estructura y parámetros han sido aprendidos con métodos estadísticos como las simulaciones de Markov Chain MonteCarlo (MCMC) e inferencia Bayesiana.

El proposito de este capítulo es caracterizar un método para el aprendizaje de una red Bayesiana y sus parámetros a partir de datos de expresión y mostrar su funcionamiento con algunos ejemplos de expresión génica de E.coli. Primero caracterizamos un método general de aprendizaje de redes Bayesianas adaptando el algoritmo de Metropolis Hasting, Luego presentamos una versión iterativa de esta adaptación para obtener conjuntos más amplios de redes en el nivel óptimo (de aquí en adelante llamadas "redes óptimas") . Posteriormente, trataremos el tema del sobre-entrenamiento de las redes obtenidas y presentamos diferentes métodos para seleccionar entre las redes óptimas una red más apropiada para el análisis. Estos métodos son orientados hacia la obtención de una red Bayesiana minimal (llamada así porque se obtiene como intersección de redes Bayesianas) donde se priorizan las aristas más frecuentes y una red reducida donde se eliminan las aristas menos frequentes de la red minimal pero conservando la conectividad de la red. Estas redes faciliten los análisis y permitan dar con redes óptimas coherentes desde el punta de vista biológico para el caso del ejemplo con E.coli como será ilustrado en el capítulo . Finalmente proporcionamos unos comentarios conluyentes.

1.2 CARACTERIZACIÓN DE UN MÉTODO GENERAL DE APRENDIZAJE DE REDES BAYESIANAS Y SUS PARÁMETROS.

En esta sección se presenta la caracterización de un método de aprendizaje de redes Bayesianas y sus parámetros basados en el enfoque de simulaciones de Markov Chain MonteCarlo (MCMC, para mayor detalle ver [**?**] y [**?**]). Este método será parte del marco de trabajo de redes Bayesianas que se propondrá y se compone de los siguientes pasos:

1. Definir la red Bayesiana inicial e inicializar la matriz de adyacencia NxN correspondiente, siendo N el número de variables o nodos de la red Bayesiana. Por defecto, la red se inicializa como una red totalmente desconectada y se convierte en *la red Bayesiana actual*.
2. Evaluar el ajuste de los datos dada la red Bayesiana *actual*, es decir, cálcular la función de *verosimilitud* para el modelo de la red Bayesiana *actual*.
3. Buscar una estructura de red Bayesiana *candidata* en el vecindario de la red Bayesiana *actual*.
4. Evaluar la función *de verosimilitud* para la red Bayesiana *candidata*.
5. Calcular la probabilidad de aceptación de la red Bayesiana candidata dada la red Bayesiana actual.
6. Aplicar el *criterio de aceptación* de la red Bayesiana *candidata* basado en el algortimo de *Metropolis Hasting*.
7. Iterar los pasos del 3. al 6. hasta alcanzar el *número máximo de simulaciones*, parametrizado previamente.

Para efectos de la caracterización de este método, se establecieron las siguientes consideraciones para la representación del modelo de una red Bayesiana y un conjunto de muestras subyacentes de este modelo. Sea B un modelo de red Bayesiana con $B = (G, \Theta)$, donde G es un grafo acíclico dirigido (GAD) cuyos nodos corresponden a un conjunto de variables aleatorias discretas $V = \{V_1, \ldots, V_N\}$, Θ un vector, donde cada elemento representa una probabilidad condicional entre las variables aleatorias de G y $X = \{X_1, \ldots, X_M\}$ un conjunto dado de M observaciones idéntica e independientemente distribuidas *(IID),* obtenidas a partir de un experimento estadístico cuyo modelo subyacente puede ser explicado por B. Adicionalmente:

- Sea $A = \left[a_{ij}\right]$ la matriz de adyacencia para representar la estructura de una red Bayesiana, donde a_{ij}toma el valor 1 si existe una arista entre los nodos i y j de la red ó 0 en caso contrario.
- Sea $P\left(X|G, \Theta\right)$ la función de verosimilitud correspondiente al modelo de la red Bayesiana B.
- Sea $P\left(G, \Theta|X\right)$ la distribución de la probabilidad del modelo de la red Bayesiana B, posterior a las observaciones X.

A continuación, se profundiza sobre algunos conceptos y operaciones involucrados en los diferentes pasos del método de aprendizaje de redes Bayesianas propuesto en esta investigación.

1.2.1 *La red Bayesiana inicial e inicialización de la matriz de adyacencia*

Ya que la estructura de la red Bayesiana inicial G será desconexa su respectiva matriz de adyacencia A de NxN deberá inicializarse así: $a_{ij} = 0$, para $i = 1 \ldots N$ y $j = 1 \ldots N$. Es posible iniciar la simulación con una red conectada, sin embargo, la estructura inicial dada podría sesgar la caminata aleatoria a traves del espacio de búsqueda de estructuras.

1.2.2 *La función de* verosimilitud *para el modelo de la red Bayesiana*

Generalmente, se optimiza el cálculo computacional de las funciones de verosimilitud modificando la escala de su valores de evaluación usando la función logaritmo natural, en nuestro caso se remplaza $P(X|G, \Theta)$ por $ln(P(X|G, \Theta))$. La transformación a escala logarítmica mantiene el orden ($a < b$ implica $ln\, a < ln$ b) y con eso los puntos óptimos. La transformación facilita computacionalmente los calculos ya que productos de variables se convierten en sumas, especialmente conveniente en el cálculo de las distribuciones conjuntas en redes Bayesianas, donde un producto de muchos valores menores que 1 puede resultar en valores muy cercanos a 0 con el riesgo de underflow computacional.

1.2.3 *El vecindario de la red Bayesiana* actual

La vecindad entre dos estructuras de red en el espacio de búsqueda está dada en términos de los siguientes operadores: *(i) agregar una arista* a la estructura de la red actual, *(ii) quitar una arista* a la estructura de red actual y *(iii) quitar una arista* a la estructura de la red actual y a la red resultante *agregar una arista* (*doble operación*). Este enfoque está basado en el trabajo de [**?**]. Sin embargo, se modificó este enfoque para poder determinar de antemano el número de vecinos de una estructura y seleccionar aleatoriamente una de ellas. Esto evita explorar algunas estructuras candidatas que al analizarlas presentan ciclos; lo que podría pasar con mayor frecuencia en la medida que se aumentan las aristas en la red. Por eso modificamos la operación *(iii)* que originalmente consiste en invertir una arista.. El procedimiento estocástico que aplicamos hace conteo del número de estructuras para cada operador (i), (ii) y (iii) y aleatóriamente selecciona una. La red Bayesiana *candidata* será obtenida aplicando uno de los tres operadores

anteriores y se denotará como $B^{'} = \left(G^{'}, \Theta^{'}\right)$ y su matriz de adyacencia $A^{'}$. A contiuación, se detallará cada una de las operaciones.

(i) Obtención de las posibles estructuras de redes acíclicas después de agregar una arista.

La matriz de adjacencia A indica los caminos de longitud 1 entre cada par de nodos i, j donde $a_{ij} = 1$, y se obtienen los caminos de longitud $2, \ldots, N-1$ calculando las potencias de A, es decir, $A^2, \ldots, A^{N-1}$. Al sumarestas matrices con la matriz identidad I de dimensión NxN, se obtiene una matriz que indica (con un 1 en la posición (i, j)) la existencia de un camino de cualquier longitud entre i y j.Por eso, si se toman las traspuestas de las matrices potencias y se suman con la matriz identidad I, la matriz resultante C indica (con un 1 en la posición (i, j)) la existencia de un camino de cualquier longitud entre j y i:

$$C = [c_{ij}], c_{ij} = \begin{cases} 1 & Existe \ camino \ \ entre \ \ j \ , i \\ 0 & No \ \ existe \ \ camino \end{cases}$$

Así, al agregar una arista entre los nodos i y j donde $c_{ij} = 1$, se crea un bucle entre ellos. Por consiguiente, solo se debe tener en cuenta los nodos i y j donde $c_{ij} = 0$ al momento de agregar nuevas aristas, si se quiere generar una estructura acíclica. El cálculo de C está dado de la siguiente manera:

$$C = I + Transpuesta(A) + Transpuesta(A^2) + \ldots + Transpuesta(A^{N-1})$$

y por eso el número de posibilidades de agregar una arista (manteniendo la red acíclica) es igual al número de o's en C.

(ii) Eliminación de una arista.

Se debe tomar un par de nodos i, j tales que $a_{ij} = 1$ e invertir su valor $a_{ij} = 0$. El número de posibilidades de eliminar una arista entonces es igual al número de 1's en A.

(iii) La doble operación

Consiste en aplicar el operador (ii) y luego aplicar el operador (i) a la estructura resultante . El número de posibles estructuras vecinas acíclicas está dado por las suma de o's en las matrices C^* que resultan de cada posible eliminación de una arista de la estructura original.

1.2.4 *Evaluación de la función* de verosimilitud *para la red Bayesiana* candidata

Teniendo en cuenta que la estructura de la red Bayesiana *candidata* cambia ligeramente con respecto a la red Bayesiana *actual* desde donde es generada, la evaluación de la función de *verosimilitud* para la red Bayesiana *candidata* $P\left(X|G^{'},\Theta^{'}\right)$ se optimizó empleando una técnica de *cache* guardando los valores de la función calculados para la red Bayesiana *actual* y teniendo en cuenta la transformación de la función logaritmo natural ln, que se mencionó anteriormente. En ese orden ideas se calcula la diferencia Δ entre los valores de evaluación de la función de la red Bayesiana *actual versus* la *candidata*, así, $\Delta = ln\left(P\left(X|G^{'},\Theta^{'}\right)\right) - ln\left(P\left(X|G,\Theta\right)\right)$, en este caso para la función de verosimilitud.

1.2.5 *La probabilidad de aceptación de la red Bayesiana candidata dada la red Bayesiana actual*

Tomando como base el algoritmo de *Metropolis Hasting (MH) [**?**], se calcula el factor* α para efectos del criterio de aceptación que nos permite evaluar la red Bayesiana *candidata* de la siguiente manera, empleando la evaluación de la función de verosimilitud:

$$\alpha = \frac{P\left(X|G^{'},\Theta^{'}\right)P\left(G,\Theta|G^{'},\Theta^{'}\right)}{P\left(X|G,\Theta\right)P\left(G^{'},\Theta^{'}|G,\Theta\right)}$$

$P\left(G^{'},\Theta^{'}|G,\Theta\right)$ denota la probabilidad de la transición de la red Bayesiana *actual* a la red Bayesiana *candidata* $G,\Theta \rightarrow G^{'},\Theta^{'}$ y se calcula así:

$$P\left(G^{'},\Theta^{'}|G,\Theta\right) = \frac{1}{númeroDeEstructurasVecinas\left(G\right)}$$

, donde $númeroDeEstructurasVecinas\left(G\right)$ es el conteo del número de estructuras posibles al aplicar los operadores *(i), (ii) y (iii)*, y ya que aleatóriamente se selecciona una estructura, tal probabilidad será 1 dividido por dicho conteo. Siguiendo un razonamiento similar se llega al siguiente cálculo:

$$\beta = \frac{P\left(G,\Theta|G^{'},\Theta^{'}\right)}{P\left(G^{'},\Theta^{'}|G,\Theta\right)} = \frac{númeroDeEstructurasVecinas\left(G\right)}{númeroDeEstructurasVecinas\left(G^{'}\right)}$$

, y así se tiene que criterio α será entonces:

$$\alpha = \frac{P\left(X|G^{'},\Theta^{'}\right)}{P\left(X|G,\Theta\right)} * \beta$$

Finalmente, teniendo en cuenta la transformación de la función a logaritmo natural el cálculo de la probabilidad de aceptación de la red Bayesiana *candidata,* se calculó de esta manera:

$$ln\left(\alpha\right) = ln\left(P\left(X|G^{'},\Theta^{'}\right)\right) - ln\left(P\left(X|G,\Theta\right)\right) + ln\left(\beta\right)$$

1.2.6 *Aplicación del* criterio de aceptación *de la red Bayesiana* candidata *basado en el algoritmo de* Metropolis Hasting

El paso final del algoritmo MH es evaluar el criterio de aceptación. La evaluación en el algoritmo de *Metropolis Hasting* se hace para $r = min(1,\alpha)$ y un aleatorio $u \sim U\left(0,1\right)$: se acepta la transición $G,\Theta \rightarrow G^{'},\Theta^{'}, si\left(u < r\right)$, es decir:

$$G \rightarrow G^{'},\Theta \rightarrow \Theta^{'}; si\left(u < r\right)$$

Para ajustar lo anterior al cálculo logarítmico se hace:

$$ln\left(r\right) = min\left(0, ln\left(\alpha\right)\right)$$

.
Y se acepta la transición a la red Bayesiana *candidata*, si $ln(u) < ln\_r$ para un aleatorio u:

$$G \rightarrow G^{'},\Theta \rightarrow \Theta^{'};\ si\left(ln\left(u\right) < ln\left(r\right)\right)$$

1.3 CASO DISCRETO

Ya que las variables w_i de expresión del gen i son continuas, se hace necesario un procedimiento adicional de discretización. Para la variable aleatoria continua w_i se define su discretización en los valores (bajo 0, mediano 1, alto 2) con la variable aleatoria discreta v_i, tal que:

$$v_i = \begin{cases} 0 & ,si \quad w_i < Q_{i,\frac{1}{3}} \\ 1 & ,si \quad Q_{i,\frac{1}{3}} \leq w_i < Q_{i,\frac{2}{3}} \\ 2 & ,si \quad w_i \geq Q_{i,\frac{2}{3}} \end{cases}$$

donde $Q_{i,\frac{1}{3}}$ y $Q_{i,\frac{2}{3}}$ son $\frac{1}{3}$ y $\frac{2}{3}$ cuantiles de la variable w_i.

1.3.1 *Verosimilitud*

En una red Bayesiana cada variable aleatoria v_i tiene asociada una tabla de probabilidad condicional (TPC). El número de filas en TPC de la variable v_i es $q_i = 3^{|Pa(v_i)|}$ porque discretizamos en 3 valores.

Cada entrada en esta tabla asigna a la k-ésima combinación de valores de las variables padre de v_i, la probabilidad $\theta_{i,j,k}$ donde $v_i = j$ para $j = 0, 1, 2$. Por lo tanto, una red Bayesiana para una estructura G será parametrizada así: $\Theta = \{\theta_{ijk} > 0 : \sum_j \theta_{ijk} = 1\}$.

Dadas N observaciones independientes $X = \{X_1, ..., X_N\}$ obtenidas de la distribución conjunta (ver la ecuación **??**), estadístico suficiente para Θ es el conjunto de conteos $\{N_{ijk}\}$ donde N_{ijk} es el número de veces que $v_i = j$ dado que las variables $Pa(v_i)$ toman los valores especificados en la $k - esíma$ entrada de la TPC de v_i (para mayor detalle ver el trabajo de Ellis y Wong en [**?**]). En ese orden de ideas los conteos $\{N_{ijk}, j = 0, 1, 2\}$ siguen una distribución *multinomial* con parámetros $N_{i.k} = N_{i0k} + N_{i1k} + N_{i2k}$ (intentos) y $\{\theta_{i0k}, \theta_{i1k}, \theta_{i2k}\}$ (vector de probabilidad). Por lo tanto, la función de verosimilitud del modelo de la red Bayesiana para el caso discreto es un producto de multinomiales:

$$P(X|G, \Theta) = \prod_{i=1}^{d} \prod_{k=1}^{q_i} Multinomial(N_{i.k}, \theta_{i0k}, \theta_{i1k}, \theta_{i2k}) \tag{1.1}$$

donde $Multinomial(N_{i.k}, \theta_{i0k}, \theta_{i1k}, \theta_{i2k}) = \binom{N_{i.k}}{N_{i0k}, N_{i1k}, N_{i2k}} \theta_{i0k}^{N_{i0k}} . \theta_{i1k}^{N_{i1k}} . \theta_{i2k}^{N_{i2k}}$.

1.3.2 *Posterior*

En la teoría de probabilidad Bayesiana, si las distribuciones *posteriores* $P(\Theta|X)$ están en la misma familia que la distribución de probabilidad *prior* $P(\Theta)$, la prior y la posterior son llamadas distribuciones conjugadas y la prior es llamada una prior conjugada para la función de verosimilitud (para mayor detalle ver los trabajos de [**?**, **?**, **?**]), volviendo al caso para una verosimilitud multinomial con parámetros $N_{i.k}$ (intentos) y $\{\theta_{i0k}, \theta_{i1k}, \theta_{i2k}\}$ (vector de probabilidad) según la teoría Bayesiana, la distribución prior conjugada es una *Dirichlet* con hiperparámetros $\alpha_{i.k} = \{\alpha_{i0k}, \alpha_{i1k}, \alpha_{i2k}\}$ y la posterior con hiperparámetros $\{N_{i0k} + \alpha_{i0k}, N_{i1k} + \alpha_{i1k}, N_{i2k} + \alpha_{i2k}\}$. Por consiguiente la posterior para el modelo de red Bayesiana es un producto de *Dirichlets* así:

$$P(G, \Theta|X) = \prod_{i=1}^{d} \prod_{k=1}^{q_i} Diritchlet(N_{i0k} + \alpha_{i0k}, N_{i1k} + \alpha_{i1k}, N_{i2k} + \alpha_{i2k}) \tag{1.2}$$

donde

$$Diritchlet(N_{i0k} + \alpha_{i0k}, N_{i1k} + \alpha_{i1k}, N_{i2k} + \alpha_{i2k}) =$$

$$\frac{\theta_{i0k}^{N_{i0k}+\alpha_{i0k}-1} . \theta_{i1k}^{N_{i1k}+\alpha_{i1k}-1} . \theta_{i2k}^{N_{i2k}+\alpha_{i2k}-1}}{\beta(N_{i0k} + \alpha_{i0k}, N_{i1k} + \alpha_{i1k}, N_{i2k} + \alpha_{i2k})},$$

$$\theta_{i0k} \approx E[\theta_{i0k}] = \frac{N_{i0k} + \alpha_{i0k}}{(N_{i0k} + \alpha_{i0k} + N_{i1k} + \alpha_{i1k} + N_{i2k} + \alpha_{i2k})},$$

$$\theta_{i1k} \approx E[\theta_{i1k}] = \frac{N_{i1k} + \alpha_{i1k}}{(N_{i0k} + \alpha_{i0k} + N_{i1k} + \alpha_{i1k} + N_{i2k} + \alpha_{i2k})},$$

$$\theta_{i2k} \approx E[\theta_{i2k}] = \frac{N_{i2k} + \alpha_{i2k}}{(N_{i0k} + \alpha_{i0k} + N_{i1k} + \alpha_{i1k} + N_{i2k} + \alpha_{i2k})}$$

y

$$\beta(N_{i0k} + \alpha_{i0k}, N_{i1k} + \alpha_{i1k}, N_{i2k} + \alpha_{i2k}) = \frac{\Gamma(N_{i0k} + \alpha_{i0k})\Gamma(N_{i1k} + \alpha_{i1k})\Gamma(N_{i2k} + \alpha_{i2k})}{\Gamma(N_{i0k} + \alpha_{i0k} + N_{i1k} + \alpha_{i1k} + N_{i2k} + \alpha_{i2k})}.$$

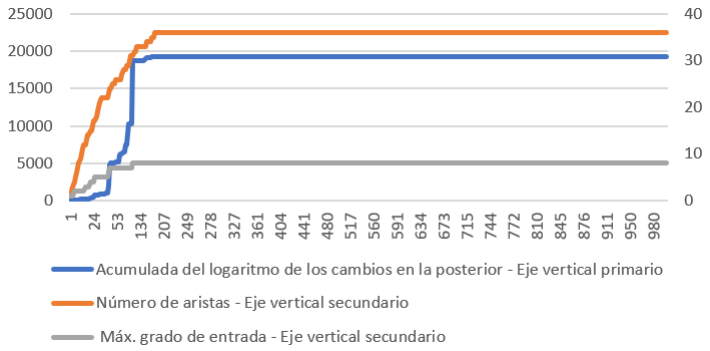
1.3.3 *Complejidad*

Como se mencionó previamente en el estado del arte, la tarea de encontrar una estructura de red que optimice la función de *scoring* es un problema de optimización combinatoria, y es conocido como un problema de complejidad computacional NP-hard [**?**]. Para ilustrar discutiremos el cálculo del resultado de la sección anterior θ_{i0k},θ_{i1k} y θ_{i2k}. Ya que k corresponde a la k-ésima combinación de valores de las variables padre de v_i y su calculo estará dado por $k = q_i = 3^{|Pa(v_i)|}$, donde se ve que el cálculo computacional además de estar afectado por el número de variables aleatorias i , también estrá afectado por la cantidad de padre de v_i. Así la complejidad que se observa aquí es $O(3^{N-1})$, donde 3 es el número de cuantiles en la discretización y $N-1$ el máximo número de padres en lo nodos de la red.

1.4 SIMULACIÓN ITERATIVA

La simulación MCMC realiza una caminata aleatoria en el espacio de búsqueda de modelos bayensianos guiada por el criterio de optimización. Por naturaleza del proceso, los modelos obtenidos giran alrededor de un óptimo local y tienden a ser repetidos al final de la simulación. Para explorar mejor el espacio de búsqueda se aplica un algorimtmo iterativo que repite varias veces la simulación MCMC a partir de la red desconectada, para lograr que se puedan tomar rutas alternas y así encontrar nuevas estructuras de red óptima. La iteración para, si sólo pocas nuevas soluciones son encontradas. Formalmente, la condición de parada del algoritmo iterativo de simulación MCMC es no encontrar un porcentaje mayor que P redes nuevas. En todas las experimentaciones de los casos de estudio que siguen se emplean la simulación iterativa que aplica 1000 pasos en cada iteración y para al encontrar solo $P = 10\,\%$ o menos soluciones nuevas.

Figura 1.1: Simulación MCMC con 1,000 iteraciones para el cluster de trrasnporte de maltose en E.coli. En esta fígura puede verse el número de aristas, el máximo grado de entrada y la función acumulada del cambio en el logaritmo de la posterior de cada red procesada.



Ejemplo de la aplicación de la simulación iterativa

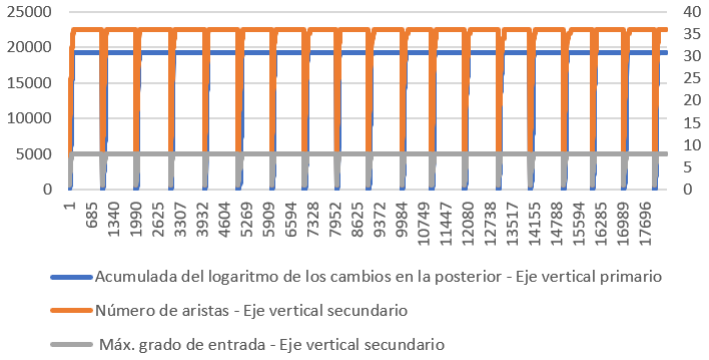
Para ilustrar un poco la idea veremos una comparación entre la simulación MCMC y la simulación iterativa con datos de expresión génica de E.coli. Tomaremos uno de los cluster identificados en el capítulo anterior (**??**) cuando se construyo la red de co-expresión génica. El cluster esta compuesto por los genes: *malQ, malT, lamB,malG,malP,malM,malE,malF y malK* (ver en la tabla **??**, del cluster 34-darkmagenta).

La simulación MCMC se realizó con 1,000 repeticiones y se obtuvieron 25 modelos distintos capturados desde 418 iteraciones que se encontraron en el rango del nivel óptimo ver la figura 1.1. En esta fígura también puede verse el número de aristas, el máximo grado de entrada y la función acumulada del cambio en el logaritmo de la posterior de cada red procesada durante la simulación MCMC. Aquí también puede verse las similitudes en estas características de las redes que se encuentran en un rango del nivel óptimo.

Por otra parte, si realizamos el mismo ejercicio con el cluster de transporte de maltose de E.coli pero esta vez con el algoritmo dse simulación iterativa encontramos que se realizan 18,430 iteraciones y se obtienen 102 modelos diferentes, es decir, 77 nuevos modelos. De la misma manera que para la simulación MCMC los resultados del experimento se muestran en la figura 1.2. En la figura puede verse tambíen que después de 18 simulaciones MCMC de 1,000 repeticiones se alcanza la condición de parada (encontrarmenos 10 % nuevas redes) y como en cada simulación MCMC que inicia se reinician de nuevo todas las carácteristicas análizadas (Acumulada, número de aristas y máx. grado de entrada).

Por lo tanto, el algoritmo de sinulación iterativa nos permite encontrar un número mayor de redes Bayesianas que se encuentran en el rango del nivel óptimo segun la función acumulada del cambio en el logaritmo de la posterior de cada red procesada.

Figura 1.2: Simulación iterativa con 18,470 iteraciones para el cluster de trrasnporte de maltose en E.coli. En esta fígura puede verse el número de aristas, el máximo grado de entrada y la función acumulada del cambio en el logaritmo de la posterior de cada red procesada.



1.5 SOBRE-ENTRENAMIENTO

La tendencia del aprendizaje de redes Bayesianas al *sobre-entrenamiento* es evidente, ya que independiente de las funciones génicas anotadas en la ontología del gen (ver [**?**]) la red resultante relaciona el mayor número de genes posibles, por su naturaleza a optimizar el puntaje de evaluación (del inglés score, en este caso el la función logaritmo natural de la posterior). El siguiente ejemplo ilustra esta situación:

Ejemplo para el sobre-entrenamiento

Se emplearon datos de expresión génica de E.coli tomados de M^{3D} (del inglés Many Microbe Microarrays Database), donde se seleccionaron aleatoriamente 5 clusters para control y 1 cluster para análisis (transporte de lactosa del inglés lactose transport) para un total de 16 genes del genoma de E.coli, que se listan a continuación con sus respectivas anotaciones de ontología del gen (las anotaciones GO fueron tomadas de Ibarra en [**?**]):

- Lactose transport:
  lacA – galactoside O-acetyltransferase monomer
  lacI – LacI DNA-binding transcriptional repressor
  lacY – lactose / melibiose:H+ symporter LacY
  lacZ – β-galactosidase monomer
- Global regulators:
  rpoD – RNA polymerase, sigma 70 (sigma D) factor
  hns – H-NS DNA-binding transcriptional dual regulator
  crp – CRP transcriptional dual regulator
- Zinc ion transport:
  znuA – Zn2+ ABC transporter - periplasmic binding protein

- Sodium ion transport:
  ttdA – L-tartrate dehydratase, α subunit
  ttdB – L-tartrate dehydratase, β subunit
  ttdR – Dan
  ttdT – tartrate:succinate antiporter
- Phosphorelay signal transduction system:
  zraP – zinc homeostasis protein
  zraR – ZraR transcriptional activator
  zraS – ZraS sensory histidine kinase
- Response to arsenic-containing substance:
  arsB – ArsB

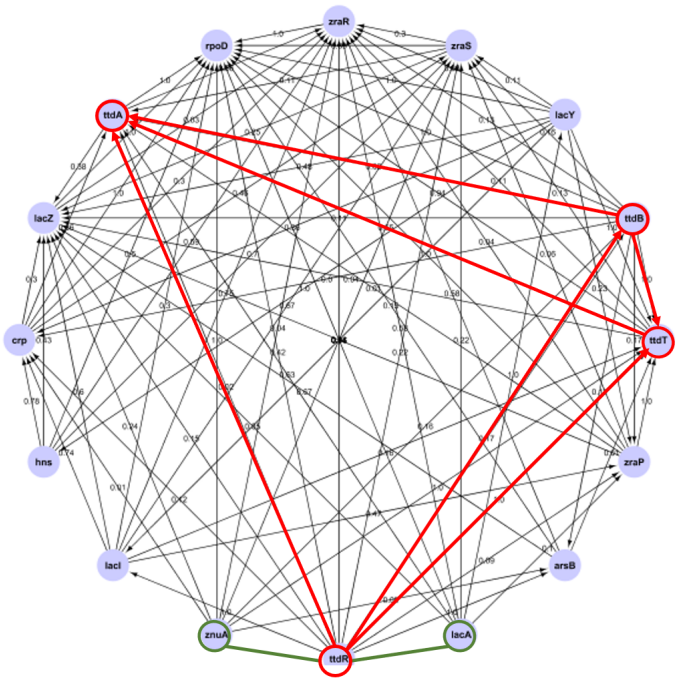
Se aplicó el proceso de aprendizaje de redes Bayesianas a este conjunto de genes, usando una simulación de 90,000 pasos. Se muestra uno de los modelos resultantes en la figura 1.3.Las etiquetas en las aristas corresponden a la frecuencia de ocurrencia en el conjunto de redes Bayesianas dentro del rango del nivel óptimo. Los nodos fueron etiquetados con el gen que representan.

La tendencia del aprendizaje de redes Bayesianas al *sobre-entrenamiento* es evidente: Se observa un gran número de aristas. Algunas de ellas pueden significar relaciones causales importantes, como por ejemplo las dependencias entre los genes del transporte de sodio (ttdA, ttdB, ttdR, ttdT) resaltados en la figura 1.4a en rojo). Otras aristas se pueden agregar porque se encuentra estadísticamente una dependencia débil entre dos nodos sin que exista una relación causal simplemente porqué es posible y mejora el *score* del modelo. Un caso típico seria la situación donde una red causal tiene 2 nodos padre; el proceso de optimización agregará aquella arista entre los padres que mejora más el score. Se verifica en la figura 1.4a que los nodos vecinos de ttdR, son nodos padres y el score del modelo se mejora si se agregan las aristas marcadas en verde con cualquier orientación. De la misma manera se comporta la red en caso de un nodo que realmente es completamente independiente del resto de la red. Es posible conectar este nodo con cualquier otro nodo de la red, sea como relación padre→hijo o hijo→padre, dando un aumento del score, probablemente considerable, sin ser originado de relaciones verdaderas, lo que se quiere ilustrar con la figura 1.4b. En general se puede decir que el proceso de optimización agrega todas las posibles aristas aunque el mejoramiento del score sea sólo pequeño.

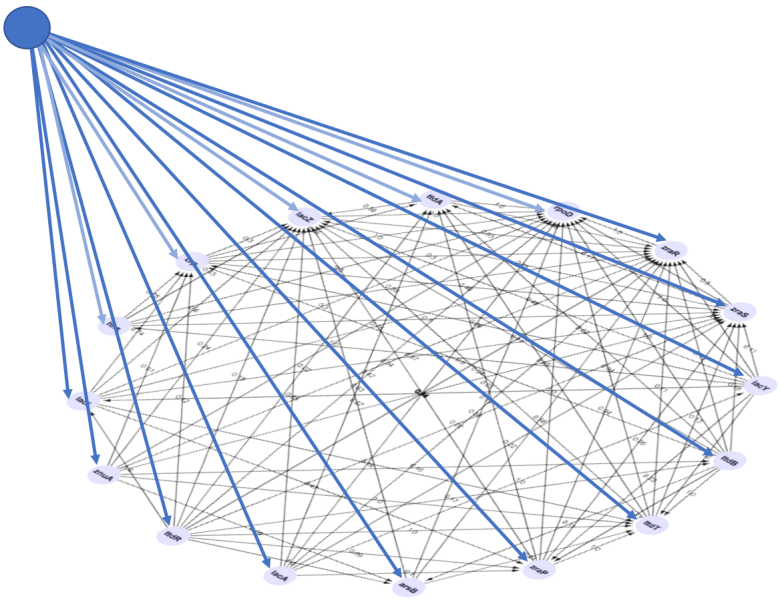
Estas observaciones han conducido a dos recomendaciones importantes:

1. Es importante aplicar el análisis Bayesiano sólo a grupos de genes que se relacionan en alguna manera causalmente. Por eso nos restringimos a clusters co-expresados.
2. Para contrarrestar la tendencia del proceso de optimización de insertar aristas, hay que buscar maneras de filtrar las que tienen

Figura 1.3: Tendencia a sobre-entrenamiento en el aprendizaje cuando se incluyen genes de diferentes clusters de E.coli en una misma simulación MCMC.



(a) Resaltando los genes del transporte de sodio (ttdA, ttdB, ttdR, ttdT).



(b) Adición de aristas en un nodo completamente independiente del resto de la red.

significado biológico de aquellas que no tienen importancia. A esta necesidad respondemos con la introducción de la red minimal y la red reducida que se describen más adelante.

1.6 CONSTRUCCIÓN DE REDES PARA EL ANÁLISIS

Todas las redes en el nivel óptimo que genera simulación iterativa pueden ser consideradas representaciones posibles de las relaciones causales entre los genes involucradas. Dada la gran cantidad de estas redes, es imperativo encontrar mecanismos que faciliten el análisis biológico. Una estrategia evidente es tener en cuenta las frecuencias con la cual una arista aparece en todos los modelos. Esto conduce a la introducción de las redes ponderadas con la cual se pueden resaltar las dependencias más importantes entre los genes considerados (*esta introducción fue sugerida por el director del laboratorio de biología de sistemas regulatorios en UNAM - Julio Augusto Freyre Gonzalez, durante una estancia doctoral*). Sin embargo, la red ponderada no es necesariamente una red Bayesiana, ya que al considerar todas las aristas que ocurren en las redes óptimas, se pueden producir ciclos. Por ende, al restringirse a redes ponderadas, se pierde la posibilidad de analizar las dependencias causales.

Una posibilidad de tener en cuenta las frecuencias es reconstruir una red Bayesiana por un proceso que agrega sucesivamente aristas respetando la frecuencia y manteniendo la red acíclica. La red resultante es la que llamamos red Bayesiana minimal, ya que se genera como intersección de conjuntos siempre más pequeños de redes Bayesianas y en consecuencia es por si mismo una red Bayesiana.

Como se verá más adelante, la red Bayesiana mínimal es siempre una red completa en el sentido que contiene el número máximo de aristas. Para más claridad en el análisis tratamos a reducir esta red, eliminando una por una las aristas menos frecuentes mientras que la red se mantiene conectada, obteniendo la red que llamamos red Bayesiana reducida.

1.6.1 *Red Ponderada (del inglés Weighted network)*

Para decidir cuando una red se considera óptima, se requiere definir lo que significa computacionalmente "el nivel óptimo" (visualmente es muy claro, observe por ejemplo la figura 1.2). Se decidió describir el nivel óptimo por un rango de valores $[minLogPosterior, maxLogPosterior]$ de la funcion de evaluación LogPosterior. Sólo las redes con un score dentro de este rango están en el nivel óptimo (las llamadas redes óptimas) son consideradas en la ponderación. La simulación iterativa fue usada para definir el rango:

- Se capturaron en cada iteración los valores del logaritmo de la distribución posterior de las redes encontradas y se determinó el valor máximo de la iteración, obteniendo el conjunto de los valores máximos de las iteraciones: $\{maxLogPos_1, .., maxLogPos_M\}$, siendo M la última iteración.
- Para determinar el valor de *maxLogPosterior* se seleccionó el valor *mayor* del conjunto de valores máximos análizados anteriormente: $maxLogPosterior = max\{maxLogPos_1, .., maxLogPos_M\}$
- Para determinar el valor de *minLogPosterior* se seleccionó el valor *menor* del conjunto de valores máximos análizados en el punto anterior: $minLogPosterior = min\{maxLogPos_1, .., maxLogPos_M\}$.

Una vez que se obtiene el conjunto de redes optimas, es decir, las redes con score en el rango $[minLogPosterior, maxLogPosterior]$, se descartan las redes repetidas y se obtienen los pesos de la siguiente manera:

- Se cuentan las ocurrencias de cada arista a lo largo del conjunto de redes óptimas
- Para cada arista, se divide el conteo realizado en el punto anterior sobre la cardinalidad del conjunto de redes óptimas.

Finalmente, se define la *red ponderada* como la red constituida de las aristas y los pesos obtenidos por el procedimiento descrito anteriormente.Bajo el supuesto que los genes incluidos en el estudio están relacionados biológicamente, es de esperar que las aristas más frecuentes reflejan las relaciones causales más importantes. Este supuesto se confirma en la experimentación.

Una desventaja de la red ponderada es que no es Bayesiana. Por eso, un análisis causal a través de la red causal es muy limitado. Sin embargo podemos basarnos en la frecuencia de las aristas para construir la red Bayesiana minimal como se muestra en la sección siguiente.

Ejemplo de la red ponderada

Basado en el ejemplo de la simulación iterativa figura 1.2, se obtuvo el nivel óptimo en el rango [19254,19, 19274,98], ver figura 1.1.

Al considerar las redes del nivel óptimo y descartando las redes repetidas, quedaron en total 102 redes óptimas, de las cuales se obtuvo la red ponderada del cuadro 1.2.

Se observa que la red no queda Bayesiana, porque en el ejemplo se forman ciclos entre cada gen y el resto. Por ejemplo, malT se relaciona con lamB con un peso de 0,81 y viceversa con un peso de 0,19.

Sin embargo, de la red ponderada podemos apreciar cuales de las aristas son muy frecuentes y concluir a una posible relación entre ellas. Por ejemplo, si observamos los pesos de las aristas desde malT hacia el resto vemos que sus pesos oscilan entre 0,66 y 0,97 lo cual es

Cuadro 1.1: Rango del nivel óptimo para el ejemplo de la simulación iterativa con el cluster de transporte de maltose en E.coli. La columna Máx log-posterior corresponde al valor máximo de la función acumulada del logaritmo de los cambios en la posterior para cada intervalo de la simulación.

Iteración inicial	Iteración final	Máx log-Posterior
1	1000	19265.78
1001	2000	19272.74
2001	3000	19271.11
3001	4000	19274.36
4001	5000	19272.35
5001	6000	19268.84
6001	7000	19274.47
7001	8000	19274.90
8001	9000	19274.98
9001	10000	19266.96
10001	11000	19272.29
11001	12000	19271.63
12001	13000	19268.76
13001	14000	19267.32
14001	15000	19272.98
15001	16000	19274.97
16001	17000	19254.19
17001	18000	19270.68
18001	18430	19261.43

Cuadro 1.2: Matriz de adyacencia de la red ponderada para el ejemplo de la simulación iterativa con el cluster de transporte de maltose en E.coli.

	lamB	*malE*	*malF*	*malG*	*malK*	*malM*	*malP*	*malQ*	*malT*
lamB	0	0.67	0.52	0.36	0.59	0.75	0.54	0.15	0.19
malE	0.33	0	0.44	0.19	0.48	0.38	0.22	0.21	0.11
malF	0.48	0.56	0	0.31	0.65	0.58	0.27	0.35	0.15
malG	0.64	0.81	0.69	0	0.80	0.83	0.76	0.25	0.21
malK	0.41	0.52	0.35	0.20	0	0.58	0.15	0.17	0.12
malM	0.25	0.62	0.42	0.17	0.42	0	0.20	0.07	0.03
malP	0.46	0.78	0.73	0.24	0.85	0.80	0	0.25	0.10
malQ	0.85	0.79	0.65	0.75	0.83	0.93	0.75	0	0.34
malT	0.81	0.89	0.85	0.79	0.88	0.97	0.90	0.66	0

muy lógico tratandose de un gen que tiene una función de activación transcripcional con los demás, según GO.

1.6.2 *Red Bayesiana minimal*

Para realizar analisis causales sobre las variables de estudio, basados en los datos observados, es importante resumir los resultados del aprendizaje en una red Bayesiana. En esta sección presentamos un método para obtener la *red Bayesiana minimal* resultado del aprendizaje basado en los datos observados. Para este efecto, construimos a partir de la ponderación, una red bayesiana paso a paso:

1. Encontramos la arista con el mayor frecuencia de toda la red, es decir, la arista que tuvo más ocurrencias en el conjunto de redes óptimas, obtenido con el procedimientno de la sección anterior. El resultado de este paso será entonces los nodos origen y destino correspondientes a la arista: i, j que forma la primera arista de la red a construir.

2. Ahora se filtra el conjunto de redes óptimas dejando unicamente las redes donde aparece la arista i, j. Se ajustan las frecuencias de todas las aristas en el conjunto filtrado de redes óptimas.

3. Se repiten los pasos 1 y 2 hasta que todas las aristas en el conjunto filtrado tengan frecuencia 1.

La red Bayesiana resultante es una red que pertenece al conjunto inicial de redes óptimas y le llamamos *minimal*.

Ejemplo de la red Bayesiana minimal

Para ilustrar este procedimiento, retomaremos de nuevo el ejemplo usado en la sección 1.4. Adicionalmente, fueron seleccionados aleatoriamente dos conjuntos de genes de control, cada uno con 9 genes. Los resultados de esta experimentación pueden verse en la figura 1.4. El orden secuencial se detecta fácilmente usando la altura de la posición de los nodos. Se verifica que cada una de las redes es completa (las aristas salientes de un nodo tienen el mismo color). El número sobre una arista indica la itercación en que fue seleccionado. Los nodos fueron etiquetados con el gen. Observamos la secuencia:

malQ→malT→lamB→malG→malP→malM→malE→malF→malK.

Las aristas más frecuentes son malT→malM (la primera, frecuencia 99 en la red ponderada), malQ→malM (la segunda con una frecuencia de 95), malT→malP (la tercera, frecuencia 92). Las etiquetas en la fígura corresponden al orden en que la arista fue seleccionada seguido de la frecuencia de ocurrencia en el conjunto de redes Bayesianas dentro del rango del nivel óptimo.

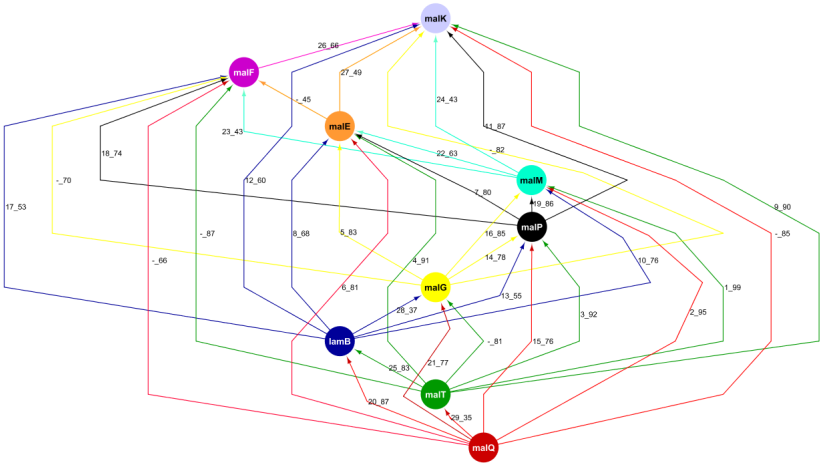
Las 3 redes tienen una estructura identica: siempre hay un nodo padre que tiene aristas con todos los demas nodos (8 nodos hijos), y en la jerarquía le sigue un hijo que es padre de los demás (7 nodos hijos), y asi sucesivamente, observándose, que en esta jerarquía no se podrian tener más aristas (8+7+...+1=36 aristas) ya que cada arista adicional convertiría la red en un grafo ciclico. En conclusión, la optimización no puede dar con una estructura de red con más aristas, lo que permite caracterizar este tipo de red como una "red Bayesiana completa". Solo el orden secuencial de los nodos (del nodo con el máximo número de aristas salientes al nodo sin aristas salientes) determina la estructura de una red bayesiana completa. Como se verá más adelante en el capítulo del caso de estudio, esta situación se evidencia también en redes con 4 genes, 5 genes y 11 genes (ver en la tabla **??**).

1.6.3 *Red Bayesiana reducida*

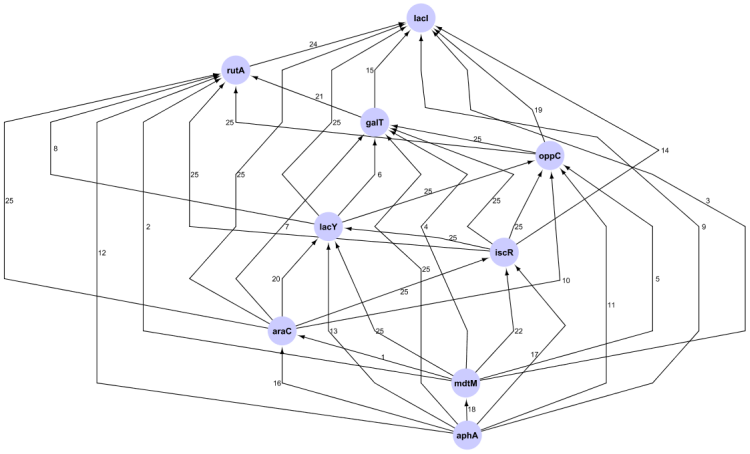
Para facilitar los análisis y encontrar carácteristicas biológicamente interesantes en las redes Bayesianas minimales, aplicamos un procedimiento adicional para dar con una red Bayesiana minimal *reducida*, que consiste de los siguientes pasos:

1. Se parte de una red completamente desconectada, es decir, sin aristas.
2. Con base en la red Bayesiana mínimal, se selecciona la arista con mayor frecuencia de ocurrencia en el conjunto de redes Bayesianas dentro del rango del nivel óptimo.

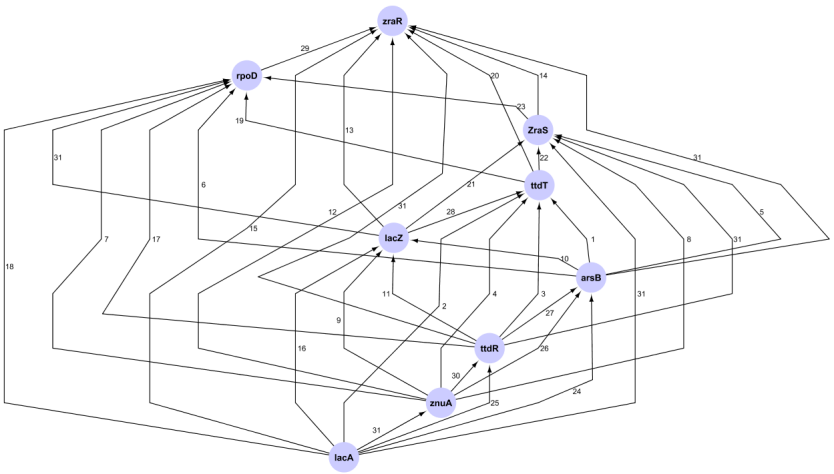
Figura 1.4: Comparación entre 3 redes Bayesianas minimales con 9 genes.



(a) Red Bayesiana minimal para los genes del cluster de transporte de maltose.

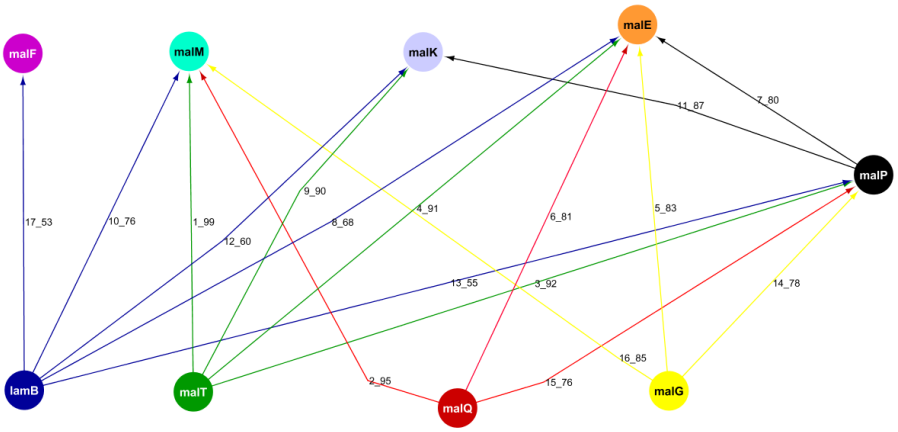


(b) Red Bayesiana minimal para los genes del cluster de genes aleatorios del control 1.



(c) Red Bayesiana minimal para los genes del cluster de genes aleatorios del control 2.

Figura 1.5: Red Bayesiana minimal reducida de los genes del cluster de transporte de maltose en E.coli. Las etiquetas en la fígura corresponden al orden en que la arista fue seleccionada seguido de la frecuencia de ocurrencia en el conjunto de redes Bayesianas dentro del rango del nivel óptimo. Los nodos fueron etiquetados con el gen.



3. Se define un nuevo sub-conjunto con las redes Bayesianas dentro del rango del nivel óptimo pero además donde la arista seleccionada en el paso anterior esta presente en la red.
4. Se repite el paso (2) pero esta vez la frecuencias se cálcula con base en el sub-conjunto definido en el paso anterior y cuidando que la arista seleccionada no haya sido ya procesada.
5. Se repite el paso (3) pero esta vez el sub-conjunto se calcula con las redes Bayesianas del último sub-conjunto y donde la arista seleccionada en el paso anterior este presente en la red.
6. El procedimiento terminal cuando al conectar la última arista transforma la red Bayesiana resultante en un red conectada, es decir, que no exista ningun nodo desconectado de la red

Ejemplo de la red Bayesiana reducida

Pasando a la práctica nuevamente con el ejemplo de datos de expresión génica de E.coli, pudimos ver en la sección anterior una red con muchas aristas, dificil aun de análizar biológicamente. Ahora aplicando el procedimiento de reducción podemos ver algunos ejemplos de resultados más prestos para un análisis causal entre genes según su función génica y anotaciones GO, ver la figura 1.5. Las etiquetas en la fígura corresponden al orden en que la arista fue seleccionada seguido de la frecuencia de ocurrencia en el conjunto de redes Bayesianas dentro del rango del nivel óptimo

La red reducida tiene 4 padres (malQ, malT, lamP, malG) y 4 hojas (malM, malE, malF, malK); sólo el nodo malP queda intermedio. Esto confirma lo dicho en la seccion 1.5 (de sobre-entrenamiento) sobre el la tendencia de agregrar aristas nuevas entre padre para mejorar el score.

La cantidad máxima de 36 aristas en la red minimal fue reducido a sólo 16 aristas en la red reducida.

1.7 COMENTARIOS CONCLUYENTES

- Los tiempos computacionales de las simulaciones MCMC se veran afectados por el número de variables, ya que al evaluarse la posterior para el nodo hoja de la red Bayesiana minimal (que es una de las redes óptimas) la complejidad será exponencial como se discutío en la sección 1.3.3.
- Como una mejora de la simulación MCMC se propusó el método de simulación iterativa, después de analizar que los caminos de exploración del espacio de búsqueda en un punto de la simulación MCMC, pueden quedar restringidos por el rumbo tomado y por la consigna de no generación de bubles en las redes del conjunto solución, y encontramos un equilibrio (trade-off) entre el número de simulaciones suficientes para agregar nuevos modelos, pero sin impactar demasiado la complejidad del problema, estableciendo como criterio de parada de las iteraciones, el porcentaje mínimo de nuevos m odelos para agregar al conjunto solución.
- Inherente al aprendizaje de redes Bayesianas es la tendencia a sobre-entrenar las redes, en todos los modelos se busca optimizar la función score (likelihood), la cual mejora cuando se agregan nuevas aristas. Entre más aristas se agregan, la red mejora su puntaje (o score), sin embargo no todas nuevas aristas tienen poder explicativo biológico como se mencionó en la sección 1.5. Por eso se recomienda la restricción de aplicar el aprendizaje de redes Bayesianas a clusters y la necesidad de emplear mecanismos capaces de mantener las relaciones biológicamente relevantes y eliminar aristas no importantes, lo que conduce a la introducción de redes minimales y reducidas (Secciones 1.6.2 y 1.6.3, respectivamente).
- La red ponderadase puede utilizar para análizar la frecuencia de ocurrencia de las aristas del conjunto de redes óptima. Dado que se consideran sólo las redes del nivel óptimo y se descartan las redes repetidas, la red ponderada aporta información para un análisis cuantitativo. La red ponderada no es Bayesiana, por eso, un análisis causal a partir de ella es limitada. Sin embargo resultó muy util en la construcción de la red Bayesiana minimal y reducida. El uso de una red ponderada fue una sugerencia para análisis comparativos entre redes bayesianas óptimas y redes regulatorias de *E.coli* durante la estancia doctoral en UNAM.
- El método para obtener la red Bayesiana minimal entrega una red Bayesiana que consolida el conjunto de redes óptimas. Es

importante resaltar que la red mínima es basada en la frecuencia de las aristas, de manera que se seleccionan las aristas más probables sobre aquellas que aparecen solo pocas veces.

- La red minimal aunque optima sigue pareciendo sobreentrenada. La red Bayesiana reducida logra mitigar el sobreentrenamiente eliminando las aristas con menos frecuencia de ocurrencia en los conjuntos de redes óptimas, pero sin dejar desconexa la red Bayesina. Es una simplificación de la red minimal, que contiene las aristas estadísticamente más relevantes. Por esta razón es un modelo más apropiado para el análisis bioinformático.