Aprendizaje por transferencia de redes bayesianas

por

Roger Luis Velázquez

Tesis sometida como requisito parcial para obtener el grado de

MAESTRO EN CIENCIAS EN EL ÁREA DE CIENCIAS COMPUTACIONALES

en el

Instituto Nacional de Astrofísica, Óptica y Electrónica
2009
Tonantzintla, Puebla

Supervisada por:

Dr. Luis Enrique Sucar Succar, INAOE
Dr. Eduardo Morales Manzanares, INAOE

Resumen

En muchos dominios, es común tener datos de problemas similares (entendiéndose así porque las distribuciones de los datos son parecidos entre sí pero no iguales). Por ejemplo, en la industria se tienen muchos productos que se obtienen por el mismo proceso industrial pero con diferentes condiciones; o en el diagnóstico industrial donde se encuentran equipos con especificaciones similares. En estos casos, es común tener una gran cantidad de datos para algunos escenarios pero muy pocos para otros, por ejemplo, para productos raros de poca producción. Cuando se cuenta con muchos datos se pueden inducir modelos a partir de éstos que pueden ser utilizados en tareas de diagnóstico y clasificación. Sin embargo, como la exactitud del modelo inducido está en función de los datos disponibles, al tener relativamente pocos datos se obtienen modelos muy pobres. Con el objetivo de mejorar la exactitud del aprendizaje de modelos para dominios con pocos datos, una posibilidad es utilizar datos y conocimiento de dominios similares. Utilizar conocimiento de dominios similares ya ha sido abordado en la literatura presentándose técnicas conocidas como APRENDIZAJE DE MÚLTIPLES TAREAS*, cuyo objetivo es mejorar múltiples modelos simultáneamente, o en otros casos mejorar un único modelo utilizando técnicas de* APRENDIZAJE POR TRANSFERENCIA*. Las redes bayesianas no han sido muy utilizadas con las técnicas mencionadas anteriormente. En esta tesis, se propone utilizar aprendizaje por transferencia en métodos de obtención de la estructura y parámetros de redes bayesianas a partir de datos. Para el aprendizaje estructural, se usan pruebas de independencia condicional, combinando medidas desde el dominio objetivo con las obtenidas de uno o más de los dominios auxiliares, transfiriendo información desde los dominios más relacionados con el objetivo de mejorar la precisión de las partes menos confiables de la red. Para el aprendizaje paramétrico, se compararon técnicas de agregación de probabilidades que combinan las probabilidades estimadas de los datos con los datos auxiliares. Mediante estas técnicas se trata de abordar dos problemas relacionados: la falta de información en los dominios con pocos*

datos utilizando problemas relacionados y una manera de transferir conocimiento desde problemas relacionados conservando aquellas características propias del modelo objetivo. Para validar la propuesta, son usadas tres redes bayesianas comúnmente utilizadas en la literatura, generándose variantes de cada modelo cambiando la estructura y los parámetros. Se aprende la estructura de una de las variantes con un pequeño conjunto de casos combinándose con la información de otras variantes. Los resultados experimentales muestran una mejora significativa en la exactitud de la recuperación en la estructura y parámetros de la red cuando se transfiere conocimiento desde problemas similares.

Abstract

In several domains, it is common to have data from different, but closely related problems (this means that the distributions of the data are similars but no equals). For instance, in manufacturing many products follow the same industrial process but with different conditions; or in industrial diagnosis, where there is equipment with similar specifications. In these cases, it is common to have plenty of data for some scenarios but very little for other, for example, for rare products of little production. When there are a lot of data they can induce models from which can be used in diagnosis and classification tasks. However, as the exactitude of the induced model is based on the data available, having relatively little data are obtained very poor models. In order to improve the accuracy of models for learning domains with little data, one possibility is to use data and knowledge of similar domains. Using knowledge of similar domains has already been addressed in previous works introducing techniques known as MULTITASK LEARNING, *whose objective is to improve multiple models simultaneously, or otherwise improve a single model using techniques known as* TRANSFER LEARNING. *The bayesians networks have not been used with the mentioned techniques previously. In this thesis, we propose a transfer learning method to learn Bayesian networks that considers both, structure and parameter learning. For structure learning, we use conditional independence tests, by combining measures from the target domain with those obtained from one or more auxiliary domains, transferring information from the most related domains with the aim of improving the accuracy of the less reliable parts of the network. For parameter learning, it's compared three techniques for probability aggregation that combine probabilities estimated from the target domain with the auxiliary data. Through these techniques is to address two related problems: the lack of information in the domains with little data using domains related and a way of transferring knowledge from those domains related retaining characteristics of the target model. To validate the approach, are used three standard Bayesian networks commonly used in literature, and generated variants of each model by chan-*

ging the structure as well as the parameters. Then learned on one of the variants with a small data set and combined it with information from the other variants. The experimental results show a significant improvement in the accuracy of the recovery in the structure and parameters when knowledge is transferred from similar problems.

Agradecimientos

A mis asesores, el Dr. Luis Enrique Sucar Succar y el Dr. Eduardo Morales Manzanares, por su orientación, por su apoyo al permitirme explorar diferentes escenarios de investigación, por su paciencia y tiempo otorgados.

A mis revisores de tesis, Dr. Manuel Montes y Gomez, Dr. Leopoldo Altamirano Robles y en especial al Dr. Jesús Gonzáles Bernal , por tomar tiempo y dedicación en sus observaciones.

A los profesores del INAOE, que supieron mostrar un nuevo mundo de conocimiento que estaba oculto ante mis ojos.

Al Consejo Nacional de Ciencia y Tecnología (CONACYT) y el Instituto Nacional de Astrofísica, Óptica y Electrónica (INAOE), por el soporte económico brindado durante mis estudios, ya que sin su ayuda no habría sido posible cursar mis estudios.

A mis padres, sin ellos no sería quien soy.

A toda mi familia, por sacrificar mucho tiempo que debí compartir con ellos.

A Berenice, por haber aguantado conmigo tantos momentos difíciles en el transcurso de nuestros estudios haciendo soportable todo el trayecto.

...Gracias

Dedicatoria

A mis padres

A mis hermanos

A mi familia

Berenice. Para ti, con quien he pasado de los mejores momentos de mi vida

Contenido

1. Introducción **1**

1.1. Motivación y definición del problema 1

1.2. Objetivos de la tesis 4

1.3. Retos 5

1.4. Desarrollo y resultados 5

1.5. Contribuciones 6

1.6. Organización de la tesis 7

2. Redes bayesianas y aprendizaje **9**

2.1. Introducción 9

2.2. Aprendizaje en redes bayesianas 12

2.3. Conclusiones 29

3. Aprendizaje por transferencia **31**

3.1. Introducción 31

3.2. Aprendizaje por transferencia 32

3.3. Aprendizaje por transferencia en redes bayesianas 36

3.4. Conclusiones 38

4. Aprendizaje estructural utilizando aprendizaje por transferencia 41

4.1. Introducción 41

4.2. Justificación del método propuesto 42

4.3. Descripción del método propuesto 43

4.4. Conclusiones 48

5. Aprendizaje paramétrico utilizando aprendizaje por transferencia 51

5.1. Introducción 51

5.2. Emparejamiento de las estructuras 52

5.3. Agregación de probabilidades 54

5.4. Métodos propuestos 55

5.5. Conclusiones 59

6. Experimentos y Resultados 61

6.1. Metodología de experimentación 61

6.2. Experimentos 64

6.3. Resultados 68

6.4. Discusión y análisis. 78

7. Conclusiones y Trabajo Futuro 81

7.1. Resumen 81

7.2. Conclusiones 82

7.3. Trabajo futuro 83

Referencias 87

A. Apéndice 93

A.1. Red Boblo 93

A.2. Red Insurance 102

Lista de Figuras

1.1. Un ejemplo sencillo de una red bayesiana. 2

2.1. Una red bayesiana es la factorización de la distribución de probabilidad conjunta. 11

2.2. Redes bayesianas con variables instanciadas. 17

2.3. Ejemplo de la estimación de dependencia/independencia para un par de variables en el algoritmo PC. 22

3.1. Aprendizaje de tres tareas sencillas mediante redes neuronales. 33

3.2. Aprendizaje de múltiples tareas mediante redes neuronales. 34

5.1. Representacion de casos para el emparejamiento de las estructuras en la combinación de probabilidades condicionales. 53

6.1. Estructura original de la red Alarm. 69

6.2. Comportamiento del método propuesto (PC-TL) ante variaciones en la cantidad de instancias de entrenamiento para el dominio objetivo (red Alarm). 70

6.3. Comportamiento del método propuesto (PC-TL) ante variaciones en la cantidad de instancias de entrenamiento para el dominio auxiliar (red Alarm). 71

6.4. Comportamiento de los métodos de aprendizaje paramétrico propuestos ante variaciones en la cantidad de instancias de entrenamiento en el dominio objetivo (para la red red Alarm). 73

6.5. Comportamiento de los métodos de aprendizaje paramétrico propuestos ante variaciones en la cantidad de instancias de entrenamiento en el dominio auxiliar (para la red red Alarm). 74

6.6. Comportamiento del método propuesto (PC-TL) mientras se varía la similitud de los dominios auxiliares con el dominio objetivo, mostrándose variaciones en la cantidad de instancias de entrenamiento para el dominio objetivo (red Alarm). . . 75

6.7. Comportamiento del método propuesto (PC-TL) mientras se varía la similitud de los dominios auxiliares con el dominio objetivo, mostrándose variaciones en la cantidad de instancias de entrenamiento para los dominios auxiliares (red Alarm). 76

6.8. Comportamiento del método propuesto (PC-TL) cuando las redes auxiliares solo varían en la cantidad de enlaces eliminados de la estructura de la red objetivo original (red Alarm). 77

6.9. Ejemplo de algunas estructuras aprendidas por el algoritmo PC y el algoritmo propuesto (PC-TL). 80

A.1. Estructura original de la red Boblo. 94

A.2. Comportamiento del método propuesto (PC-TL) ante variaciones en la cantidad de instancias de entrenamiento para el dominio objetivo (red Boblo). 95

A.3. Comportamiento del método propuesto (PC-TL) ante variaciones en la cantidad de instancias de entrenamiento para el dominio auxiliar (red Alarm). 96

A.4. Comportamiento de los métodos de aprendizaje paramétrico propuestos ante variaciones en la cantidad de instancias de entrenamiento en el dominio objetivo (para la red red Boblo). 97

A.5. Comportamiento de los métodos de aprendizaje paramétrico propuestos ante variaciones en la cantidad de instancias de entrenamiento en el dominio auxiliar (para la red red Boblo). 98

A.6. Comportamiento del método propuesto (PC-TL) mientras se varía la similitud de los dominios auxiliares con el dominio objetivo, mostrándose variaciones en la cantidad de instancias de entrenamiento para el dominio objetivo (red Boblo). . . 99

A.7. Comportamiento del método propuesto (PC-TL) mientras se varía la similitud de los dominios auxiliares con el dominio objetivo, mostrándose variaciones en la cantidad de instancias de entrenamiento para los dominios auxiliares (red Boblo). 100

A.8. Comportamiento del método propuesto (PC-TL) cuando las redes auxiliares solo varían en la cantidad de enlaces eliminados de la estructura de la red objetivo original (red Boblo). 101

A.9. Estructura original de la red Insurance. 103

A.10. Comportamiento del método propuesto (PC-TL) ante variaciones en la cantidad de instancias de entrenamiento para el dominio objetivo (red Insurance). 104

A.11. Comportamiento del método propuesto (PC-TL) ante variaciones en la cantidad de instancias de entrenamiento para el dominio auxiliar (red Insurance). 104

A.12.Comportamiento de los métodos de aprendizaje paramétrico propuestos ante variaciones en la cantidad de instancias de entrenamiento en el dominio objetivo (para la red red Insurance). 106

A.13.Comportamiento de los métodos de aprendizaje paramétrico propuestos ante variaciones en la cantidad de instancias de entrenamiento en el dominio auxiliar (para la red red Insurance). 107

A.14.Comportamiento del método propuesto (PC-TL) mientras se varía la similitud de los dominios auxiliares con el dominio objetivo, mostrándose variaciones en la cantidad de instancias de entrenamiento para el dominio objetivo (red Insurance). 108

A.15.Comportamiento del método propuesto (PC-TL) mientras se varía la similitud de los dominios auxiliares con el dominio objetivo, mostrándose variaciones en la cantidad de instancias de entrenamiento para los dominios auxiliares (red Insurance). 109

A.16.Comportamiento del método propuesto (PC-TL) cuando las redes auxiliares solo varían en la cantidad de enlaces eliminados de la estructura de la red objetivo original (red Insurance). 110

Lista de Tablas

6.1. Descripción de las redes usadas en los experimentos. 62
6.2. Características de las redes auxiliares utilizadas en los experimentos tipo I para la red Alarm. 65
6.3. Características del conjunto 1 de redes utilizadas en los experimentos tipo II para la red Alarm. 67
6.4. Características del conjunto 2 de redes utilizadas en los experimentos tipo II para la red Alarm. 67

A.1. Características de las redes auxiliares utilizadas en los experimentos tipo I para la red Boblo. 94
A.2. Características del conjunto 1 de redes utilizadas en los experimentos tipo II para la red Boblo. 100
A.3. Características del conjunto 2 de redes utilizadas en los experimentos tipo II para la red Boblo. 101
A.4. Características de las redes auxiliares utilizadas en los experimentos tipo I para la red Insurance. 102
A.5. Características del conjunto 1 de redes utilizadas en los experimentos tipo II para la red Insurance. 109
A.6. Características del conjunto 2 de redes utilizadas en los experimentos tipo II para la red Insurance. 110

Lista de algoritmos

2.1. Pseudocódigo del algoritmo PC 20
2.2. Pseudocódigo para el algoritmo K2 28
4.1. Pseudocódigo del algoritmo propuesto (PC-TL) 49

Capítulo 1

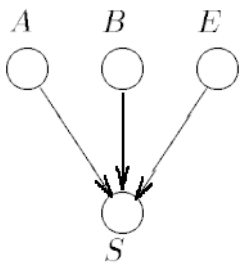
Introducción

1.1. Motivación y definición del problema

Las redes bayesianas proveen una forma compacta e intuitiva de describir la estructura de dependencias probabilísticas entre variables en un dominio usando un grafo acíclico dirigido. Esta descripción intuitiva de la estructura de las dependencias en las redes bayesianas motiva su utilización en sistemas expertos donde el conocimiento experto puede ser construido a mano a través de grafos de dependencias. Adquirir el conocimiento experto de los humanos es una tarea difícil y costosa ya que se requiere un proceso de refinación del conocimiento antes de que la información aportada comience a ser útil, por lo que se han llevado a cabo investigaciones en aprender las redes bayesianas a partir de datos [Bun96].

Este tipo de aprendizaje ofrece la posibilidad de inducir la estructura gráfica de la red a partir de los datos observados y de obtener los parámetros asociados a cada variable basándose también en dichos casos. A estas dos fases se les denomina aprendizaje estructural y aprendizaje paramétrico respectivamente. A continuación se resume cada una de estas dos fases:

- Aprendizaje estructural: obtiene la estructura de la red bayesianas a partir de



A	B	E	$\Pr(S\|A,B,E)$
1	1	1	0.95
1	1	0	0.95
1	0	1	0.20
1	0	0	0.05
0	1	1	0.00
0	1	0	0.00
0	0	1	0.00
0	0	0	0.00

Figura 1.1: Un ejemplo sencillo de una red bayesiana. A la izquierda se muestra el grafo y las variables, a la derecha la distribución de probabilidad asociada para la variable S (mostrando los valores de probabilidad para $S = 1$).

bases de datos, es decir, las relaciones de dependencia e independencia entre las variables involucradas.

- Aprendizaje paramétrico: dada una estructura y las bases de datos, obtiene las probabilidades *a priori* y condicionales requeridas.

La figura 1.1 muestra un ejemplo de la distribución de probabilidad de la variable S (derecha) dada la estructura gráfica presente (izquierda).

En la literatura se encuentran muchas propuestas en el área de aprendizaje estructural de redes bayesianas a partir de datos, dentro de los cuales se pueden distinguir dos categorías de acuerdo a la forma en que trabajan. Una categoría de algoritmos usa métodos de búsqueda basados en una heurística para construir un modelo que entonces es evaluado usando una medida de ajuste a los datos. La búsqueda se realiza en el espacio de posibles redes bayesianas intentando encontrar la red óptima. Diferentes criterios de medida han sido aplicados a estos algoritmos, tales como métodos de Medida Bayesiana [CH92, HGC95], métodos basados en entropía [HC90], y métodos de Descripción de Longitud Mínima [Suz96]. La otra categoría de algoritmos construye redes bayesianas por medio de análisis de dependencias entre sus nodos. Las relaciones de independencia son medidas por algún tipo de prueba de independencia

condicional (IC). Ambas categorías de algoritmos tienen sus ventajas y desventajas. Generalmente, la primera categoría de algoritmos tiene menor complejidad en tiempo en el peor caso (cuando el grafo base está densamente conectado), pero podría no encontrar la mejor solución debido a su naturaleza heurística. La segunda categoría de algoritmos es asintóticamente correcta cuando la distribución de probabilidad de los datos satisface ciertas consideraciones [Pea88][1], pero las relaciones de independencia condicional de orden alto[2] podrían no ser confiables a menos que se disponga de una gran cantidad de datos [CH92].

Para muchas aplicaciones de aprendizaje computacional, se asume que se tiene una gran cantidad de datos disponibles para los cuales se puede inducir un modelo a partir de los datos. En algunos dominios, sin embargo, es difícil obtener suficientes datos o no se encuentran disponibles. Por ejemplo, en la industria algunos productos son raramente producidos o en medicina algunas enfermedades son muy poco comunes. Cuando expertos se enfrentan con un problema en un nuevo dominio, ellos usan su experiencia en dominios relacionados para resolver el problema. Recientemente se ha incrementado el interés de la comunidad de aprendizaje automático en una técnica conocida como aprendizaje por transferencia, particularmente cuando no se cuenta con suficientes datos [Bax97, Thr96, Car97]. El aprendizaje por transferencia desde un dominio familiar hacia otro dominio nuevo o poco conocido pero muy relacionado es un aspecto fundamental dentro del aprendizaje humano. Por ejemplo, los atletas hacen uso de aprendizaje por transferencia cuando ellos practican sus habilidades fundamentales para mejorar su entrenamiento en actividades más complicadas, trabajadores de cualquier índole se adaptan rápidamente a nuevas circunstancias cuando su entorno de trabajo cambia o bien cuando se cambian a otro tipo de trabajo parecido al que realizaban. Aunque la noción es intuitiva una simple propuesta de tratar ambos dominios como idénticos y combinar los datos de entrenamiento usualmente no funciona bien. Este es el principio básico detrás del aprendizaje por transferencia: un sistema

[1] Estas condiciones son la condición de Markov y la condición de fidelidad, y se mencionará brevemente en la página 19 al comentar el algoritmo más representativo de esta categoría, el algoritmo PC.

[2] Se denomina así a las relaciones de independencia condicional en donde el conjunto condicionante tiene un número elevado de variables. Si por el contrario este conjunto contiene un número reducido de variables se le denomina relaciones de independencia de orden bajo.

entrenado en una tarea debería de mejorar el desempeño más rápidamente en tareas relacionadas que los sistemas que no cuentan con dicho conocimiento de tareas similares.

El aprendizaje por transferencia no ha sido muy explorado en el área de aprendizaje estructural de redes bayesianas. Hasta ahora las investigaciones se han enfocado en aprender grafos de dependencias para un problema aislado donde una hipótesis para un problema (también llamado tarea) es inducido de un conjunto de ejemplos de entrenamiento sin considerar aprendizajes previos o la retención de conocimiento de tareas para futuros aprendizajes; sin embargo, en muchas situaciones existen datos disponibles para múltiples problemas relacionados. En este trabajo se propone la técnica conocida como aprendizaje por transferencia aplicado al aprendizaje estructural y paramétrico de redes bayesianas. Mediante esta forma de aprendizaje se incrementa la exactitud en la recuperación del modelo original cuando solo se dispone de una pequeña cantidad de casos de entrenamiento y se dispone de suficientes datos para problemas relacionados, comparándolo con los métodos de aprendizaje que utilizan únicamente datos de entrenamiento.

1.2. Objetivos de la tesis

Desarrollar un método de aprendizaje de redes bayesianas a partir de varias fuentes de datos de dominios similares, enfocado especialmente en la mejora de la precisión de aquellos construidos con relativamente pocos datos.

Para esto nos planteamos los siguientes objetivos específicos:

- Seleccionar e implementar un método de aprendizaje estructural de redes bayesianas adecuado para el problema.
- Proponer métodos de agregación de información estructural desde las redes similares.
- Proponer métodos de agregación de información paramétrica desde las redes similares.

1.3. Retos

La falta de información disponible para el aprendizaje ha sido un área de estudio desde hace décadas, el aprendizaje por transferencia [Car97, Thr96, Bax97] ha sido utilizado en diversas áreas dentro de la comunidad de Aprendizaje Computacional. En su utilización no se tienen medidas prescritas acerca de la similitud de las tareas y es difícil obtener una única respuesta de ¿cómo y cuándo transferir información entre tareas? y ¿cuánta información transferir?

El reto para los sistemas de aprendizaje por transferencia es seleccionar qué conocimiento debería ser transferido y cómo hacerlo. Como no hay garantía *a priori* de que las tareas auxiliares y objetivo son suficientemente similares, un algoritmo debe usar los datos disponibles para guiar el aprendizaje por transferencia y al mismo tiempo observar que la transferencia no degrade el desempeño del aprendizaje. Para superar estos inconvenientes el método propuesto de aprendizaje estructural busca similitudes en las dependencias/independencias condicionales presentes entre los dominios de interés y auxiliares utilizándolos en la construcción del modelo.

1.4. Desarrollo y resultados

El aprendizaje en redes bayesianas involucra dos aspectos principales: aprendizaje estructural y aprendizaje paramétrico. En esta tesis se desarrollarán métodos de aprendizaje estructural y paramétrico que utilizan aprendizaje por transferencia. El algoritmo de aprendizaje estructural combina las medidas de dependencia obtenidas de los datos en el dominio objetivo, con las obtenidas de los datos en los dominios auxiliares. Se propone una función de combinación que toma en cuenta la consistencia entre estas medidas para detectar el grado de independencia entre las variables y con esto construir la red. Los algoritmos de aprendizaje paramétrico emplean un método de agregación, combinando los parámetros estimados desde el dominio objetivo, con aquellos estimados de los datos de los dominios auxiliares. Basado en técnicas de combinación lineal (la cual se detalla en la sección 5.3), se proponen dos variantes:

1. Combinación Lineal basada en Distancia (DBLP), la cual toma en cuenta una medida de distancia entre los parámetros auxiliares y los parámetros objetivo, y

2. Combinación Lineal Local (LoLP), la cual únicamente incluye los parámetros de los dominios auxiliares cuando se encuentran cercanos en distancia al dominio objetivo, pesada por la cantidad de datos en el dominio de interés. En los experimentos comparamos ambos métodos, y utilizamos como base la combinación lineal mencionado anteriormente.

Evaluamos experimentalmente la estructura y parámetros de las redes recuperadas por los métodos propuestos y comparamos los resultados contra utilizar únicamente datos del dominio de interés. Para esto consideramos 3 redes bayesianas comúnmente usadas como base de pruebas en los algoritmos de aprendizaje (Alarm, Boblo e Insurance), y generamos redes auxiliares cambiando la estructura y parámetros a partir de los modelos originales. Generamos datos de las redes originales y sus variantes y entonces probamos los métodos combinando los datos. Evaluamos la estructura, mostrándose una mejora notable en la recuperación de la estructura de la red cuando la cantidad de datos de la red objetivo es relativamente pequeña, así mismo, encontrándose mejores resultados al incrementar la cantidad de datos para las redes auxiliares. En cuanto al aprendizaje paramétrico, los métodos propuestos obtienen igual o menor error que utilizar sólo los datos del dominio de interés, mejorando al método base de combinación lineal.

1.5. Contribuciones

Las principales aportaciones de este trabajo se tienen desde dos aspectos: por un lado, se presenta un método novedoso del uso de aprendizaje por transferencia aplicado al aprendizaje estructural de redes bayesianas basado en detección de independencias (capítulo 4); y por otro lado, se presentan dos variantes de métodos de aprendizaje paramétrico que hacen uso del aprendizaje por transferencia (capítulo 5). En ambos casos se presenta una mejora de la exactitud en la recuperación de la estructura y los

parámetros de la red al utilizar información de dominios similares (ver capítulo 6) creándose un precedente del uso de transferencia de aprendizaje en el área de aprendizaje de redes bayesianas.

1.6. Organización de la tesis

En el capítulo 2 se explica de manera general los fundamentos del aprendizaje en redes bayesianas. Dentro de este capítulo se pueden distinguir básicamente dos tipos de métodos para recuperar la topología de una red, los métodos que utilizan criterios de independencia y los que utilizan alguna métrica y técnica de búsqueda. Los primeros hacen hincapié en las relaciones de independencia que son capaces de representar en el modelo gráfico, mientras que los segundos tratan de encontrar el modelo que mejor se aproxime a los datos según algún criterio de bondad de ajuste. Para la recuperación de los parámetros, se revisa un método sencillo basado en el conteo de frecuencias en los datos.

Posteriormente en el capítulo 3 se presentan algunas técnicas de aprendizaje por transferencia de la literatura actual aplicado a diversos campos. También se discuten algunos métodos de aprendizaje por transferencia más relacionados con nuestra problemática.

En el capítulo 4 se expone el método de aprendizaje estructural propuesto, el cual busca las relaciones de independencia presentes en los datos de la tarea objetivo utilizando algún grado de aprendizaje por transferencia de acuerdo al grado de similitud con las tareas auxiliares.

En el capítulo 5 se proponen dos técnicas de agregación de probabilidades lineales que hacen uso en algún grado de la información proveniente de las tablas de probabilidad de las tareas auxiliares con el objetivo de mejorar las tablas de probabilidad condicional de la tarea objetivo.

En el capítulo 6 se presentan los resultados obtenidos de aplicar los métodos de aprendizaje estructural y paramétricos propuestos sobre las bases de datos de prueba. Se

analizan las variaciones de los resultados, empleando diversos parámetros de acuerdo a la cantidad de casos utilizados en las tareas de interés y auxiliares así como el comportamiento ante variaciones en similitud de dichos dominios.

Finalmente, en el capítulo 7 se exponen las conclusiones del trabajo y el trabajo futuro.

Capítulo 2

Redes bayesianas y aprendizaje

En esta sección se definen formalmente las redes bayesianas y se mencionan algunos métodos relacionados con su aprendizaje. En el aprendizaje paramétrico se describe un método simple mediante un conteo de ocurrencias de las observaciones. Por otra parte se describen dos tipos de algoritmos para la obtención de la estructura de una red bayesiana a partir de datos: un tipo de algoritmos trata de detectar las independencias que existen en las variables y construir con ello la estructura, mientras que la otra categoría de algoritmos busca encontrar el mejor modelo de red que se adapte a los datos mediante una función de calidad.

2.1. Introducción

Las redes bayesianas describen la distribución de probabilidad concerniente a un conjunto de variables especificando suposiciones de independencia condicional junto con probabilidades condicionales. Así, las redes permiten especificar relaciones de independencia entre conjuntos de variables, lo cual nos permite describirlas a partir de la probabilidad condicionada de cada nodo, en vez de dar la distribución de probabilidad conjunta, que requeriría un número de parámetros exponencial en el número de nodos.

Una red bayesiana es un grafo acíclico dirigido que describe la distribución de probabili-

dad conjunta que gobierna un conjunto de variables aleatorias. Sea $U = \{X_1, X_2, ..., X_n\}$ un conjunto de variables aleatorias, formalmente, una red bayesiana para U es un par $B =< G, T >$ en el que [Pea88]:

- G es un grafo acíclico dirigido en el que cada nodo representa una de las variables $X_1, X_2, ..., X_n$, y en el que cada arco representa relaciones de dependencia directas entre las variables. La dirección de los arcos indica que la variable "apuntada" por el arco depende de la variable situada en su origen.
- T es un conjunto de parámetros que cuantifica la red. Contiene las probabilidades $P(x_i|pa(x_i))$ para cada posible valor o estado x_i de cada variable X_i y cada posible valor x_i de $pa(x_i)$, donde éste último denota al conjunto de padres de X_i en G.

Así, una red bayesiana B define una distribución de probabilidad conjunta única sobre U dada por

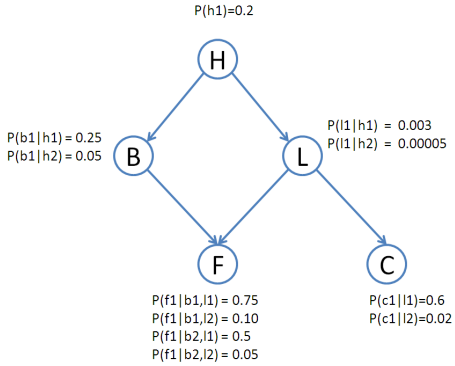
$$P(X_1, X_2, ..., X_n) = \prod_{i=1}^{n} P(X_i|pa(X_i))$$

Ejemplo 2.1 En vez de especificar la distribución de probabilidad conjunta de todas las variables de un dominio de problema, solo es necesario especificar el conjunto de probabilidades condicionales independientes. La figura 2.1 ilustra una distribución de probabilidad conjunta para las variables del dominio (a) y su representación basada en una red bayesiana (b) donde se puede apreciar la parte estructural definida mediante nodos y enlaces representando las variables del dominio del problema y las relaciones entre ellas. La información cuantitativa de una red bayesiana viene definida por la probabilidad inicial de los nodos que no tienen padre, $P(H)$, y por la probabilidad condicional de los nodos con padres, $P(B|H)$, $P(L|H)$, $P(F|B, L)$, y $P(C|L)$.

Para la red bayesiana de la figura 2.1, la factorización de la distribución de probabilidad conjunta es:

H	B	L	F	C	Pr
X	X	X	X	X	#
X	X	X	X	X	#
X	X	X	X	X	#
X	X	X	X	X	#
X	X	X	X	X	#
...	...	...	...	...	...
X	X	X	X	X	#

(a) En la distribucion de probabilidad conjunta se tiene que asignar una probabilidad de ocurrencia para toda combinacion de estados de las variables.



(b) Una red bayesiana es la factorizacion de la distribucion de probabilidad conjunta.

Figura 2.1: Ejemplo de (a) una distribución de probabilidad conjunta y (b) su representacion basada en una red bayesiana ilustrando la estructura y probabilidades asociadas a cada nodo.

$$P(H, B, L, F, C) = P(H) \cdot P(B|H) \cdot P(L|H) \cdot P(F|B, L) \cdot P(C|L)$$

Es importante observar que la topología o estructura de la red no sólo proporciona información sobre las dependencias probabilísticas entre las variables, sino también sobre las independencias condicionales de una variable o conjunto de ellas dada otra u otras variables. Cada variable es independiente de las variables que no son descendientes suyas en el grafo, dado el estado de sus variables padre.

La inclusión de las relaciones de independencia en la propia estructura del grafo hace de las redes bayesianas una buena herramienta para representar conocimiento de forma compacta, reduciendo el número de parámetros necesarios. Además, proporcionan métodos eficientes de razonamiento basados en la propagación de las probabilidades a lo largo de la red de acuerdo con las leyes de la teoría de la probabilidad. El coste de representar la distribución de probabilidad conjunta de n variables es $O(2^n)$, la representacion de redes bayesianas permite una representacion mas compacta. Supo-

niendo que cada nodo de la red tenga a lo máximo k padres, un nodo necesitará 2^k de espacio para representar la influencia de sus padres, por lo tanto el espacio necesario será de $O(n2^k)$. Por ejemplo, con 5 variables y suponiendo 2 padres como máximo tenemos 20 frente a 32, con 50 variables y suponiendo 5 padres tenemos 1600 frente a aproximadamente 10^{15}.

2.2. Aprendizaje en redes bayesianas

Para definir una red bayesiana, incluyendo la estructura de dependencias del modelo y las distribuciones de probabilidad condicional asociadas, es necesario contar con un experto que provea su conocimiento en forma de un grafo de dependencias y de las tablas de probabilidad requeridas; es de suponer que esto constituye un considerable esfuerzo, sobre todo si el dominio del problema da lugar a redes complejas. Dado que no es fácil disponer de un experto, y que en muchos de los campos se dispone de una gran cantidad de datos, se planteó la alternativa del aprendizaje a partir de datos o bien a partir de datos y de conocimiento del experto. Ya se han diseñado numerosas herramientas computacionales para el aprendizaje de redes bayesianas a partir de datos, para la mayoría de los cuales se asume que se dispone de una gran cantidad de datos.

El obtener una red bayesiana a partir de datos es un proceso de aprendizaje, el cual se divide, naturalmente, en dos aspectos:

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

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

A continuación se presentarán brevemente algunos métodos de aprendizaje paramétrico y estructural.

Aprendizaje paramétrico

El aprendizaje paramétrico consiste en encontrar los parámetros asociados a una estructura dada de una red bayesiana. Dichos parámetros consisten en las probabilidades previas de los nodos raíz y las probabilidades condicionales de las demás variables, dados sus padres.

Si se conocen todas las variables, es fácil obtener las probabilidades requeridas. Las probabilidades previas corresponden a las marginales de los nodos raíz, y las condicionales se obtienen de las conjuntas de cada nodo con su(s) padre(s).

En el caso de las redes bayesianas, las fórmulas para las probabilidades previas $p(x_i)$ y condicionales $p(x_i|pa(x_i))$ se estiman como sigue:

PROBABILIDADES PREVIAS
$p(x_i) = \frac{n_i}{t}$

PROBABILIDADES CONDICIONALES
$p(x_i\|pa(x_i)) = \frac{n_{is}}{n_S}$

donde t es el número total de ejemplos de entrenamiento, n_i es el número de ejemplos con $X_i = x_i$, n_S es el número de ejemplos con los estados de las variables de $pa(X_i) = pa(x_i)$, n_{is} es el número de ejemplos con los estados de las variables de $pa(X_i) = pa(x_i)$ y $X_i = x_i$.

Cuando se dispone de pocos datos o se tienen muchos atributos y estados para los atributos se pueden tener probabilidades igual a cero lo que origina problemas en algoritmos que utilicen éstos parámetros. Una manera de resolver esto es *suavisar* los valores evitando el problema de frecuencia cero. El suavizado de Laplace consiste en inicializar todas las probabilidades en forma uniforme, y después incrementarlas con los datos. Utilizando suavizamiento de Laplace, los cálculos para la obtención de los parámetros es como sigue:

PROBABILIDADES PREVIAS
$p(x_i) = \frac{2}{v_i} + \frac{n_i}{t}$

PROBABILIDADES CONDICIONALES
$p(x_i \vert pa(x_i)) = \frac{2}{v_{is}} + \frac{n_{is}}{n_S}$

donde v_i es el numero de estados o valores del atributo $X_i = x_i$, y v_{is} es el producto del numero de estados de los atributos de $X_i = x_i$ y de $pa(X_i) = pa(x_i)$. Normalizándose los resultados para obtener la estimación final de cada probabilidad.

Un enfoque más adecuado, pero un poco más complejo, es utilizar una distribución de probabilidad para éstas. Normalmente se utiliza, para el caso de variables binarias, la distribución Beta y para variables multivaluadas, su generalización que es la distribución Dirichlet [Nea03].

Aprendizaje estructural en redes bayesianas

Las técnicas de aprendizaje estructural dependen del tipo de estructura de red que se busque: árboles, poliárboles y redes multiconectadas. Otra alternativa es combinar conocimiento subjetivo del experto con aprendizaje. Para ello se parte de la estructura dada por el experto, la cual se valida y mejora utilizando datos estadísticos. En general, el proceso de aprendizaje consiste en un procedimiento de búsqueda, guiado por los datos disponibles a través del espacio más o menos restringido de modelos, para hallar algún modelo que más se ajuste a los datos. Para este proceso se hace la suposición de que los datos son una presentación de la distribución de probabilidad que sigue la población y que se tiene un conjunto suficiente de muestras de datos.

El formato de representación de las redes bayesianas está restringido al tipo de modelo que cada algoritmo está orientado a aprender. Como modelos gráficos, para las redes bayesianas nos podemos encontrar diversos tipos de grafos. En orden creciente de representación se encuentran los árboles y poliárboles (que incluyen a los primeros como un caso particular). Los poliárboles son grafos en los que no existe más de un camino (no dirigido) que conecta cualquier par de nodos, esto es, son grafos que no

contienen ciclos no dirigidos. Un tipo más general de grafo son los grafos simples. Son grafos dirigidos acíclicos donde cada par de nodos con un hijo común no tienen antecesores comunes ni uno es antecesor del otro. Esto significa que en un grafo simple sólo están permitidos un tipo especial de ciclos no dirigidos: los que contienen al menos dos nodos cabeza-cabeza[1]. Por último nos encontramos con los grafos acíclicos generales con el mayor poder de representación de todos los grafos (incluidos los grafos no dirigidos), aunque resultan menos operativos pues, los métodos de abducción e inferencia resultan más costosos de llevar a cabo sobre este tipo de estructuras. Existen por tanto algoritmos orientados a cada uno de estos modelos.

Si revisamos la literatura, podemos encontrar un gran cantidad de algoritmos relacionados al aprendizaje de redes bayesianas, no obstante, podemos clasificar estos métodos de aprendizaje de acuerdo al tipo de técnica utilizada para recuperar la topología de la red, de esta forma tenemos dos tipos de métodos:

- Métodos basados en detección de independencias
- Métodos basados en algún criterio de evaluación y técnica de búsqueda

Los algoritmos que utilizan un criterio de independencia usan como entrada una lista $\mathbb{G}$ de relaciones de independencia condicional entre variables y su objetivo es encontrar el grafo que trata de representar la mayor parte de esas relaciones de independencia. El elemento central son las aseveraciones de independencia entre variables, obtenidas a partir de una base de datos $\mathbb{D}$ mediante complejas y numerosas pruebas de independencia condicional, lo que constituye su principal inconveniente, ya que las pruebas de independencia condicional con conjuntos condicionantes de gran tamaño pueden resultar poco fiables (a menos que se disponga de un enorme volumen de datos) [CH92] y muy costosas. Sin embargo tienen el atractivo de su solidez teórica. Dependiendo del tipo de modelo que se emplea para representar la lista $\mathbb{G}$, se pueden encontrar algoritmos para poliárboles, grafos simples o grafos generales. Por otra parte, el objetivo del

[1] También llamados *v-estructuras*, son una terna de nodos con enlace dirigidos $X \rightarrow Y \leftarrow Z$ en donde no existe ninguna arista entre X y Z.

segundo tipo de métodos es encontrar un grafo que, teniendo el menor número de arcos posible, represente mejor los datos. La calidad del ajuste de cada red candidata a los datos se establece mediante alguna función de evaluación (también llamada ajuste o métrica). Esta función, en general es una función del tipo $f(\mathbb{G}, \mathbb{D})$, como veremos, tiene muy diferentes expresiones. La función de evaluación permite ordenar los grafos por su valor de calidad o ajuste a los datos. En un algoritmo de este tipo, asociado a la función que mide la calidad de cada red candidata se tiene un proceso de búsqueda habitualmente heurístico (debido al tamaño mas que exponencial[2] del espacio de búsqueda) que explore el espacio de posibles soluciones. Los algoritmos basados en una métrica resultan computacionalmente más eficientes aunque pueden no encontrar la mejor solución debido a su naturaleza heurística. Cada algoritmo de esta clase se caracteriza por el tipo de métrica y de búsqueda especifica que utiliza. Aparte de los dos mencionados, también existen enfoques híbridos, que utilizan de forma conjunta una técnica de búsqueda orientada por una métrica y la detección de independencias [SV93, AdC96].

Aprendizaje estructural utilizando criterios de independencia

Se basa en que la estructura de una red bayesiana codifica un grupo de relaciones de independencia condicional entre los nodos, de acuerdo al concepto de separación-d [Pea88].

Separación-D Sean X, Y y Z tres subconjuntos disjuntos de nodos en un grafo dirigido acíclico D; entonces se dice que Z d-separa X e Y si y solo si a lo largo de todo camino no dirigido entre cualquier nodo de X y cualquier nodo de Y existe un nodo intermedio A tal que

[2] Robinson [Rob77] mostró que el número de posibles estructuras, $r(n)$, para una red bayesiana teniendo n nodos, es dado por la fórmula recurrente siguiente:

$$r(n) = \sum_{i=1}^{n} (-1)^{i+1} \begin{pmatrix} n \\ i \end{pmatrix} 2^{i(n-1)} r(n-i) = n^{2^{\mathcal{O}(n)}}$$

Por lo que para $r(1) = 1$, $r(2) = 3$, $r(3) = 25$, $r(5) = 29281$ y $r(10) \simeq 4{,}2 \times 10^{18}$.

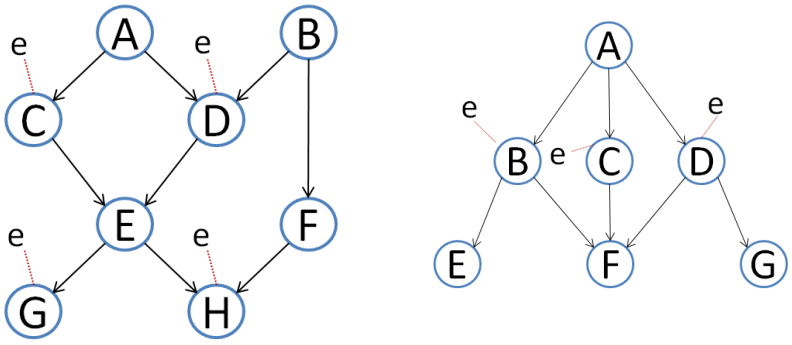


Figura 2.2: Redes bayesianas con variables instanciadas (*evidencia dura*). a) Aunque todos los vecinos de E son instanciados, éste es conectado-d a F, B y A. b) F es separado-d del resto de las variables no instanciadas.

- A es un nodo de aristas convergentes en el camino y ni A ni sus descendientes están en Z (evidencia o variables instanciadas), o bien
- A no es un nodo de aristas convergentes en el camino y A esté en Z.

Si X e Y no son separados-d, las llamaremos *conectados-d*.

Mediante el consepto de separación-d es posible decidir si un par de variables en una red bayesiana cualquiera son independientes dada la evidencia aplicada a la red. La figura 2.2 muestra dos ejemplos de la definición.

Nótese que aunque A y B son conectados-d, cambios en la creencia en A no necesariamente cambiaran la creencia en B. A veces se dice que A y B son estructuralmente independientes si son separados-d.

El aprendizaje de la estructura se realiza por identificación de relaciones de independencia condicional (IC) entre los nodos. Usando alguna prueba estadística (como ji-cuadrado e información mutua), se pueden buscar las relaciones de independencia condicional entre los atributos y usar esa relación como restricción para construir una red bayesiana. Estos algoritmos son referidos como algoritmos basados en restricciones o algoritmos en base a pruebas de independencia.

Como indicamos, este tipo de algoritmos no tratan de obtener una red que cuantitativamente *mejor* represente los datos mediante una función de ajuste a los datos, sino que hacen un estudio cualitativo de las relaciones de dependencia/independencia del modelo subyacente a los datos, de acuerdo al principio de separación-d [Pea88] y a partir de ellas tratan de encontrar una red que represente esas relaciones. Cuando el estudio cualitativo de las relaciones está dirigido por los datos éste se basa en las pruebas de independencia (como el estadístico ji-cuadrado e información mutua) para determinar si las sentencias de independencia están apoyadas por los datos.

Para probar las pruebas de hipótesis de independencia podemos usar la prueba de significancia estadística de máxima verosimilitud [3], G^2.En el caso de pruebas de independencia condicional entre X e Y dado un subconjunto condicionante S, el estadístico es calculado como

$$G^2 = 2 \sum_{x,y,z} N_{xyz} \log \left(\frac{N_{xyz}}{\mathbb{E}_{xyz}} \right)$$

donde $\mathbb{E}_{xyz} = \frac{N_{xz} N_{yz}}{N_z}$, N_{xyz} especifica el número de casos en la base de datos D donde $X = x$, $Y = y$ y $Z = z$, z es una configuración de estados del subconjunto condicionante S.

Se conoce que bajo la asunción de independencia (ejemplo, independencia condicional de X e Y), la máxima verosimilitud de la prueba del estadístico G^2, sigue una distribución ji-cuadrado, X^2, con un apropiado numero de grados de libertad, df. El valor de df es definido como

$$df = (r_X - 1)(r_Y - 1) \prod_{Z \in S} r_Z$$

donde $r_X(r_Y, r_Z)$ es el número de valores de $X(Y, Z)$ respectivamente en D.

Si la probabilidad de la distribución X^2, con sus correspondientes grados de libertad, mayor a G^2 es menor a un nivel de significancia α, entonces la hipótesis de indepen-

[3] La prueba G^2 es una estadística de distribución ji-cuadrado que está siendo utilizada paulatinamente en situaciones donde las pruebas de ji-cuadrada fueron previamente recomendadas.

dencia es rechazada. Es decir, cuando $P_{X^2(df)}(x \geq G^2) < \alpha$ se encuentra dependencia en los datos. En caso de aceptar independencia no significa que los datos apoyen la independencia, sino que no hay evidencia en los datos contra ella.

A continuación vamos a describir brevemente alguno de los algoritmos. Los principales inconvenientes comunes a todos ellos son el elevado costo computacional que supone una prueba con un gran número de variables implicadas además del gran número de pruebas necesarias, potencialmente exponencial.

Los autores Spirtes, Glymour y Scheines en [SGS93] exponen varios algoritmos de este tipo como SGS y PC. Veamos este último.

Algoritmo PC

El algoritmo asume que la Condición de Markov[4] y la Condición de Fidelidad[5] así como las decisiones estadísticas son correctas. Es decir, las relaciones de independencias entre las variables son representadas exactamente por un grafo G mediante el criterio de separación-d [Pea88]. Por $I(X, Y|S)$ denotaremos a las variables X y Y como condicionalmente independientes dado el conjunto de variables S. El algoritmo PC está basado en la existencia de un procedimiento (definido mediante pruebas estadísticas de independencia condicional) que estima la veracidad o no de las relaciones de independencia $I(X, Y|S)$ que encontradas en los datos, se encuentre reflejado en el grafo G.

El algoritmo toma como entrada una base de datos sobre un conjunto de variables V, una prueba de independencia condicional $I(X, Y|S)$, y un nivel de significancia $0 < \alpha < 1$. Implícitamente el algoritmo también toma un ordenamiento, $order(V)$, sobre los nodos, lo cual especifica el orden de prueba dado a las variables para probar las independencias. El pseudocódigo se muestra descrito en el algoritmo 2.1.

[4] Dado un modelo de red bayesiana G, cualquier variable es independiente de sus no descendientes en G dado sus padres.

[5] Una estructura de red bayesiana G y una distribución de probabilidad P generadas por G son fieles una a la otra sí y sólo sí cada relación de independencia condicional válida en P está contenida por la Condición de Markov en G.

Entrada: Una base de datos D sobre un conjunto de variables V, una prueba de independencia condicional: $I(X, Y|S)$, un nivel de significancia: $0 < \alpha < 1$, y un orden $order(V)$ sobre V.

Salida: Un grafo patrón sobre V.

1. Construir un grafo completo no dirigido sobre el conjunto de variables V.
2. Para todos los nodos adyacentes, $X - Y$, intentar separar los nodos probando primero relaciones de independencia de bajo orden entre $X - Y$. Comprobar las relaciones de independencia condicional $I(X, Y|S)$ sí y sólo sí todas las variables en S son adyacentes a X o Y. Si una relación de independencia condicional es descubierta entre $X - Y$, removemos la arista entre $X - Y$, de esta manera decrementamos el número de posibles conjuntos S. Las independencias condicionales deberían ser comprobadas en el orden especificado por $order(V)$.
3. Para cada tripleta de nodos (X, Y, Z) tal que X es adyacente a Y, además Y es adyacente a Z pero X no es adyacente a Z, orientar $X - Y - Z$ como $X \rightarrow Y \leftarrow Z$ sí y sólo sí Y no estaba en el conjunto S que separa a X y Z en el paso 2.
4. Repetir, hasta que no se puedan dirigir más arcos:
   - *a*) Dirigir todos los arcos necesarios para evitar la formación de nuevas *v-estructuras* (una terna de nodos con enlace dirigidos $X \rightarrow Y \leftarrow Z$ en donde no existe ninguna arista entre X y Z).
   - *b*) Dirigir todos los arcos necesarios para evitar ciclos.

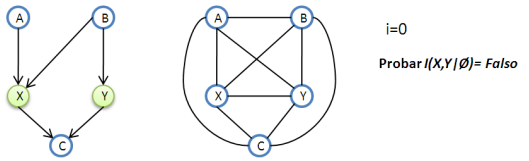
Algoritmo 2.1: Pseudocódigo del algoritmo PC

Primero el algoritmo intenta encontrar el esqueleto del grafo (es decir, la estructura subyacente del grafo no dirigido) mediante los pasos 1 y 2 del algoritmo 2.1 y posteriormente realiza algunos procedimientos para orientar las aristas del mismo mediante los pasos 3 y 4 del mismo. La idea básica es que si el conjunto de independencia es fiel al grafo, entonces no hay un enlace entre $X - Y$, sí y sólo sí no existe un subconjunto S de nodos adyacentes a X tal que $I(X, Y|S)$. Para cada par de variables $X - Y$, dicho subconjunto S será usado en pasos posteriores en el proceso de orientación del grafo.

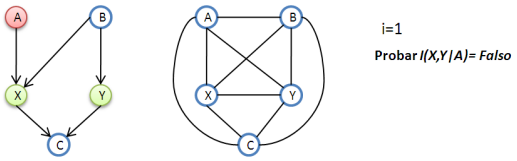
Si el conjunto de independencia es fiel al grafo y se tiene una forma perfecta de determinar cuándo $I(X, Y|S)$, entonces el algoritmo garantiza producir un grafo equivalente (que representa el mismo conjunto de independencias) al original. En la práctica estas condiciones no son garantizadas y la independencia es comprobada usualmente por medio de una prueba Chi-cuadrada basado en la medida estadística de entropía cruzada sobre la muestra [SGS93].

Como las pruebas estadísticas tienen errores, es posible no poder recuperar el grafo original. El número de errores se incrementa cuando las muestras de datos son pequeñas o el tamaño del conjunto condicionante se hace grande. En ambos casos, debido a la naturaleza frecuentista de las pruebas estadísticas, hay una tendencia a siempre decidir independencia.

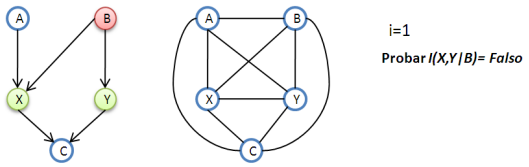
Ejemplificando el procedimiento anterior (algoritmo 2.1), se mostrara con más detalles el procedimiento de separación-d para un par de variables (es decir, el paso 2 del algoritmo 2.1). Para describir el procedimiento solamente intentaremos separar el par de nodos $X - Y$ probando primero las independencias de bajo orden sobre el grafo completo (paso 1 del algoritmo 2.1) generado sobre las variables presentes. La figura 2.3 (a, b y c) muestra los pasos de las pruebas de independencia condicional necesarios para comprobar la separación de las variables $X - Y$ sobre el grafo completo siendo encontrada la separación en la figura 2.3 (d). Nótese que cada iteración sobre el tamaño del conjunto condicionante se aplica a todos los pares de variables. En la figura se muestran, por propósitos de claridad, los pasos para estimar la dependencia/independencia del par de variables $X - Y$.



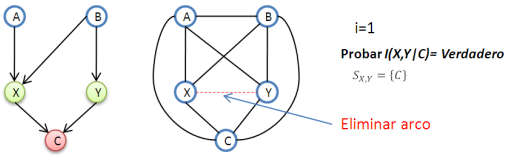
(a) Probando la separación de $X-Y$ dado $\{\phi\}$.



(b) Probando la separación de $X-Y$ dado $\{A\}$.



(c) Probando la separación de $X-Y$ dado $\{B\}$.



(d) Separacion de $X-Y$ dado $\{C\}$ encontrada.

Figura 2.3: Ejemplo del algoritmo PC para la estimación de dependencia/independencia de los nodos $X - Y$. Los incisos a, b, c muestran las pruebas realizadas para intentar encontrar independencia entre el par de variables $X - Y$ dados conjuntos condicionantes cada vez más grandes. En el inciso d se ha encontrado que el par de variables $X - Y$ es separado por C por lo que se elimina la arista que los une.

Número de pruebas de independencia condicional necesarias para el algoritmo PC. La complejidad del algoritmo depende del numero de adyacentes que tengan los nodos en el grafo. Sea k el mayor número de adyacentes para un nodo en un grafo G, y sea n el número de vértices en el grafo. Entonces el número de pruebas de independencia condicional necesitados por el algoritmo está acotado por:

$$2\begin{pmatrix} n \\ 2 \end{pmatrix}\sum_{i=0}^{k}\begin{pmatrix} n-1 \\ i \end{pmatrix}$$

Está cantidad esta acotada por:

$$\frac{n^2(n-1)^{k-1}}{(k-1)!}$$

para hacer el análisis en el peor caso, se asume que todo par de variables está separado por un subconjunto con cardinalidad k. En un caso general, el numero de pruebas de independencia condicional requeridos por grafos con cardinalidad máxima k será mucho menor. De todas formas, los requerimientos computacionales crecen exponencialmente con k.

Frente al algoritmo PC, se encuentran los algoritmos que recuperan árboles como [CL68] y poliárboles como los de [RP87] que resultan más operativos aunque el poder de descripción de estas estructuras es más reducido. Estos permiten recuperar estructuras donde la presencia de cierto tipo de ciclos está permitida, los ciclos simples (donde los nodos con descendientes directos comunes son marginalmente independientes entre sí). Existen además otros trabajos muy interesantes sobre estos tipos de métodos, para más detalle ver [Bun96].

Aprendizaje estructural utilizando métricas y técnicas de búsqueda

Todo método de aprendizaje de este tipo emplea alguna técnica de búsqueda heurística (*greedy* en su mayoría) para explorar el espacio de búsqueda. El tipo de métrica que emplean es muy variado, aunque se pueden clasificar según el principio en que se basan: la entropía, medidas bayesianas y el principio de descripción de longitud mínima, principalmente.

Medidas basadas en entropía. Los métodos basados en entropía tratan de encontrar aquella red que minimice su entropía con los datos. El principio de máxima entropía

se emplea cuando no se tiene suficiente información. Los algoritmos que emplean este principio tratan las dependencias presentes en los datos como restricciones a la distribución subyacente desconocida. Por lo tanto en aras de disminuir el *desconocimiento*, estos algoritmos extraen de los datos una lista de relaciones de dependencia significativas que son las que se busca representar en el modelo gráfico. Así, al tratar de minimizar esta medida se favorecen las conexiones entre aquellas variables que manifiestan un alto grado de dependencia. Entre estos métodos hay los que aprenden estructuras sencillas, como árboles [CL68], poliárboles [RP87] y grafos generales[HC90]. En algunos de los casos, debido a las características de las estructuras, el proceso de búsqueda (explícita) se reemplaza por un proceso analítico, lo que da lugar a algoritmos muy eficientes.

El algoritmo más conocido y utilizado para árboles es el de Chow y Liu [CL68]. Este obtiene a partir de los datos una lista de los pares de variables ordenadas en orden decreciente por el valor de la medida de información mutua, esto es, la *cantidad de información* $I(x_i, x_j)$ la cual es calculada mediante la ecuación

$$I(x_i, x_j) = \sum_{x_i, x_j} P(x_i, x_j) \log \frac{P(x_i, x_j)}{P(x_i)P(x_j)}$$

Entre las propiedades de la medida $I(x_i, x_j)$ se destaca que siempre es positiva o nula, alcanzando el mínimo (cero) cuando las dos variables son independientes. Cuanto mayor sea el valor de la cantidad de información la dependencia entre las variables será mayor. De la lista de pares ordenados se seleccionan aquellos pares que tienen valores significativos de dependencia. A partir de los pares se construye el árbol que representa una aproximación a la distribución original de los datos. Si la distribución de los datos es representable por un árbol, el algoritmo es capaz de recuperar el árbol que la representa.

El algoritmo de Rebane y Pearl [RP87] puede considerarse como una extensión para poliárboles del método de Chow y Liu. En la primera fase, el algoritmo construye el esqueleto de la estructura utilizando el método anterior (ambas estructuras la de arboles y poliárboles son simplemente conectadas). En la segunda fase se trata de orientar las aristas buscando los nodos cabeza-cabeza para luego completar la orientación de

las aristas restantes de forma que no se introduzcan nuevos patrones cabeza-cabeza. El algoritmo Kutato [HC90] para redes generales, requiere de un orden completo entre las variables. Éste, determina la estructura a partir de un grafo inconexo[6] al que le van añadiendo aquellos arcos que manteniendo el grafo sin ciclos minimiza la entropía de la red. El paso de añadir arcos intenta encontrar la relación entre variables que más restrinja la distribución subyacente. El proceso continúa hasta que se alcanza un determinado umbral.

Medidas de Descripción de Longitud Mínima (MDL). La idea de esta métrica procede de la teoría de la codificación, donde se trata de codificar una cadena en el menor número de bits, para ello se divide la cadena en subcadenas de tal forma que las cadenas más frecuentes se codifican con el menor número de bits. Una codificación de una cadena (los datos) está formada por dos partes, la descripción de la codificación del modelo utilizado y la propia codificación de los datos. El principio de descripción de longitud mínima [Ris86] establece que la mejor representación de un conjunto de datos es aquella que minimiza la suma de estos dos componentes, estableciéndose un compromiso entre éstos. Para el caso en que el modelo de codificación para los datos es una red bayesiana, la primera parte de la codificación (la descripción del modelo), se tiene que codificar la estructura gráfica para cada variable, la lista de sus padres y una lista de las probabilidades condicionadas de cada nodo. Ambas codificaciones (que se pueden medir en bits) aumentan conforme el grafo es más denso. La segunda parte de la descripción, la codificación de los datos dado el modelo, conforme mayor sea la información que nos proporcione el modelo, menor será su longitud (definida mediante una medida de entropía).

Medida Bayesiana Este tipo de métrica se basa en la filosofía de la estadística Bayesiana [CGH97, pag. 509, 538]. Tratan de maximizar la probabilidad de obtener una determinada estructura condicionada a la base de datos de que se dispone, utilizando para ello la fórmula de Bayes. La idea básica de las medidas de calidad bayesiana

[6] Un grafo es conexo si entre dos cualesquiera de sus nodos hay al menos un camino; inconexo en cualquier otro caso. Un grafo inconexo está formado por varios *componentes* conexos.

consiste en asignar a toda red un valor de calidad que es función de su probabilidad *a posteriori* (véase [Hec95]). Como ejemplo más destacado de un algoritmo de este tipo podemos citar al algoritmo K2 [CH92], el cual utiliza como métrica (debido a que se aseguran ciertas condiciones) una fórmula que establece la probabilidad conjunta de un grafo G y una base de datos D. Mediante una búsqueda local va modificando el grafo, inicialmente vacío, de tal forma que se vaya incrementando la probabilidad de la estructura resultante.

Un aspecto crítico de la teoría Bayesiana es la elección de las distribuciones a priori basadas en el conocimiento que se dispone en cada campo; si no se seleccionan cuidadosamente se puede llegar a unas distribuciones *a posteriori* inadecuadas. Así por ejemplo, la distribución *a priori* de cada red en muchos casos se supone que es uniforme.

Algoritmos de búsqueda de redes bayesianas

En las secciones anteriores se han discutido varias medidas de calidad de redes bayesianas. Estas medidas son usadas por los algoritmos de búsqueda para encontrar redes bayesianas de alta calidad. El número de posibles estructuras de red puede ser tan grande que es prácticamente imposible evaluar cada una de ellas. Esta sección presenta tres algoritmos de búsqueda que intentan buscar la red bayesiana con la mayor calidad dada una cierta información inicial y un conjunto de datos. Se tienen algunos algoritmos más representativos de búsqueda de modelos como K2, algoritmo de Buntine y CB, mencionados a continuación.

Algoritmo K2. Cooper y Herskovits [CH92] describen el uso de un algoritmo de búsqueda *greedy* para identificar la estructura más probable dado un conjunto de datos de entrenamiento. Este algoritmo asume que un orden ha sido establecido para las variables por lo que el espacio de búsqueda es reducido. El hecho de que $X_1, X_2, ..., X_n$ sea un orden de las variables implica que únicamente los predecesores de X_k en la lista puedan ser nodos padres en la red aprendida. El algoritmo también asume que todas las redes tienen igual probabilidad, pero por ser un algoritmo con método de

búsqueda tipo *greedy* no se asegura que la red resultante del proceso de aprendizaje sea la red más probable dados los datos. En particular, se usa un algoritmo que inicia asumiendo que los nodos no tienen padres, y entonces añade incrementalmente los padres cuya adición incremente la probabilidad de la estructura resultante. Cuando la adición de algún padre no incremente la probabilidad de la estructura, se detiene el proceso de añadir padres al nodo. El pseudocódigo del algoritmo K2 se muestra en el algoritmo 2.2.

Se usa la siguiente ecuación para determinar la estructura más probable dado los datos de un conjunto de variables con un ordenamiento dado

$$g(i, \pi_i) = \prod_{j=1}^{q_i} \left(\frac{(r_i - 1)!}{(N_{ij} + r_i - 1)!} \prod_{k=1}^{r_i} N_{ijk}! \right)$$

donde N_{ijk} es el cálculo relativo a π_i siendo éste ultimo el padre de x_i y relativo a la base de datos D, la cual se deja implícitamente. Además se usa una función $pred(x_i)$ que regresa el conjunto de nodos que precede a x_i en el ordenamiento dado.

La complejidad media del algoritmo K2 es $O(mu^2n^2r)$. Donde m es el número de casos en la base de datos, u es el máximo número de padres para cualquier nodo, y es proporcionado por el usuario, n es el numero de variables en el modelo y r es el máximo número de posibles valores para cualquier variable. En el peor caso, $u = n$, la complejidad de K2 es $O(mn^4r)$.

Para detalles del algoritmo K2 referirse a [CH92].

Entrada: Un conjunto de n nodos, un orden en los nodos, un limite en el número de padres que un nodo puede tener, una base de datos D conteniendo m casos.

Salida: Para cada nodo, la lista de padres del nodo.

Para $i = 1$ a n hacer

1. $\pi_i = \phi$
2. $p_{old} = g(i, \pi_i)$
3. $ok_{toproceed} = true$
4. Mientras $ok_{toproceed}$ Y $|\pi_i| < max_{padres}$ hacer

   a) sea nodo $z \in pred(i) - \pi_i$ que maximice $g(i, \pi_i \bigcup \{z\})$;
   b) $p_{new} = g(i, \pi_i \bigcup \{z\})$
   c) Si $p_{new} > p_{old}$ entonces

      1) $p_{old} = p_{new}$
      2) $\pi_i = \pi_i \bigcup \{z\}$

   d) caso contrario

      1) $ok_{toproceed} = false$

5. Imprimir('Nodo:', x_i, 'Padres del nodo: ', π_i)

Algoritmo 2.2: Pseudocódigo para el algoritmo K2

El principal problema del algoritmo K2 es que necesita conocer *a priori* un orden entre las variables. Si no se tiene este orden, es posible seleccionar un orden aleatorio, donde la estructura resultante puede ser optimizada posteriormente.

Algoritmo de Buntine. Un algoritmo que no requiere un ordenamiento de los nodos fue propuesto por Buntine [Bun91]. Este algoritmo comienza con un conjunto vacío de padres y en cada paso se agrega un nuevo enlace de tal manera que no se formen

ciclos y que se maximice el incremento de la métrica de calidad. El proceso se repite hasta que no es posible incrementar la calidad o hasta que se haya encontrado una red completa.

Algoritmo CB. Singh y Voltara [SV95], propusieron una extensión al algoritmo K2 llamado CB. Este algoritmo utiliza pruebas de independencia condicional para generar una *buena* ordenación de nodos a partir del conjunto de datos y luego usa el algoritmo K2 para generar la red bayesiana a partir de una muestra de entrenamiento D usando el ordenamiento anterior. Empieza con un grafo no dirigido completo con todas las variables, luego remueve aquellos arcos entre nodos adyacentes que son condicionalmente independientes. CB orienta los ejes en el grafo y obtiene un nuevo ordenamiento de las variables. Luego se usa el algoritmo K2 para construir la red correspondiente.

2.3. Conclusiones

Generalmente los algoritmos de construcción de la estructura de redes bayesianas pueden agruparse en dos categorías: una categoría de algoritmos usan un método de búsqueda heurístico para construir un modelo y posteriormente lo evalúan usando algún método de ajuste a los datos. El problema es resuelto buscando en el espacio de posibles modelos de redes bayesianas intentando encontrar la red con la mejor evaluación. La otra categoría de algoritmos construye la red bayesiana analizando las relaciones de dependencia entre nodos. La relación de dependencia es medida usando algún tipo de prueba de independencia condicional. El ejemplo principal de esta categoría es el algoritmo PC [SGS93].

Los procedimientos de búsqueda y evaluación han recibido más atención debido a algunas claras ventajas [CHM97]. Una es que el aprendizaje basado en análisis de dependencias realiza desde un principio decisiones categóricas, basadas en pruebas estadísticas que podrían ser erróneas y afectando el comportamiento futuro del algoritmo. Otra es que los procedimientos de búsqueda y evaluación permiten comparar modelos muy diferentes mediante una función de evaluación que puede ser interpreta-

do como la probabilidad de creencia del verdadero modelo. Como consecuencia, podemos seguir una alternativa bayesiana considerando muchos modelos alternativos, cada uno de ellos con su correspondiente probabilidad, y usarlos para determinar la decisión posterior (por promedio de modelos).

Por otra parte, el algoritmo PC tiene algunas ventajas. Una de estas es que tiene una base intuitiva y bajo algunas condiciones ideales [Pea88, CH92] como la condición de Markov, la condición de Fidelidad y las decisiones estadísticas correctas, se garantiza la recuperación de un grafo equivalente al modelo verdadero a partir de los datos. El punto base de la utilización del algoritmo PC en este trabajo es que éste puede proveer un conjunto de estrategias que pueden ser combinadas con otras ideas para producir buenos algoritmos de aprendizaje los cuales puedan ser adaptados a diferentes situaciones [AGOM03]. Un ejemplo de esto es cuando en [vDvdGT03] proponen un nuevo enfoque que combinan las ventajas de ambos tipos de algoritmos de aprendizaje. En su enfoque, usan pruebas de independencia con los conjuntos condicionantes de orden cero y uno para construir un grafo no dirigido en la red bajo construcción. La red es usada para restringir el espacio de búsqueda para un procedimiento de tipo búsqueda y evaluación.

Desafortunadamente, hay muchos dominios de interés, donde la proporción del número de observaciones al número de variables es bajo (ejemplo, cuando tenemos bases de datos muy pequeñas), la selección de un umbral para las pruebas de independencia condicional en esta clase de algoritmos puede ser difícil, y el uso repetido en las pruebas puede causar inconsistencias [DD99]. Además debido a la naturaleza de las pruebas a encontrar independencia cuando la cantidad de datos es escasa los modelos encontrados tienden a tener muy pocos enlaces teniendo modelos muy sencillos y poco cercanos al modelo verdadero.

Para tratar esta dificultad, se propone la utilización de conocimiento relativo al dominio del problema mediante técnicas conocidas como aprendizaje por transferencia o transferencia inductiva. En el siguiente capítulo se describe la aplicación del aprendizaje por transferencia utilizando otras representaciones, y después se presenta su utilización en el aprendizaje de redes bayesianas.

Capítulo 3

Aprendizaje por transferencia

3.1. Introducción

La mayoría de las investigaciones en Aprendizaje Computacional se han enfocado en aprender un problema aislado, donde un modelo para un problema es inducido de un conjunto de ejemplos de entrenamiento sin considerar aprendizajes previos. Aunque se ha tenido un gran éxito en este tipo de trabajos, es claro que no considera ciertos aspectos fundamentales en la forma del aprendizaje humano y animal. En contraste, los humanos toman la ventaja de aprendizajes previos reteniendo el conocimiento de la tarea y transfiriendo este conocimiento cuando se aprende una nueva tarea relacionada. Los humanos enfrentamos el aprendizaje de una tarea nueva equipados con el conocimiento obtenido de aprendizajes previos. Por ejemplo, el aprendizaje de un nuevo idioma podría facilitarse si se conocen otros idiomas parecidos al que se quiere aprender, porque ambos idiomas podrían poseer estructuras gramaticales o sintácticas similares que favorezcan un aprendizaje más eficiente. Es natural intentar aplicar estas observaciones al aprendizaje automático, por lo que ya se han desarrollado técnicas, conocidas como aprendizaje por transferencia o transferencia inductiva que toman ventajas del conocimiento adquirido de tareas similares.

El aprendizaje de múltiples tareas (MTL) es un método de transferencia inductiva[1] que mejora la generalización utilizando implícitamente información del dominio en el aprendizaje de tareas relacionadas. La meta de la transferencia inductiva es utilizar fuentes adicionales de conocimiento para mejorar el desempeño en el aprendizaje de la tarea actual. Las fuentes de información pueden tomarse de varias formas [Car96], incluyendo:

- información del dominio de conocimiento,
- modelos para la misma tarea de aprendizaje obtenidos con otros métodos de aprendizaje,
- modelos para la misma tarea aprendidos desde diferentes distribuciones,
- señales de entrenamiento para o desde modelos aprendidos de tareas relacionadas, por ejemplo, cuando se utilizan ejemplos de entrenamiento de tareas relacionadas en el aprendizaje mediante redes neuronales.

La transferencia inductiva puede ser usada para mejorar la exactitud de la generalización, la velocidad del aprendizaje, y la legibilidad de los modelos aprendidos. En este trabajo nos enfocamos en mejorar la exactitud de los modelos, especialmente cuando se tiene poca información para su aprendizaje mediante señales de entrenamiento aprendidas desde tareas relacionadas.

3.2. Aprendizaje por transferencia

La metodología estándar en el aprendizaje computacional es aprender una tarea a la vez. Sin embargo, cuando la cantidad de ejemplos de entrenamiento disponibles para el aprendizaje es relativamente pequeña, generalmente no se tienen buenos desempeños. En muchas situaciones, existen datos disponibles para múltiples problemas

[1] El aprendizaje por transferencia o transferencia inductiva es un problema de investigación en el área de Aprendizaje Computacional enfocado en almacenar el conocimiento obtenido mientras se resuelve un problema y aplicarlo a problemas diferentes pero relacionados.

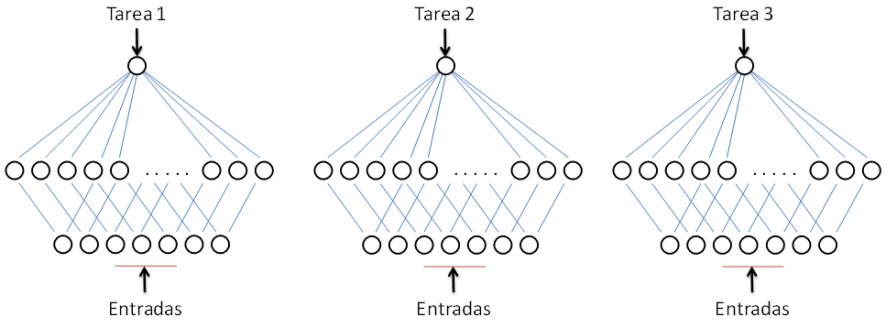


Figura 3.1: Aprendizaje de tres tareas sencillas mediante redes neuronales, cada una con las mismas entradas.

relacionados. En estos casos, la transferencia inductiva [Car97, Bax97, Thr96] sugiere que podría ser posible aprender modelos más exactos por medio de transferencia de información entre los problemas.

Muchos métodos de aprendizaje por transferencia están esencialmente basados en los trabajos desarrollados por Caruana [Car97]. La idea básica es aprender m tareas relacionadas en paralelo utilizando redes neuronales, por lo que todas las tareas están definidas en el mismo espacio de entrada. Las diferentes tareas son relacionadas virtualmente por medio de nodos ocultos. La esperanza es que con el entrenamiento y alternando los ejemplos de las diferentes tareas, las características comunes sean aprendidas rápidamente. Por ejemplo, la figura 3.1 muestra tres redes neuronales separadas. Cada red es una función de las mismas entradas, y tiene una salida. Cada red es entrenada por separado utilizando retro-propagación. Como las tres redes no están conectadas, no es posible para una red aprender con la ayuda de otras redes.

La figura 3.2 muestra una red neuronal con las mismas entradas que la figura 3.1, pero que posee tres salidas, una para cada red de la figura 3.1. El aprendizaje por retro-propagación es realizado en paralelo para las tres salidas en una red multi-tarea. Como las tres salidas comparten una capa común de nodos ocultos, es posible que la representación interna en las capas ocultas sea utilizada por varias tareas.

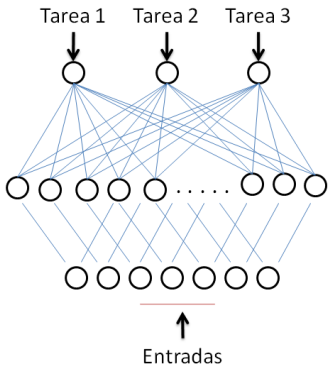


Figura 3.2: Aprendizaje de múltiples tareas (tres) con las mismas entradas.

En la literatura se tienen varios algoritmos de aprendizaje por transferencia utilizando diversas representaciones: representaciones internas basadas en redes neuronales, aprendizaje de medidas de distancia, una nueva representación de los datos. En algunas soluciones de aprendizaje de múltiples tareas todas las tareas no son relevantes para las otras. En [XLCK07] se asume que las tareas se agrupan en conjuntos, y las que vienen del mismo agrupamiento son generadas con los mismos parámetros. La tareas pueden estar relacionadas en varias maneras. Por ejemplo, la relación entre las tareas puede modelarse asumiendo que las funciones aprendidas son cercanas unas a las otras por medio de una función [BH03, ZGY05]. Las tareas también pueden ser relacionadas en la forma en que ellas intercambian una representación común [BDS03, Car97, Bax00]. A continuación se mencionarán algunos trabajos.

El algoritmo de agrupamiento de tareas (TC) descrito en [TO96], ha sido diseñado para soportar el aprendizaje en tareas de clasificación (binaria) para conjuntos grandes de datos, definidas sobre el mismo espacio de entrada. Transfiere el conocimiento *selectivamente,* desde el conjunto de tareas más relacionado únicamente utilizando medidas de distancia. Para llevar a cabo esto, el algoritmo TC calcula la similitud de información mutua entre tareas, y construye una jerarquía de clases de tareas. Cuando una tarea nueva se presenta, se determina el agrupamiento de tareas más relacionado desde las tareas anteriormente aprendidas. El aprendizaje es transferido selectivamente desde

un único agrupamiento. Un resumen del procedimiento se muestra a continuación:

1. El algoritmo TC clasifica por vecinos más cercanos usando una *medida de distancia euclidiana globalmente pesada.*
2. Se transfiere el conocimiento entre las múltiples tareas aprendidas.
3. Se enfoca en transferir las tareas más relacionadas. TC calcula la matriz de transferencia de tareas, la cual mide la relación mutua entre tareas por medio de la métrica de vecinos más cercanos.
4. Construye una jerarquía de tareas agrupando las tareas por medio de la matriz de transferencias.
5. Cuando llega un nuevo objeto para aprender, la medida de distancia es transferida selectivamente desde el grupo de tareas más relacionado únicamente.

En [KP07] presentan un problema llamado aprendizaje de subtareas relevantes, una variante del aprendizaje de multitareas. Su objetivo es construir un clasificador para una tarea la cual posee muy pocos datos. Este problema lo tratan asumiendo que los datos de todas las tareas son una mezcla de ejemplos relevantes e irrelevantes para alguna tarea, lo que significa que contienen algunos ejemplos que son clasificados de la misma forma que la tarea de interés; y modelan la parte irrelevante con un modelo flexible, basado en regresión logística, tal que éste no distorsione al modelo con datos relevantes.

Lee y Carrier [LGC07] proponen una técnica para transferir aprendizaje en el contexto de árboles de decisión, especialmente cuando la tarea objetivo es un superconjunto de las tareas fuente. El algoritmo aprende las nuevas tareas objetivo desde un modelo parcial de un árbol de decisión, inducida por el algoritmo ID3 [Qui86], el cual captura el conocimiento de tareas previamente aprendidas. Como es un algoritmo de aprendizaje de tipo semi-incremental, actualiza el objetivo después de cada instancia de entrenamiento, pero requiere más de una iteración sobre este conjunto. Primero, identifica los atributos de la tarea objetivo que no se encuentran en los árboles de las tareas fuente y

determina el orden en el cual éstos atributos deberían ser considerados. Posteriormente, aplica transformaciones a los árboles de las tareas fuente asignando los atributos faltantes de la tarea objetivo en el lugar correcto, junto con una etiqueta asociada para la tarea objetivo. Para mayor información referirse a [LGC07].

Mihalkova y Mooney [MM06] han propuesto una forma de aprendizaje por transferencia para redes lógicas de Markov (MLN). Una MLN consiste de un conjunto de *fórmulas lógicas de primer orden*, con un peso aplicado a cada una de ellas y proveen un modelo para la distribución de probabilidad del conjunto de variables. En su propuesta, primero diagnostican la MLN provista desde la tarea fuente y explota la similitud entre tareas enfocándose en reaprender únicamente las partes incorrectas. Su enfoque se puede dividir en dos etapas: primero, el algoritmo inspecciona la MLN dada y determina para cada fórmula de su estructura si ésta es muy general, específica o no requiere cambios. El propósito de este paso es enfocarse en buscar nuevas fórmulas en las partes que la MLN necesite actualizar. En la segunda etapa, se procede a actualizar las clausulas, especializando las marcadas como muy generales y generalizando las marcadas como muy específicas. Para mayor información referirse a [MM06].

3.3. Aprendizaje por transferencia en redes bayesianas

Como una alternativa para compensar la falta de datos en el aprendizaje de redes bayesianas se puede incorporar conocimiento de expertos. Un trabajo reciente en esta dirección es propuesto por [RD03], donde consideran la combinación de las opiniones de múltiples expertos. En este trabajo de tesis no se le ha considerado propiamente dentro del área de aprendizaje por transferencia, pero la utilización de varias fuentes de información provenientes de diversos expertos humanos que ofreciendo soluciones diferentes pero similares, merece la inclusión en esta sección. En su trabajo, desarrollan un método bayesiano en el que el conocimiento experto es codificado como una distribución inicial de las posibles estructuras, y los datos son usados para refinar la estructura y el aprendizaje de los parámetros. El conocimiento experto es representado como un *meta* red bayesiana, donde las decisiones de los expertos son codificadas como la probabilidad de que un arco exista (y su dirección) entre dos variables. De

esta forma, el conocimiento experto es usado para obtener una distribución de probabilidad sobre las estructuras. La *mejor* estructura es obtenida utilizando un método de búsqueda de ascenso de colina *(hill climbing*), posteriormente los parámetros para la estructura son calculados en base a los datos.

El aprendizaje simultáneo de múltiples tareas relacionadas ha sido desarrollado empíricamente [AZ05, BH03, EMP05, Jeb04, ZGY05] y teóricamente [AZ05, BDS03], mostrando que en la mayoría de las ocasiones mejora el desempeño relativo a aprender las tareas aisladamente. Este es el caso, por ejemplo, cuando solamente unos pocos datos son disponibles; existe una ventaja en *combinar* datos a través de todas las tareas relacionadas utilizando estas técnicas. Pocos son los trabajos relativos a la obtención de un modelo de red bayesiana utilizando técnicas de aprendizaje por transferencia, a continuación se detallará el más cercano avocado a nuestra problemática.

En [NMC07] consideran el problema de aprender estructuras de redes bayesianas para tareas relacionadas. Consideran que existen k conjuntos de datos para problemas relacionados, y proponen una medida de calidad y un método de búsqueda para aprender simultáneamente las estructuras de todas las redes bayesianas, una para cada tarea. Asumen que los parámetros de las diferentes redes son independientes, y definen la distribución de probabilidad posterior de un conjunto de k estructuras dados sus k conjuntos de datos haciendo los parámetros de las diferentes redes independientes *a priori* como sigue:

$$P(G_1, ..., G_k | D_1, ..., D_k) \propto P(G_1, ..., G_k) \prod_{p=1}^{k} P(D_p | G_p)$$

donde $G_1, ..., G_k$ es la estructura de las k redes bayesianas, $D_1, ..., D_k$ es el conjunto de los k conjuntos de datos respectivos.

Usando una medida para la probabilidad inicial que penaliza cada arco que se encuentra en una estructura y no se encuentra en las otras mediante:

$$P(G_{1,...,}G_k) = Z_{\delta,k} \prod_{1 \leq s \leq k} P(G_s)^{\frac{1}{1+(k-1)\delta}} \prod_{1 \leq s \geq t \leq k} \left(\prod_{(X_i,X_j) \in G_s \triangle G_t} (1-\delta) \right)^{\frac{1}{k-1}}$$

Definen un algoritmo de búsqueda tipo *greedy* para encontrar la mejor configuración a través del espacio de todas las configuraciones posibles, donde una nueva configuración es obtenida añadiendo/eliminando/invirtiendo un solo arco en todas las estructuras (teniendo a n^2 como el espacio de búsqueda para probar cada par de nodos y 3^k el número de operaciones elementales sobre los arcos para las k redes presente, hace que la búsqueda de la mejor estructura dentro de una configuración sea del orden de $O(n^2 3^k)$). Para mayores detalles del algoritmo referirse a [NMC07].

3.4. Conclusiones

Como hemos revisado en este capítulo, la utilización de las técnicas de aprendizaje por transferencia en el área de aprendizaje computacional es relativamente reciente, y aunque se han desarrollado algunos trabajos al respecto aun no se tiene una única forma de como medir la similitud entre las tareas que se utilizan en los algoritmos de aprendizaje. Posiblemente esto se deba a la gran cantidad de algoritmos y aplicaciones distintas existentes en la literatura. Tomemos por ejemplo los trabajos desarrollados por Caruana [Car97]. En su propuesta, tratan de resolver el problema de aprender simultáneamente m tareas relacionadas y modelan el aprendizaje por transferencia utilizando nodos ocultos en algoritmos de redes neuronales. Como la capa oculta es compartida por varias tareas al mismo tiempo, parece ser un medio muy natural para transferir información faltante entre tareas. Sin embargo la cantidad de información a transferir está limitado por la cantidad de ejemplos de entrenamiento disponibles para cada tarea durante la fase de entrenamiento y no se enfoca en mejorar un solo modelo. El trabajo de [KP07] presenta un problema llamado aprendizaje de subtareas relevantes cuyo objetivo es mejorar un clasificador para una sola tarea, y emplean modelos de regresión logística para modelar datos relevantes e irrelevantes de cada

tarea auxiliar.

Existen por tanto algunos métodos propuestos para aprovechar y medir la similitud de tareas utilizando diversas representaciones pero hay poco trabajo en cuanto al aprendizaje de redes bayesianas. Los trabajos previos para redes bayesianas [NMC07] se enfocan a aprender varias tareas simultáneamente, y no tanto en transferir conocimiento de una o varias tareas a otra. A diferencia de estos trabajos la presente propuesta trata el problema de aprender un modelo para un dominio con pocos datos aprovechando los datos disponibles de dominios relacionados.

El trabajo en esta tesis propone utilizar los métodos de aprendizaje por transferencia abordando los dos aspectos de su aprendizaje: el aprendizaje estructural y paramétrico. Como se mostrará con más detalle en el capítulo 4, se intenta obtener para el aprendizaje estructural las máximas similitudes en los dominios auxiliares y con esto transferir conocimiento faltante en la fase de construcción de la red. El capítulo 5 presenta dos propuestas de aprendizaje paramétrico basadas en una combinación de medidas de los parámetros presentes en las tablas de probabilidad condicional de la red objetivo y de las provenientes de tareas auxiliares.

Capítulo 4

Aprendizaje estructural utilizando aprendizaje por transferencia

4.1. Introducción

El método de aprendizaje estructural de redes bayesianas propuesto se puede ver como una extensión al conocido método de aprendizaje estructural algoritmo PC (ver sección 5) basado en pruebas de independencia condicional. Este nuevo método propuesto tiene la característica principal de utilizar un conjunto de datos auxiliares, provenientes de dominios relacionados[1] al dominio que se quiere aprender, sobre los cuales se apoya el algoritmo para incrementar la exactitud en la recuperación de la estructura del conjunto de datos objetivo cuando de este se poseen muy pocos datos (relativos al tamaño del modelo que se quiere recuperar).

[1]Consideramos a dos dominios como similares o relacionados cuando ambos dominios comparten similitudes en la distribución de los datos, es decir, no se consideran dominios iguales porque no poseen la misma distribución en los datos.

4.2. Justificación del método propuesto

Si el conjunto de independencias presente en los datos está contenido en el grafo y se tiene una forma perfecta de determinar $I(X, Y|S)$, entonces algoritmos como PC garantizan producir un grafo equivalente (representa el mismo conjunto de independencias) al original. Sin embargo, en la práctica ninguna de estas condiciones es verificada. La independencia es decidida por las pruebas de independencia condicional basados sobre un conjunto de datos D. La forma usual de hacer estas pruebas es por medio de la prueba ji-cuadrada basada en la estadística de entropía cruzada medida en la muestra [SGS93] utilizando un umbral de significancia que no siempre es posible obtener de forma satisfactoria[2]. El número de errores de las pruebas estadísticas se incrementa cuando la muestra es pequeña o la cardinalidad del conjunto condicionante S es grande[3] [SGS93, p. 116] y si las prueban de independencia estadística regresan sentencias de dependencia/independencia incorrectas, estos errores podrían afectar la estructura gráfica. La función de prueba de independencia $I_F(X, Y|S)$ utilizando aprendizaje por transferencia (presentada en la ecuación 4.4) intenta dar solución a las desventajas presentes en este tipo de algoritmos cuando se tienen muestras relativamente pequeñas mediante los siguientes puntos:

- Se utiliza aprendizaje por transferencia selectivamente desde los dominios más similares y relacionados utilizando una función que mide la similitud en las pruebas de independencia estadística.
- Se transfiere más información hacia el dominio objetivo cuando se tiene insuficiencia de datos de acuerdo a la complejidad de la prueba de hipótesis de

[2] Utilizar niveles de significancia altos implica una mayor confianza en los resultados de las pruebas estadísticas, pero se requiere una gran cantidad de datos. Cuando no se dispone de suficientes datos debido a la tendencia de las pruebas, la cantidad de independencias encontrados en los datos es alta, lo que se traduce en estructuras pobremente conectadas. La elección del nivel de significancia influye directamente sobre la topología de la red calculada. Si el nivel es demasiado alto con respecto al tamaño de los datos disponibles, las dependencias débiles escaparán de la identificación y la red tendrá aristas perdidas. Por otro lado, si el nivel de significancia es demasiado bajo, correlaciones coincidentes en los datos serán confundidas con dependencias falsas y la red incluirá mas aristas incorrectas.

[3] Esto es porque el número de los datos requeridos para las pruebas estadísticas se incrementa exponencialmente con la cardinalidad del conjunto condicionante.

independencia condicional en la fase de aprendizaje de la red. En este caso la información proveniente de los dominios auxiliares más similares tiende a ser más relevante en la combinación final conforme se incrementa la complejidad de las pruebas, sin embargo, si los dominios auxiliares no son lo suficientemente relacionados la recuperación final de la estructura podría ser menor a no utilizar aprendizaje por transferencia.

Al tratarse de una extensión al método base, el costo computacional del proceso de aprendizaje se incrementa debido al cálculo de la transferencia de información en las pruebas de independencia condicional, aunque este costo se mantiene en un factor lineal con respecto al algoritmo PC debido a la transferencia selectiva desde un solo dominio relacionado para cada prueba de cada par de variables durante el proceso de aprendizaje.

A continuación se describe el método propuesto.

4.3. Descripción del método propuesto

Asumamos que tenemos un dominio de interés con un pequeño conjunto de datos, D_0; y n conjuntos de datos adicionales, $D_1, ..., D_n$, de dominios auxiliares relacionados. Se asume que el conjunto de atributos del dominio objetivo es un subconjunto de los atributos de los dominios auxiliares[4]. Asociado al dominio de interés existe un modelo basado en una representacion por medio de una red bayesiana, BN_0, con una estructura G_0 y sus parámetros P_0. Entonces el problema es obtener BN_0 a partir de $D_0, D_1, ..., D_n$, lo cual se espera que sea una buena aproximación al modelo que obtendríamos si dispusiéramos de un *gran* conjunto de datos D_0. Iniciaremos aprendiendo la estructura de la red bayesiana, posteriormente, en el capítulo 5 se detallará el aprendizaje de sus parámetros.

[4]Es posible establecer un mapeo entre los atributos del dominio objetivo y los dominios auxiliares con el fin de usar el mismo conjunto de atributos para el algoritmo propuesto. La recuperación automática de este mapeo es un problema de investigación muy interesante y no se ha tomado en cuenta en este trabajo.

De forma similar al algoritmo PC iniciaremos con un grafo no dirigido completamente conectado, y mediremos las independencias condicionales para cada par de variables dado un subconjunto de otras variables mediante una prueba de independencia estadística. Con esto obtendremos un conjunto de medidas de independencia para cada par de variables en el dominio objetivo, I_0, y de forma similar para cada dominio auxiliar, $I_1, ..., I_n$. Posteriormente combinaremos estas medidas para construir la estructura del dominio objetivo, G_0. En la segunda fase, siguiendo al algoritmo PC, se procede a dirigir las aristas del esqueleto encontrado en la fase anterior.

El algoritmo se basa en la combinación de medidas obtenidas por la existencia de una prueba de independencia estadística que puede decidir cuando dos variables $\{X, Y\}$ son independientes dado un subconjunto S, dígase $I(X, Y|S)$ en los dominios objetivo y auxiliares. Como estas pruebas de independencia estadística no son confiables cuando se posee una cantidad limitada de datos, necesitamos definir una función para medir la fiabilidad para éstas pruebas, la cual será presentada en la ecuación 4.1.

La medida de entropía cruzada utilizada en el algoritmo PC depende del tamaño del conjunto de datos; se ha demostrado empíricamente en [FY96] que el error de esta prueba es asintóticamente proporcional a $\frac{\log N}{2N}$, donde N es el tamaño del conjunto de datos. Basados en esto, definimos la siguiente función[5] para estimar la medida de fiabilidad, α, de las pruebas de independencia entre dos variables X y Y dado un subconjunto condicionante S:

$$\alpha(X, Y|S) = 1 - \frac{\log N}{2N} \times T \tag{4.1}$$

donde $T = |X| \times |Y| \times |S|$, donde $|X|$ es el tamaño del dominio de X, $|Y|$ de Y, y $|S|$ de S. Si la diferencia comienza a ser negativa definimos esta función a un valor muy pequeño, $\alpha = 0{,}05$. Esta función es proporcional a la fiabilidad de la prueba de independencia condicional. Utilizamos este término para cuantificar la medida de independencia para los datos objetivo y auxiliares antes de combinarse, lo que nos permitirá observar la fiabilidad de las pruebas de independencia para los conjuntos

[5] Muy similar a la función umbral presentada en [SZ06] utilizada para filtrar dependencias no significativas.

de datos de interés y poder transferir conocimiento cuando sea necesario desde los dominios auxiliares (conforme a la función de combinación presentada posteriormente en la ecuación 4.4).

Antes de iniciar la construcción del modelo, necesitamos estimar la similitud de los datos disponibles de los dominios auxiliares con respecto al dominio objetivo, la cual será requerida en pasos posteriores del algoritmo, como un indicador de transferencia entre los dominios auxiliares y objetivo. Para esto, realizamos un conteo del número de concordancias en las pruebas de independencia condicional para todos los pares de variables del dominio objetivo para cada dominio auxiliar, SG_n, es decir

$$SG_{Dn} = dep + ind$$

donde dep y ind es el conteo de concordancias en las afirmaciones de dependencia e independencia[6], respectivamente, entre el dominio objetivo y el dominio auxiliar Dn. Utilizaremos esta medida como la similitud global del dominio objetivo con cada dominio auxiliar lo que nos permitirá reconocer los dominios auxiliares más similares al dominio objetivo. Adicionalmente, como es posible tener solo similitudes parciales con el dominio objetivo, necesitamos calcular la similitud local, SL_{Dn}, entre las variables analizadas durante la construcción del modelo. Durante el proceso de construcción del modelo, al analizar $I(X, Y|S)$ en el dominio objetivo, mediremos la similitud entre el dominio objetivo y cada dominio auxiliar, y el dominio auxiliar que presente mayor similitud, nombremosle D_{XY}, será el que se utiliza como fuente de transferencia de información . Es decir, D_{XY} es obtenida mediante la siguiente función:

$$D_{XY} = arg\,max_{Dn \in D_{aux}} (SL_{Dn} SG_{Dn}) \tag{4.2}$$

donde D_n es el dominio analizado perteneciente a los dominios auxiliares D_{aux} y

[6] Las afirmaciones de (in)dependencia pueden ser comprobadas por medio de pruebas estadísticas utilizando un umbral de significancia muy pequeño. En este trabajo se ha utilizado el valor α, definido en la ecuación 4.1, como umbral de significancia ya que permite obtener confiabilidades mas altas conforme mas datos se tenga a disposición para dichas pruebas.

$$SL_{Dn} = \begin{cases} 1{,}0 & \text{Si } I_0(X,Y|\theta) = I_{Dn}(X,Y|\theta) \\ 0{,}5 & \text{Si } I_0(X,Y|\theta) \neq I_{Dn}(X,Y|\theta) \end{cases} \quad (4.3)$$

La similitud local, SL_{Dn}, es calculada probando si coinciden las decisiones de independencia para el par de variables XY analizado. $I_0(X,Y|\theta)$ y $I_{Dn}(X,Y|\theta)$ denotan las pruebas de independencia estadística sobre el dominio objetivo y el dominio auxiliar Dn. Como puede observarse en la función, si las decisiones de independencia coinciden, el valor de transferencia calculado es completo, si no coinciden, es reducido en un factor de $0{,}5$[7]. Finalmente, D_{XY} será utilizado como fuente de transferencia de información sobre el par de variables X, Y del dominio objetivo. El valor de transferencia que influenciará al dominio objetivo en el proceso de construcción de la red está dado por

$$D^*_{XY} = SL_{D_{XY}} SG_{D_{XY}}$$

Ya que tenemos la máxima similitud para el par $\{X,Y\}$, D^*_{XY}, encontrada en el dominio D_{XY}, lo utilizamos como fuente de transferencia futura en la construcción de la red objetivo. Combinamos las medidas de independencia de este dominio con las del dominio objetivo usando la siguiente ecuación:

$$I_F(X,Y|S) = (\alpha_0(X,Y|S) \times sgn(I_0(X,Y|S))) + D^*_{XY}\left(\alpha_{D_{XY}}(X,Y|S) \times sgn(I_{D_{XY}}(X,Y|S))\right) \quad (4.4)$$

donde $sgn(I)$ es $+1$ si las pruebas de independencia son positivas (las variables son independientes dado un subconjunto S) y -1 de otra forma, $\alpha_0(X,Y|S)$ es la medida

[7] Una mayor o menor influencia ejercida por los dominios auxiliares en el dominio objetivo es obtenida ajustando los valores para las constantes en la función SL_{Dn}. Si los valores son incrementados, se aumenta la influencia mínima y máxima para el dominio auxiliar seleccionado y si se decrementa, disminuye la influencia del dominio auxiliar. Los valores presentados son los que se utilizaron a lo largo de los experimentos y parecen presentar buenos resultados ejerciendo un equilibrio entre la influencia de los dominios auxiliares y el propio modelo del domino objetivo respaldado por sus propios datos.

de confiabilidad de las pruebas de independencia estadística sobre el dominio objetivo, de forma similar, $\alpha_{D_{XY}}(X, Y|S)$ es la medida de confiabilidad de la prueba de independencia estadística dada al dominio mas similar, ambas dadas sobre el par de variables $\{X, Y\}$.

Finalmente usamos los resultados de las medidas de independencia combinadas $I_F(X, Y|S)$ para iniciar la construcción de la estructura de la red bayesiana. Notemos que el factor D^*_{XY} y D_{XY} define la cantidad de transferencia de información y el dominio auxiliar respectivamente durante el proceso de construcción de la red, por lo que podemos ahorrar costo computacional almacenando los resultados de los cálculos futuros en las siguientes pruebas de independencia para conjuntos condicionantes de orden mayor.

Esta función de combinación de pruebas de independencia estadística, $I_F(X, Y|S)$, la podemos dividir en dos partes para su análisis:

1. La primera parte, $(\alpha_0(X, Y|S) \times sgn(I_0(X, Y|S)))$, calcula el grado de fiabilidad que se puede esperar debido a la complejidad de la prueba de independencia condicional y la cantidad de datos presente. Si las variables de la prueba son dependientes/independientes se asigna un valor como el grado de la fuerza de dependencia/independencia encontrado en el dominio objetivo.

2. La segunda parte de la función de combinación,

$$D^*_{XY}\left(\alpha_{D_{XY}}(X, Y|S) \times sgn(I_{D_{XY}}(X, Y|S))\right)$$

   primero define el grado de transferencia D^*_{XY} del dominio auxiliar mas similar D_{XY} encontrado previamente. Después aplica la misma función de fiabilidad al dominio seleccionado D_{XY}. Si las variables de la prueba son dependientes/independientes se asigna un valor como el grado de la fuerza de dependencia/independencia en el dominio auxiliar.

Posteriormente al combinar los resultados, se puede observar que siempre se asigna una mayor confianza al dominio objetivo aun cuando se cuenta con una similar cantidad

de datos para los dominios auxiliares. El grado de influencia de los dominios auxiliares está acotado por $(0, 1]$.

Un resumen del procedimiento, que incluye la función de prueba de independencia con aprendizaje por transferencia (ecuación 4.4), es dado en el algoritmo 4.1. En este algoritmo, ADJ_X es el conjunto de nodos adyacentes a X en el grafo G'. La base es que si el conjunto de independencias es fiel al grafo, entonces no habrá una arista entre XY, si y solo si hay un subconjunto S de nodos adyacentes a X tal que $I(X, Y|S)$. Para cada par de variables, S_{XY} contendrá este conjunto, si es encontrado. Éste será usado en la etapa de orientación del algoritmo 4.1 (pasos 4-6).

Una vez que obtuvimos la estructura del modelo, estimamos los parámetros también usando la información de los dominios auxiliares (ver sección 5).

4.4. Conclusiones

En este capítulo se presentó la utilizacion de aprendizaje por transferencia aplicado a un método de aprendizaje estructural de redes bayesianas basado en detección de independencia. El método propuesto usa pruebas de independencia condicional, combinando medidas del dominio objetivo con las obtenidas en los dominios auxiliares. Para la combinación de estas medidas se realiza un cálculo de la similitud con los dominios objetivos, y aquellos que son más similares, se seleccionan como fuente de información para la combinación. Cuando los dominios están suficientemente relacionados, el método permite tomar ventaja de estas relaciones mejorando la precisión de aquellos dominios, que por falta de datos, las pruebas de independencia condicional no tienen la precisión requerida presentando fallas. También se presenta una manera de pesar los resultados de las pruebas de independencia condicional, en base a la cantidad de datos utilizados en las mismas, esto lo traducimos como una medida de fiabilidad, y nos permite medir los requerimientos de transferencia de información en la función de combinación final presentada en la ecuación 4.4. Finalmente la unión de estos dos aspectos brindan la posibilidad al método propuesto de recuperar con mayor precisión la estructura de redes bayesianas cuando se tienen pocos datos utilizando

El procedimiento toma como entrada una base de datos, D_0, sobre un conjunto de variables $X_1, ..., X_n$, un conjunto de bases de datos de problemas auxiliares $D_1, ..., D_n$, y una prueba de independencia condicional que utiliza métodos de aprendizaje por transferencia $I_F(X, Y|S)$ dado en la ecuación 4.4.

1. Iniciar con un grafo completo no dirigido G'
2. i=0
3. Repetir
   - *a*) Para cada $X \in \mathbb{X}$
     - 1) Para cada $Y \in ADJ_X$
       - Probar si $\exists S \subseteq ADJ_X - \{Y\}$con $|S| = i$ con $I_F(X, Y|S)$ dado en la ecuación 4.4.
       - Si el conjunto existe
         - Hacer $S_{XY} = S$
         - Remover el enlace $X - Y$ de G'
   - *b*) $i = i + 1$
4. Hasta $|ADJ_X| \leq i$, $\forall X$
5. Para cada tripleta de nodos (X, Y, Z) tal que X es adyacente a Y y Y es adyacente a Z pero X no es adyacente a Z orientar $X - Y - Z$ como $X \rightarrow Y \leftarrow Z$ únicamente si Y no esta en el conjunto S que ha separado X y Z en el paso 3.
6. Repetir, hasta que no se encuentren más enlaces dirigidos:
   - *a*) Dirigir todos los arcos necesarios evitando nuevas estructuras-V.
   - *b*) Dirigir todos los arcos necesarios para evitar ciclos.

Algoritmo 4.1: Pseudocódigo del algoritmo propuesto (PC-TL)

información de dominios relacionados.

Este método se probará en el capítulo 6, comparándolo con el algoritmo PC y observando su comportamiento frente a variaciones en la similitud de los dominios auxiliares participantes y a la cantidad de datos.

Capítulo 5

Aprendizaje paramétrico utilizando aprendizaje por transferencia

5.1. Introducción

Al especificar una red bayesiana, además de su estructura gráfica, debemos proporcionar un conjunto de distribuciones de probabilidad condicionadas (las tablas de probabilidad condicional), una para cada variable (efecto) condicionada a sus padres (causas) dentro de la representación gráfica de la red. Esto es, si el conjunto de padres de una variable Y viene dado por $\{X_1, ..., X_n\}$, tendremos que especificar $p(Y|X_1, ..., X_n)$. Nuevamente, para tener una buena exactitud en la recuperación de los parámetros se requiere una base de datos muy grande (relativos al modelo que se quiere aprender), por lo que, cuando no se dispone de suficientes datos es común tener estimaciones alejadas de la distribución verdadera que se quiere recuperar. El propósito del método propuesto es mejorar la estimación de las tablas de probabilidad condicional (TPC) usando funciones de combinación de probabilidades mediante aprendizaje por transferencia; es decir, utilizaremos la información de las tablas de probabilidad condicional de la redes bayesianas auxiliares para mejorar la exactitud de las TPC de la red objetivo.

5.2. Emparejamiento de las estructuras

Al intentar mejorar la estimación de las TPC de la red del dominio objetivo, se puede utilizar TPC de redes provenientes de otros dominios similares (como simplificación asumiremos que se tiene el mismo conjunto de variables para cada red). Es deseable disponer de la misma configuración de las TPC´s en la red del dominio objetivo que las provenientes de las redes de dominios similares para que la aplicación de los métodos de combinación de parámetros sea realizado de forma transparente. Como es muy posible no tener la misma configuración de variables (o lo que es lo mismo, la misma subestructura gráfica), se tienen varias circunstancias generales a considerar:

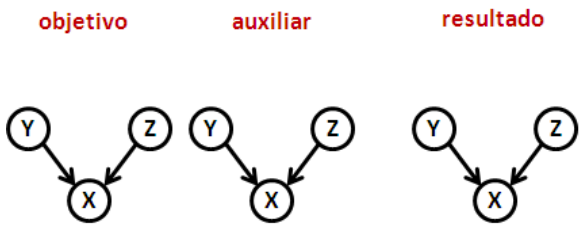
- Combinar TPC's que tienen las mismas variables y la misma subestructura de hijo y padres. Este es el caso más simple y no se requiere pasos adicionales.

- Combinar TPC´s que no comparten las mismas variables en común. En este caso podemos marginalizar[1] sobre la distribución inicial para después aplicar métodos de propagación [Pea88] por condicionamiento[2] sobre las variables desconocidas. Los pasos que componen el proceso para cada variable X contenida en la subestructura son los siguientes [DSM00, pag. 169]:

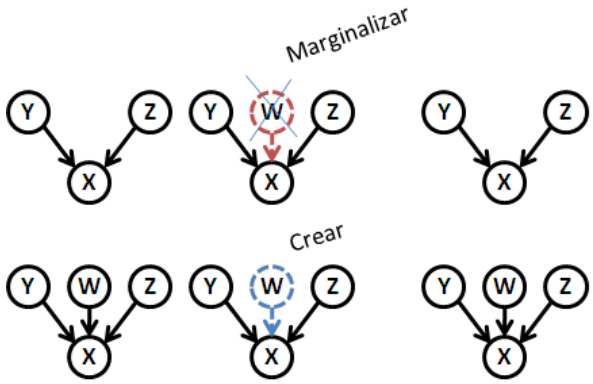
  - Obtener su conjunto de padres, $pa(X)$, en la estructura de la red objetivo.
  - Tomar cada una de las configuraciones de sus padres como evidencia y, mediante propagación, obtener $p(X|C_{pa(X)})$.

[1] Corresponde a la idea de la eliminación de nodos en el modelo gráfico. Por ejemplo, si queremos combinar $P(X|Y,Z)$ en la red objetivo con $P(X|Y,Z,W)$ de una red auxiliar, podemos obtener $P(X|Y,Z)$ de la subestructura de la red auxiliar marginalizando sobre W en la red auxiliar, de esta manera, $P(X|Y,Z) = \sum_i P(X|Y,Z,W_i)$.

[2] Es un método de inferencia para calcular las probabilidades posteriores de todas las variables no conocidas en redes multiconectadas. Funciona de la siguiente manera: si instanciamos (asignamos un valor) una variable, ésta bloquea las trayectorias de propagación de la evidencia instanciada. Entonces, asumiendo valores para un grupo seleccionado de variables podemos descomponer la estructura gráfica de la red en forma sencilla. Propagamos para cada valor posible de dichas variables y luego promediamos las probabilidades.



(a) Combinando las mismas subestructuras



(b) Combinando subestructuras diferentes

Figura 5.1: Representacion de casos para emparejamiento de las estructuras en la combinación de probabilidades condicionales.

donde $C_{pa(X)}$ representa una asignación de valores a las variables que son padres de X en la red que deseamos recuperar. En el caso de que la variable para la que se este estimando su función de probabilidad en el submodelo no tenga padres, se hará la propagación sin considerar evidencia alguna.

La figura 5.1 muestra una representacion de los casos para el emparejamiento de las subestructuras.

Al tener un conjunto de TPC´s en términos de la red objetivo involucrando el mismo conjunto de variables, podemos proceder a combinarlas con una función de combinación o agregación de probabilidades.

5.3. Agregación de probabilidades

Hay muchas funciones de agregación que han sido propuestas en la literatura [GZ86, CC96, CCT96]. Estas funciones tratan de combinar distintas distribuciones de probabilidades provistas por expertos o desde los datos. Las más comunes son:

- Combinación de opinión lineal (también conocida como media ponderada). La probabilidad de consenso asignada a la hipótesis X, $p(X)$, es una media aritmética pesada de las distintas probabilidades que cada uno de los expertos ha asignado

$$p(X) = k \sum_{i=1}^{n} w_i p_i(X)$$

  donde $P_i(X)$ representa la probabilidad condicional de la i-ésima red involucrando la variable X, w_i es el peso asociado con esta probabilidad relacionado a la confianza que se tiene al experto y k es un factor de normalización.

- Combinación de opinión logarítmica. La probabilidad de consenso asignada a X, $p(X)$, es una media geométrica pesada de todas las probabilidades que los distintos expertos han asignado

$$p(X) = k \prod_{i=1}^{m} (p_i(X))^{w_i}$$

El interés que suscita la combinación de opinión lineal y logarítmica mencionadas anteriormente radica en que pertenecen a la clase de "medias ponderadas elementales". El atractivo especial de esta clase reside en que además de ser suficientemente amplia para incluir diversas posibilidades sus propiedades matemáticas son bien conocidas.

La combinación de opinión lineal verifica las propiedades de suposición de contexto libre, conservación de cero y marginalización. Mientras que la combinación de opinión logarítmica satisface la preservación de independencia y es externamente bayesiana. Además, existe otro aspecto que hay que considerar, la influencia que tienen en las

distintas escalas que cada sujeto emplea para expresar sus grados de creencia personales. Mientras que la combinación lineal se basa en la suposición de que todas las opiniones se expresan aplicando la misma escala de probabilidad, la combinación de opinión logarítmica es invariante frente al cambio de escala de los grados de creencia individuales. Para mayor información acerca de las propiedades antes mencionadas referirse a [DSM00, pag. 102-104].

En los métodos clásicos, los pesos que miden la fiabilidad de la opinión de los expertos se emplean en las funciones de agregación. Si la distribución de fiabilidad del experto es uniforme, se aplicara un peso ω_i igual a cada probabilidad asignada por el experto.

5.4. Métodos propuestos

Los pesos asociados a las probabilidades condicionales, en esta propuesta dependen de la confianza asignada a las probabilidades en base a la cantidad de casos utilizados para estimarlas. Se proponen dos nuevos métodos de agregación:

- Combinación Lineal Basada en Distancia
- Combinación Lineal Local

Combinación Lineal Basada en Distancia (DBLP)

El primer método propuesto para agregar distribuciones de probabilidad puntuales $\{p_1, ..., p_n\}$, llamado Combinación Lineal Basada en Distancia (DBLP), va a tener en cuenta el número de fuentes que expresan valores de probabilidad más o menos próximos en consenso[3]. Involucra los siguientes pasos:

[3] En [DSM00] se trabaja en la fusión cuantitativa de redes causales, un aspecto del problema que involucra la combinación de diversas fuentes de conocimiento (expertos) relativo al mismo problema. En este trabajo se ha tomado el procedimiento base de combinación de probabilidades (presentado en [DSM00, pag. 125]) adaptándolo para involucrar distintos dominios relacionados.

1. Calcular el punto medio de todas las distribuciones de probabilidad aplicando descuento por fiabilidad a los datos:

$$\overline{p} = k \sum_{i=1}^{n} (f_i \times p_i) \tag{5.1}$$

donde f_i nos arroja un valor de fiabilidad sobre las TPC´s y k es un factor de normalización.

$$f_i = \begin{cases} 1 - \frac{log(c_f)}{c_f} & \text{si } c_f \geq 3 \\ 1 - \frac{c_f \times log(3)}{3} & \text{si } c_f < 3 \end{cases} \tag{5.2}$$

donde $c_f = \frac{N}{T \times 2}$ representa la cantidad de error que se puede esperar desde las TPC´s, que depende de T, el tamaño de la TPC (número de posibles valores en los estados de las variables) y de N, el número de casos. La función f_i al no tener continuidad en el intervalo de valores posibles, presenta dos valores diferentes para dicha función[4].

2. Obtener las distancias mínimas (d_{min}) y máximas (d_{max}) entre la probabilidad del conjunto de datos objetivo y el anterior promedio, $\overline{p}$, mediante $d_i = d(p_i, p), i = 1, ..., n$. Encontrar

$$d_{min} = min\{d_i, i = 1, ..., n\}$$

$$d_{max} = max\{d_i, i = 1, ..., n\}$$

las distancias mínima y la máxima, d_{min}, d_{max} respectivamente.

3. Evaluar la nueva probabilidad condicional de la red objetivo como sigue:

$$p_{target} = (1 - c_i) p_{target} + c_i \overline{p} \tag{5.3}$$

donde el coeficiente de agregación, c_i, expresa que tanto considerar las TPC´s de las otras redes. El coeficiente c_i básicamente expresa lo siguiente: si las TPC´s de la red objetivo son similares al promedio de todas las TPC´s, enton-

[4]Como la función f_i presentada solo es continua en el intervalo $c_f \geq 3$, el primer componente de la función, cuando $c_f \geq 3$, esta dado en una escala logarítmica, y el segundo componente, cuando $c_f < 3$, se ha interpolando linealmente en el intervalo $[0, 3]$.

ces se dará más peso al promedio, o de otra forma se dará más peso a la TPC objetivo, es decir, los parámetros se mueven en el sentido de las opiniones más próximas en consenso. Esto es expresado como sigue:

$$c_i = (d_i - d_{min}) \times \left(\frac{c_{max} - c_{min}}{d_{max} - d_{min}} \right) + c_{min} \tag{5.4}$$

donde c_{max} y c_{min} son parámetros que indican que grado de influencia queremos considerar de las otras TPC´s. En nuestro caso usamos $c_{max} = 0{,}75$ y $c_{min} = 0{,}25$.

EJEMPLO 5.1 Apliquemos el método de agregación, tomando como parámetro de agregación $c_{min} = 0{,}25$ y $c_{max} = 0{,}75$, utilizando una fiabilidad uniforme $f_i = 1$ para propósitos de nuestro ejemplo para las distribuciones de probabilidad puntual:

$p_1 = (0{,}2)$ $p_2 = (0{,}7)$ $p_3 = (0{,}4)$ $p_4 = (0{,}1)$

Tomamos como la probabilidad de interés $p_{interes} = p_1$, y calculamos el punto medio de estas distribuciones como $\overline{p} = \frac{1}{4}((1 \times 0{,}2) + (1 \times 0{,}7) + (1 \times 0{,}4) + (1 \times 0{,}1)) = 0{,}35$. Las distintas distancias serán: $d_1 = \overline{p} - 0{,}2 = 0{,}15$ $d_2 = \overline{p} - 0{,}7 = 0{,}35$ $d_3 = \overline{p} - 0{,}4 = 0{,}05$ y $d_4 = \overline{p} - 0{,}1 = 0{,}25$, y por tanto tomaremos como distancia mínima $d_{min} = d_3$ y como distancia máxima $d_{max} = d_2$. Ahora como solo estamos interesados en $p_{interes}$ obtenemos el parámetro $c_i = (0{,}2 - 0{,}05) \times \left(\frac{0{,}75 - 0{,}25}{0{,}35 - 0{,}05} \right) + 0{,}25 = 0{,}5$, cuyo valor está comprendido en el intervalo $[c_{min}, c_{max}]$. La probabilidad de p_1 una vez calculada es $p_{interes} = (1 - 0{,}5) \times 0{,}2 + 0{,}5 \times 0{,}35 = 0{,}275$.

El uso de los parámetros posibilita que los valores de las tablas de probabilidad condicional objetivo se muevan en el sentido de las probabilidades más próximas y se aleje de las más lejanas. Es decir, será mayor el peso de las probabilidades más cercanas a coincidir y menor el de las lejanas. Ahora el comportamiento del método dependerá de los valores que tomen c_{min} y c_{max}.

Combinación Lineal Local (LoLP)

La idea de la segunda función de agregación, llamada Combinación Lineal Local (LoLP), es usar únicamente las probabilidades más cercanas o las probabilidades locales y pesarlas de acuerdo a la cantidad de datos que se utilizaron para calcular las tablas de probabilidad condicional. Esta estrategia tiene los siguientes pasos:

1. Obtener el promedio de las probabilidades para los conjuntos de datos *auxiliares*, pero únicamente entre las probabilidades más cercanas, en nuestro caso, aquellas que están entre la diferencia entre la probabilidad objetivo y el promedio de las probabilidades restantes $\bar{p}$, es decir:

$$\overline{p}_{local} = \frac{1}{n}\sum_{i=1}^{n} p_i \;\; \forall p_i \;\; \text{tal que } p_i \in \{p_{target} \pm (p_{target} - \overline{p})\} \tag{5.5}$$

2. Obtener la nueva probabilidad condicional de la red objetivo como sigue:

$$p_{target} = f_{target} \times p_{target} + (1 - f_{target}) \times \overline{p}_{local} \tag{5.6}$$

   donde f_{target} arroja un valor de fiabilidad sobre las TPC's de interés basado en la ecuación (5.2).

El cálculo de un nuevo punto medio, en el paso 1 del método, nos restringe en transferir información sobre los valores de los parámetros cercanos y omitir aquellos que se encuentran a una distancia más lejana de la distribución de la probabilidad de interés.

EJEMPLO 5.2 Apliquemos el método de agregación utilizando una fiabilidad uniforme $f_i = 0{,}5$ a las distribuciones de probabilidad puntual del ejemplo 5.1. Tomamos como la probabilidad de interés $p_{interes} = p_1$, y calculamos el punto medio de estas distribuciones como $\overline{p} = \frac{1}{4}((1\times0{,}2)+(1\times0{,}7)+(1\times0{,}4)+(1\times0{,}1)) = 0{,}35$. Las distintas distancias serán: $d_1 = 0{,}15$, $d_2 = 0{,}35$, $d_3 = 0{,}05$ y $d_4 = 0{,}25$. Solo aquellas que se encuentren entre las diferencia entre la probabilidad objetivo y el promedio de las probabilidades restantes son usadas (en nuestro

ejemplo solo son utilizados aquellos cuyo valor se encuentre en el intervalo $0{,}2 \pm (0{,}2 - 0{,}35) \implies 0{,}2 \pm -0{,}15 \implies [0{,}05 - 0{,}35]$), d_2 y d_4 ya no se usa en los cálculos por estar alejados de este intervalo por lo que el nuevo punto medio $\bar{p}_{local} = \frac{1}{2}((1 \times 0{,}2) + (1 \times 0{,}4)) = 0{,}3$. La probabilidad de p_1 una vez transformado es $p_{interes} = 0{,}5 \times 0{,}2 + (1 - 0{,}5) \times 0{,}3 = 0{,}25$.

5.5. Conclusiones

Aun cuando existe una gran cantidad de trabajos relacionados al proceso de combinar opiniones procedentes de expertos [GZ86], no es mucho el trabajo relativo a la mejora de los parámetros numéricos a partir de bases de datos similares utilizando aprendizaje por transferencia. En este capítulo se presentaron dos métodos de combinación de probabilidades a partir de datos, los cuales obtienen conocimiento adicional de tareas similares con la finalidad de mejorar la estimación de las tablas de probabilidad condicional.

Ambos métodos están basados en el método de combinación de decisión lineal (ver sección 5.3) por ser uno de los más sencillos y rápidos. Sin embargo, el método de combinación de decisión lineal realiza una media ponderada sobre todas las fuentes a combinar, pesadas por una medida de creencia de cada fuente según un valor asignado *a priori* (generalmente dada por un experto), no considerando que las fuentes provienen de bases de datos de diversos tamaños, y por lo tanto de diversa fiabilidad con respecto a los valores presentados. Para tratar estos nuevos factores, los métodos propuestos tienen la característica de incorporar una medida de la fiabilidad (ecuación 5.2) de los datos presentes en las tablas de probabilidad condicional al momento de realizar la combinación, después del emparejamiento de las estructuras a combinar. De esta forma se logra tener cierta resistencia a los cambios de las TPC de la red objetivo cuando se poseen suficientes datos, y a la vez permitiendo cambios utilizando las tablas auxiliares que conforman las bases de datos similares cuando la cantidad de datos por TPC decrece.

El primer método, llamado Combinación Lineal Basada en Distancia (DBLP), considera

la fiabilidad de las probabilidades de todas las TPC´s participantes de todas las redes bayesianas, y modificando la TPC de interés *moviéndose* en el sentido de las opiniones más próximas entre ellas, y alejándose de las más lejanas a la misma. Mientras el segundo método, llamado Combinación Lineal Local (LoLP), sólo descarta aquellos valores de probabilidad que se encuentran lejanos a una determinada distancia (dada por la ecuación 5.5) de la medida de probabilidad dada por la TPC de interés y los combina con el promedio de las probabilidades restantes, no sin antes calcular la proporción de la combinación dada por el factor de fiabilidad f_{target} aplicado a la TPC de interés (calculado en el paso 2 del método LoLP descrito en la sección 19).

En el siguiente capítulo se presentarán los resultados experimentales realizados a los métodos de aprendizaje estructural y paramétrico propuestos.

Capítulo 6

Experimentos y Resultados

6.1. Metodología de experimentación

El propósito principal de nuestros experimentos es confirmar los beneficios del método propuesto: la utilización de información proveniente de tareas similares puede ayudar a mejorar modelos aprendidos con relativamente muy pocos datos. Para comparar la red bayesiana original con la construida a través de los métodos propuestos debemos evaluar la calidad de la aproximación:

- Cualitativamente, contabilizando las aristas que se pierden, añaden o invierten con respecto al modelo original.
- Cuantitativamente, midiendo la diferencia entre las distribuciones de probabilidad codificadas en ambas redes aplicando la medida de diferencia del error cuadrático medio.

En esta sección vamos a detallar los experimentos realizados para evaluar los distintos métodos de aprendizaje estructural y paramétrico propuestos. Las pruebas realizadas se llevaron a cabo con varias redes bayesianas comúnmente utilizados para evaluar los algoritmos de aprendizaje, obtenidos de Internet, específicamente en la pagina del

Proyecto Elvira [Elv02]. Cabe mencionar que los métodos propuestos fueron implementados en ésta plataforma de desarrollo, y el método propuesto de aprendizaje estructural se codificó utilizando como base la implementación del algoritmo PC, utilizado como comparación en los experimentos. En nuestros experimentos, usamos tres redes bayesianas cuya breve descripción y características (número de nodos y enlaces) se encuentra en la tabla 6.1.

Nombre	Descripción	n	α
Alarm	Una red creada en el campo de diagnóstico médico para el monitoreo de pacientes en cuidados intensivos	37	46
Boblo	Un sistema de ayuda para verificación del ganado a través de la identificación del tipo de sangre	23	24
Insurance	Una red para clasificación en aplicaciones de seguros de automóviles.	27	52

Tabla 6.1: Descripción de las redes usadas en los experimentos, se muestra el número de nodos (n) y de aristas (α) para la estructura original.

Para evaluar los distintos métodos de aprendizaje estructural y paramétrico de redes bayesianas adoptamos parte del procedimiento de evaluación propuesto en [NMC07]. Creamos nuevos problemas relacionados a partir de las redes objetivo descritas en la tabla 6.1, cambiando la red original añadiendo/borrando enlaces y alterando las TPC aplicando ruido gaussiano con media 1 y diferentes valores de desviación estándar. Estas redes alteradas serán usadas como redes auxiliares para intentar reproducir la red original utilizando sólo un subconjunto de los datos originales.

Los conjuntos de datos usados en nuestros experimentos son muestreados (mediante muestreo lógico probabilístico como está implementado en la plataforma Elvira) desde las redes bayesianas objetivo y auxiliares creadas a partir de las redes construidas en el apartado anterior. De cada una de las redes se generaron conjuntos de datos de diferentes tamaños (generándose conjuntos de 25, 50, 100, 200, 400, 500, 800, 1000, 2000, 4000 y 8000 instancias) y corrimos los diferentes procedimientos de aprendizaje

estructural y paramétricos tomando algunos de los conjuntos generados de acuerdo a los distintos experimentos realizados. Los procedimientos no tienen acceso a las redes originales, sólo tienen acceso a los conjuntos de datos muestreados de las redes objetivo y similares. La meta es recuperar tan cercanamente como sea posible la estructura y los parámetros numéricos de la red objetivo. Para incrementar la exactitud del análisis estadístico, se repitió cada experimento 10 veces (con muestreos independientes para cada conjunto de datos).

Se utilizó una medida para comparar la diferencia estructural de las redes bayesianas generadas por los métodos de aprendizaje y una medida para determinar la diferencia entre la distribución de probabilidades de las mismas, las cuales son la distancia de edición y el error cuadrático medio, respectivamente.

DISTANCIA DE EDICIÓN La principal idea de la distancia de edición es definir la similitud de un grafo por medio de la cantidad de operaciones, reflejando los pequeños cambios estructurales necesarios para transformar un grafo en otro. Para nuestras pruebas, medimos la distancia de edición como la cantidad de cambios necesarios medido en el número de arcos añadidos y eliminados, necesarios para llegar a la verdadera estructura. La cantidad de arcos revertidos no se contabilizó porque no se considera importante la causalidad de las redes en los métodos analizados.

ERROR CUADRÁTICO MEDIO Esta medida de distorsión se define como:

$$MSE = \frac{1}{N}\sum_{i=1}^{N}\left(p(x_i) - p(x_i')\right)^2$$

donde $p(x_i)|$ es la probabilidad estimada
$p(x_i')$ es la probabilidad verdadera
N es el número de probabilidades analizadas

y la usamos para cuantificar la diferencia en las distribuciones de probabilidades codificadas en las tablas de probabilidad condicional entre la verdadera distribución y las obtenidas con los procedimientos de aprendizaje paramétrico presentados en esta tesis. Así, para medir la diferencia en la distribución de probabilidad

entre ambas redes analizadas, calculamos la medida de error y la redefiniremos como sigue

$$MSE' = \frac{1}{N} \sum_{X_i \subset RedObj} \sum_{x_i \subset X_i} \sum_{pa(x_i) \subset X_i} \left(p(x_i|pa(x_i)) - p(x'_i|pa(x'_i))\right)^2$$

donde x_i es un estado de la variable $X_i = x_i$, $pa(x_i)$ los estados de las variables cuando $pa(X_i) = pa(x_i)$, $p(x_i|pa(x_i))$ es la probabilidad cuando los estados de las variables son $X_i = x_i$ y $pa(X_i) = pa(x_i)$, siendo $pa(x_i) \subset X_i$ el conjunto de estados de los padres de la variable X_i, $x_i \subset X_i$ el conjunto de estados de la variable X_i, $X_i \subset RedObj$ el conjunto de variables que pertenecen a la red objetivo y N el número total de probabilidades involucrados en el cálculo.

6.2. Experimentos

Se realizaron dos tipos principales de experimentos para mostrar el comportamiento en los algoritmos de aprendizaje estructural y paramétrico cuando información de problemas relacionados es tomada en cuenta. El primer tipo de experimento es *aplicado a los métodos de aprendizaje estructural y paramétrico*. Está enfocado en estudiar el comportamiento del método propuesto cuando existen variaciones en la cantidad de instancias utilizadas en el dominio de interés y los dominios relacionados, utilizados como dominios auxiliares para el aprendizaje de la red objetivo. El segundo tipo de experimento estudia *solamente el comportamiento del método de aprendizaje estructural* cuando la relación de similitud entre el dominio objetivo y los dominios auxiliares varía gradualmente.

A continuación se presentan los detalles de los experimentos y posteriormente se presentarán los resultados para cada uno de los 3 modelos de redes bayesianas utilizados.

EXPERIMENTO TIPO I: VARIANDO EL NÚMERO DE EJEMPLOS. Para este experimento variamos la cantidad de ejemplos utilizados tanto en el dominio objetivo como en los dominios auxiliares. Para esto creamos las redes auxiliares aplicando el siguiente procedimiento:

		Arcos añadidos	Arcos eliminados	Arcos revertidos	Distancia de Edición	Ruido aplicado a las TPC (%)	Error Cuadrático Medio (MSE)
Alarm	Red Mas Similar	2	1	0	3	5%	0.019
	Red Menos Similar	6	8	0	14	20%	0.041

Tabla 6.2: Características de las redes auxiliares utilizadas en los experimentos tipo I para la red Alarm.

- Una red similar agregando arcos aleatoriamente en un 5 % del número de enlaces presentes seguido por la eliminación de 5 % de los enlaces iniciales e introduciendo 5 % de ruido en las TPC´s multiplicándolas por un ruido aleatorio (recuperando previamente los parámetros a través de un conjunto de datos con $100,000$ casos muestreados de la red original).

- Una red menos similar agregando arcos aleatoriamente en un 20 % del número de enlaces presentes seguido por la eliminación de 20 % de los enlaces iniciales e introduciendo 20 % de ruido en las TPC´s multiplicándolas por un ruido aleatorio (recuperando previamente los parámetros a través de un conjunto de datos con $100,000$ casos muestreados de la red original).

La tabla 6.2 muestra, entre otros datos, las similitudes basadas en el número de operaciones elementales (añadir, borraron invertir la dirección de un arco) para llegar a la estructura original de la que fueron basados. Estas redes modificadas servirán como los dominios auxiliares necesarios para los experimentos.

Primero fijamos la cantidad de instancias de entrenamiento de los dominios auxiliares a $1,000$ casos y fuimos variando la cantidad de ejemplos para el dominio objetivo desde 25 casos hasta $2,000$ casos. Esto nos permitirá observar el comportamiento cuando la cantidad de instancias en el dominio objetivo tiende a la cantidad de instancias en los dominios auxiliares, e incluso siendo mayor seguir al comportamiento del algoritmo PC. En este caso se hace innecesario el aprendizaje por transferencia y el algoritmo propuesto ya no toma ventaja de las relaciones de similitud presentes debido a que la confianza en su propia estructura es tal, que ya no necesita el apoyo de otros dominios. Posteriormente invertimos la relación, y mantenemos fija la cantidad de instancias de entrenamiento para el dominio objetivo en 100 casos y variamos la cantidad

de ejemplos para los dominios auxiliares desde 500 casos hasta $8,000$ casos. Con esto queremos observar que el aprendizaje por transferencia desde dominios auxiliares ayuda a mejorar la estructura de la red objetivo en la medida que se tiene una cantidad mayor de instancias en los dominios auxiliares. El conjunto de datos original es tomado como la red objetivo. Como base de comparación para los métodos de aprendizaje estructural construimos una red bayesiana con un subconjunto de datos muestreado de la red original usando el algoritmo PC como está implementado en Elvira [Elv02] con un valor de 90 % de significancia. Para el método propuesto se utiliza un subconjunto generado de la red original y algunos subconjuntos de datos muestreados de las redes auxiliares.

Siguiendo al procedimiento anterior, se medirá la diferencia en las distribuciones de probabilidad obtenidas por los algoritmos de aprendizaje paramétrico sin utilizar y utilizando información de problemas relacionados (utilizando el método propuesto de aprendizaje estructural y utilizando como base de comparación el método de aprendizaje presentado en la sección 2.2 utilizando suavizado de Laplace y el método de combinación lineal presentado en la sección 5.3).

Para todos los casos se estudia el efecto de utilizar redes auxiliares similares y menos similares en el aprendizaje.

EXPERIMENTO TIPO II: VARIANDO LA SIMILITUD ENTRE LOS DOMINIOS PARTICIPANTES. En los experimentos anteriores queremos mostrar que el algoritmo propuesto mejora al método de aprendizaje estructural utilizando como método base el algoritmo PC, cuando la similitud de los dominios es clara. En este segundo experimento mostraremos la capacidad del método propuesto de transferir aprendizaje cuando los dominios auxiliares no son tan relacionados al dominio objetivo. También mostraremos la importancia de utilizar varios dominios cuando no se conoce *a priori* el grado de similitud de los dominios utilizados como auxiliares. Para este experimento creamos dos conjuntos diferentes de redes auxiliares; para el primer conjunto creamos seis redes relacionadas agregando arcos aleatoriamente un $10\,\% - 60\,\%$ del número de enlaces presentes, seguido por la eliminación de $10\,\% - 60\,\%$ de los enlaces iniciales, utilizando la estructura de la red (recuperando los parámetros a través de un conjunto de datos con $100,000$ casos muestreados de la red original). Para el segundo conjunto creamos nueve redes

		Arcos añadidos	Arcos eliminados	Arcos revertidos	Distancia de Edición	Error Cuadrático Medio (MSE)
Red Alarm	Red Similar 1	4	5	0	9	0.018
	Red Similar 2	9	11	0	20	0.033
	Red Similar 3	8	14	0	22	0.027
	Red Similar 4	10	22	0	32	0.047
	Red Similar 5	11	33	0	44	0.055
	Red Similar 6	12	52	0	64	0.032

Tabla 6.3: Características del conjunto 1 de redes utilizadas en los experimentos tipo II para la red Alarm.

		Arcos añadidos	Arcos eliminados	Arcos revertidos	Distancia de Edición	Error Cuadrático Medio (MSE)
Red Alarm	Red Similar 1	3	0	0	3	0.018
	Red Similar 2	10	0	0	10	0.044
	Red Similar 3	13	0	0	13	0.050
	Red Similar 4	16	0	0	16	0.043
	Red Similar 5	24	0	0	24	0.079
	Red Similar 6	29	0	0	29	0.088
	Red Similar 7	32	0	0	32	0.094
	Red Similar 8	40	0	0	40	0.094
	Red Similar 9	43	0	0	43	0.100

Tabla 6.4: Características del conjunto 2 de redes utilizadas en los experimentos tipo II para la red Alarm.

relacionadas, iniciando de la verdadera estructura de la red eliminando arcos aleatoriamente un $10\,\% - 90\,\%$ de los arcos iniciales. El propósito del primer conjunto de redes similares es observar el comportamiento del algoritmo propuesto frente a variaciones en la similitud de los dominios relacionados y observar cuando el aprendizaje por transferencia comienza a dañar el desempeño en la recuperación de la estructura. Mientras que para el segundo conjunto de pruebas, el propósito es observar el comportamiento cuando, difieren en la cantidad de independencias presentes en el dominio objetivo (es decir, varía la cantidad las aristas presentes entre las variables de la red bayesiana).

Las características de los conjuntos de redes bayesianas utilizadas se encuentran condensadas en las tablas 6.3 y 6.4.

Ya que describimos los conjuntos de experimentos que realizaremos, procederemos a

mostrar los resultados obtenidos en las tres redes bayesianas utilizadas como referencia en esta tesis.

6.3. Resultados

Por propósitos de presentación sólo se muestran los resultados para la red Alarm en la presente sección. Para las dos redes siguientes, Boblo e Insurance, los resultados se presentan en el apéndice A en esta tesis.

Red Alarm

La red Alarm (A Logical Alarm Reduction Mechanism) [BSCC89], que se muestra en la Figura 6.1, es una aplicación de las redes bayesianas en el campo del diagnóstico médico, en concreto a la detección de causas de alarma en un sistema de monitorización de un paciente.

EXPERIMENTO TIPO I

La figura 6.2 muestra el comportamiento del algoritmo PC elegido como base comparado CON EL ALGORITMO DE APRENDIZAJE ESTRUCTURAL PROPUESTO (PC-TL), donde se muestra el comportamiento utilizando datos auxiliares muestreados de la tabla 6.2. El objetivo es caracterizar el comportamiento del método a medida que aumenta la cantidad de instancias de entrenamiento utilizadas para la construcción de la red objetivo y manteniendo fija la cantidad de instancias para los dominios auxiliares utilizados por el algoritmo propuesto PC-TL, fijándose en $1,000$ instancias. La figura muestra el comportamiento del método propuesto cuando se incrementa gradualmente la cantidad de instancias de aprendizaje de los datos del dominio auxiliar pero manteniendo fija la cantidad de instancias para la red objetivo. La figura muestra el comportamiento del algoritmo propuesto (PC-TL) manteniendo fija la cantidad de instancias para la red objetivo (en 100 casos) mientras se varía la cantidad de datos para los dominios auxiliares. De forma similar al primer tipo de experimento, utiliza datos de un dominio muy

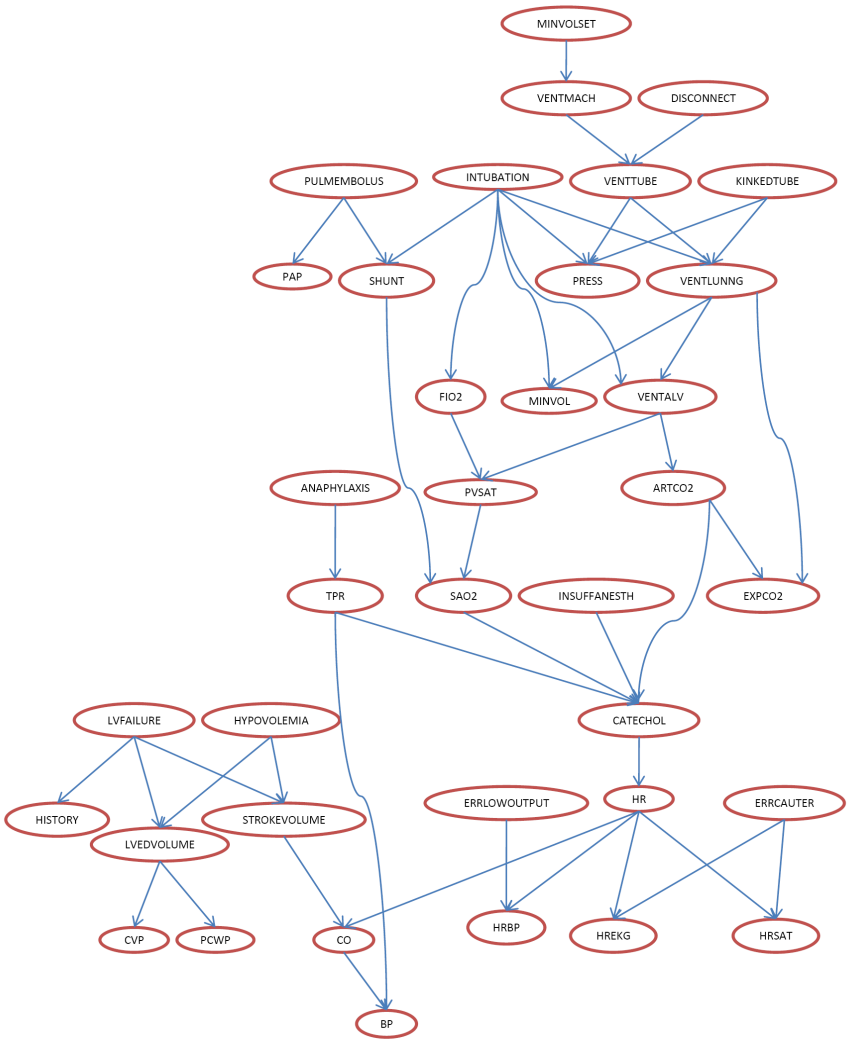


Figura 6.1: Estructura original de la red Alarm.

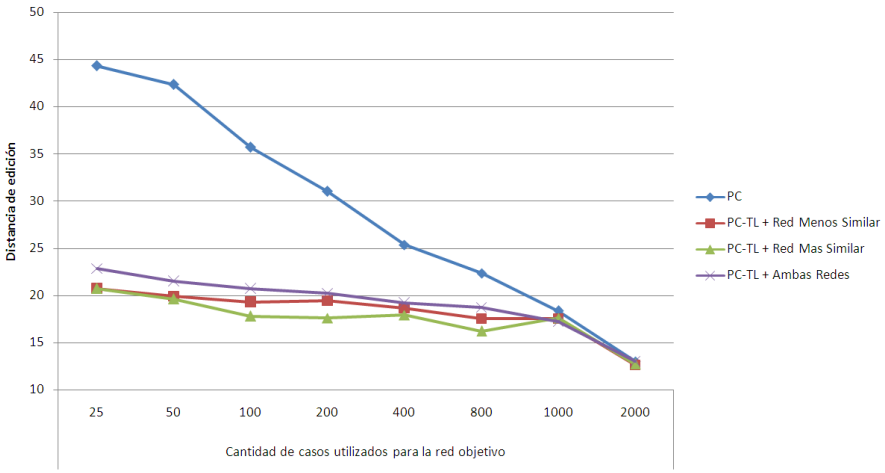


Figura 6.2: Comparación entre el algoritmo PC y PC-TL considerando diferente número de casos y utilizando dos conjuntos de datos auxiliares. La figura muestra el comportamiento medido como la reducción en la distancia de edición mientras se incrementa el número de casos para la red objetivo mientras se mantiene fijo para las redes auxiliares en la red Alarm.

similar, datos de un dominio menos similar y la combinación de los anteriores (cuyas características se muestran en la tabla 6.2).

De la figura 6.2 puede verse el comportamiento esperado. El método propuesto se desempeña mejor que el algoritmo PC, especialmente cuando se tiene un número relativamente pequeño de casos, y ambos tienden a converger al mismo resultado cuando se incrementa el tamaño del conjunto de entrenamiento de la red objetivo. No parece hacer mucha diferencia utilizar conjuntos de datos más o menos similares, aunque se requieren pruebas adicionales para observar cuando el desempeño del algoritmo se degrada conforme se utilizan conjuntos de datos más diferentes al objetivo. Más aún, como se muestra en la figura 6.3, al utilizar varios conjuntos de datos de dominios auxiliares, la recuperación de la estructura parece mejorar conforme aumenta la cantidad

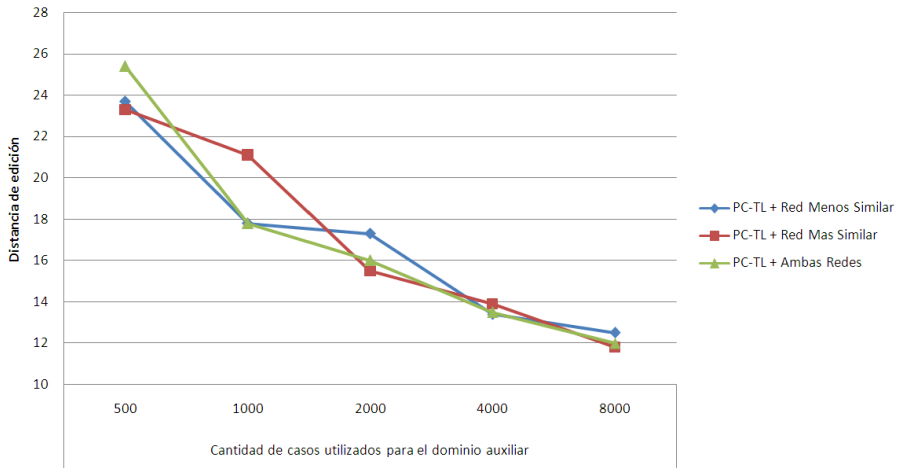


Figura 6.3: Comparación entre el algoritmo PC y PC-TL considerando diferente número de casos y utilizando dos conjuntos de datos auxiliares. La figura muestra el comportamiento medido como la reducción en la distancia de edición mientras se incrementa el número de casos para las redes auxiliares mientras se mantiene fijo para la red objetivo en la red Alarm.

de instancias de entrenamiento que cuando se utiliza un solo dominio auxiliar. También se observa que se pueden obtener mejores resultados cuando se utilizan varios dominios similares como se puede notar.

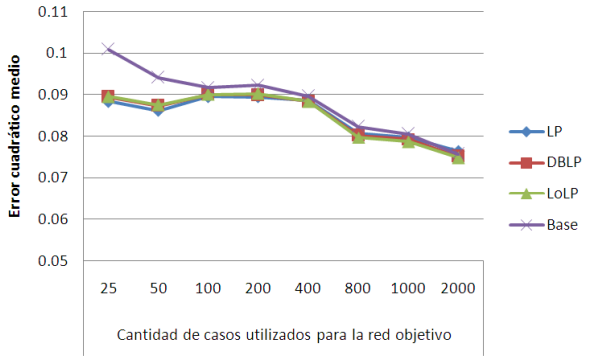
Similarmente, se muestran los RESULTADOS para los ALGORITMOS DE APRENDIZAJE PARAMÉTRICO. Se requiere anticipar que los resultados de los métodos de aprendizaje paramétrico se limitaran al experimento tipo I ya que no parece necesario continuar con los experimentos restantes ya que el único indicador de importancia es la similitud en las distribuciones de las probabilidades de los dominios objetivos y auxiliares. La figura 6.4 muestra una comparación en la recuperación de los parámetros medido en términos del error cuadrático medio a partir de la verdadera distribución de los parámetros y las TPC's de las probabilidades condicionales construidas únicamente con los datos de las redes objetivo (Base), el algoritmo de agregación lineal (LP), y los dos algoritmos propuestos (DBLP y LoLP). Las figuras muestran el comportamiento de los métodos de aprendizaje paramétrico cuando se incrementa el número de datos para la red objetivo utilizando la medida del error cuadrático medio. En la figura 6.4(a) muestra el comportamiento cuando los parámetros de la red auxiliar son muy similares a los pa-

rámetros de la red objetivo buscada, en 6.4(b) se muestra los resultados con una red auxiliar menos similar mientras que en 6.4(c) se muestra utilizando los dos dominios auxiliares anteriores. El número de ejemplos para los conjuntos de datos auxiliares se mantuvo fijo a $1,000$ instancias. Mientras en las figuras 6.5(a,b,c) se muestra el comportamiento cuando se incrementa el número de instancias de entrenamiento para los conjuntos de datos auxiliares utilizando redes auxiliares similares, menos similares y ambas redes de forma similar a las figuras 6.4. El número de instancias utilizadas para la recuperación de la red objetivo se mantuvo fija a 100 ejemplos.

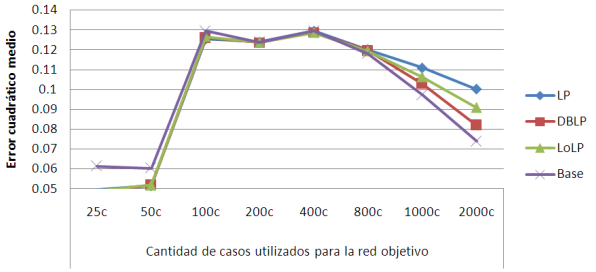
De la figura 6.4 podemos ver que en general el método propuesto se desempeña de forma similar al método base, todos ellos tienen una tendencia a disminuir la medida de error presentada conforme aumenta la cantidad de instancias usadas para el entrenamiento (y manteniendo fija en $1,000$ instancias los datos de dominios auxiliares). Sin embargo, como puede observarse en la figura 6.5, cuando se aumenta la cantidad de instancias utilizadas para los dominios auxiliares, se mejora la precisión en la recuperación comparándolo con el método base, por lo que podemos pensar que utilizar conjuntos de datos adicionales puede mejorar la recuperación de parámetros de las redes bayesianas. La alta variabilidad de las curvas (figuras 6.4 y 6.5) dentro de la recuperación de los parámetros nos hace pensar que como se utilizan estructuras diferentes en cada punto de la gráfica, los resultados se presentan de forma errática. Por lo anterior, para tener una buena perspectiva de comparación se utilizan como línea base, los resultados generados por el método básico de aprendizaje estándar (ver sección 2.2), que es realizado por simple conteo sobre los datos disponibles.

EXPERIMENTO TIPO II

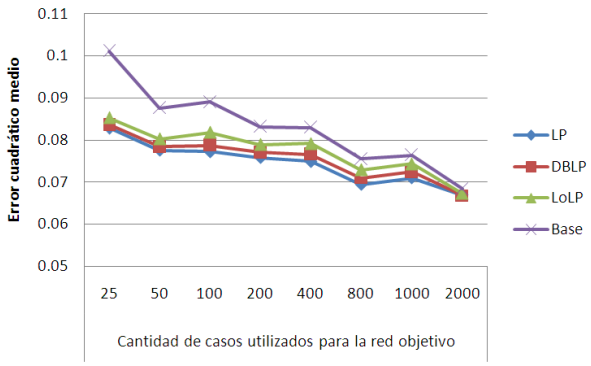
Este tipo de experimentos muestran la capacidad del algoritmo propuesto para transferir correctamente aprendizaje. La figura 6.6 muestra el comportamiento del método propuesto en la recuperación de la verdadera estructura definida mediante la distancia



(a) Red auxiliar muy similar

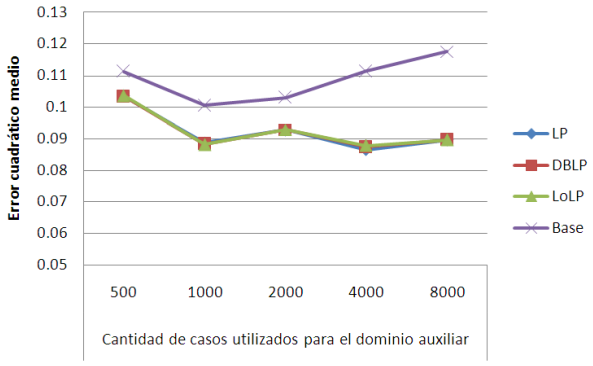


(b) Red auxiliar menos similar

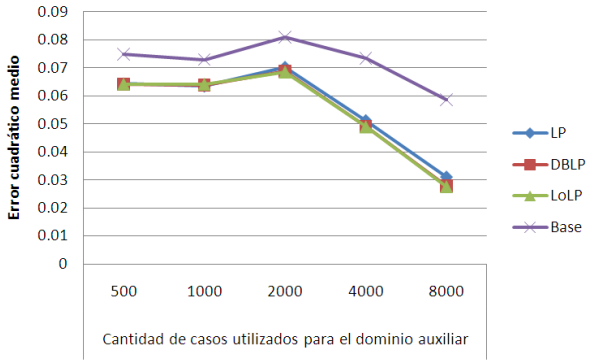


(c) Ambas redes auxiliares (similar y menos similar)

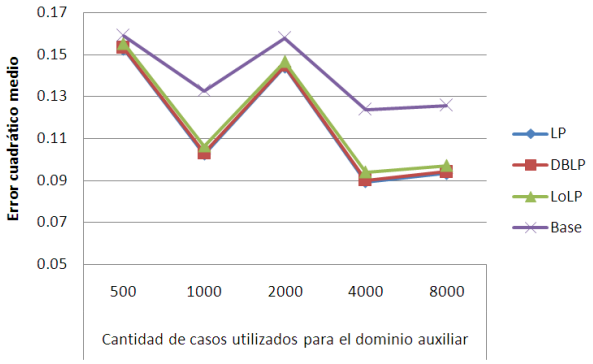
Figura 6.4: Comparación entre los métodos de aprendizaje paramétrico cuando se incrementa el número de casos para la red objetivo en la red Alarm. La figura a la izquierda muestra la reducción en el error utilizando una red auxiliar muy similar, la figura central utilizando una red menos similar mientras que en la figura a la derecha se muestra utilizando las dos redes auxiliares previas.



(a) Red auxiliar muy similar



(b) Red auxiliar menos similar



(c) Ambas redes auxiliares (similar y menos similar)

Figura 6.5: Comparación entre los métodos de aprendizaje paramétrico cuando se incrementa el número de casos para las redes auxiliares en la red Alarm. La la izquierda figura muestra la reducción en el error utilizando una red auxiliar muy similar, la figura central utilizando una red menos similar mientras que la figura a la derecha se muestra utilizando las dos redes auxiliares previas.

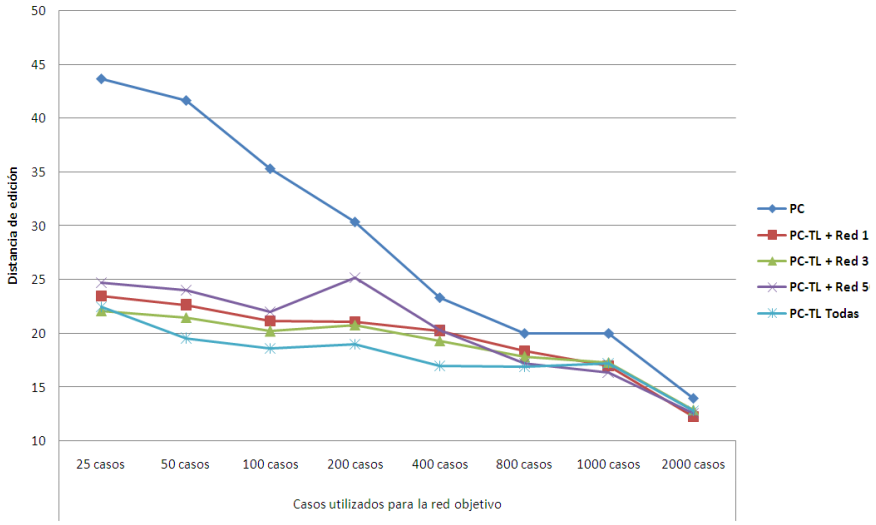


Figura 6.6: Comportamiento del método propuesto (PC-TL) cuando se utilizan redes auxiliares cuyas características de similitud con la red objetivo varían, para la red Alarm. El eje X muestra la variación en el tamaño de la red objetivo, variando de $25 - 2000$ casos. El eje Y muestra el error medido en la distancia de edición con la verdadera estructura sobre cada prueba. La cantidad de casos utilizados para los dominios objetivos ha sido fijada a 1000 casos.

de edición ante variaciones en la similitud de las redes auxiliares, y variaciones en la cantidad de ejemplos utilizados en la recuperación de la red objetivo de $25 - 2,000$ ejemplos, siempre manteniendo la cantidad de instancias para la red auxiliar a $1,000$ ejemplos. Mientras en la figura 6.7 se muestra el comportamiento ante variaciones en los dominios auxiliares y la cantidad de ejemplos usados en los dominios auxiliares (variando desde $500 - 8,000$ ejemplos), manteniendo fija la cantidad de ejemplos usados en el aprendizaje del dominio objetivo (utilizando 100 instancias). El eje X muestra la variación en la cantidad de ejemplos utilizados para generar los dominios auxiliares mientras se varía la similitud con el dominio objetivo desde un $10\,\% - 60\,\%$, mostrándose en el eje Z, como se presenta en la tabla 6.3, y el eje Y muestra la cantidad del error encontrado en cada una de las pruebas.

Ambas pruebas utilizaron alguna de las redes auxiliares presentadas en la tabla 6.3.

El último experimento presentado es cuando las redes auxiliares sólo difieren de la red

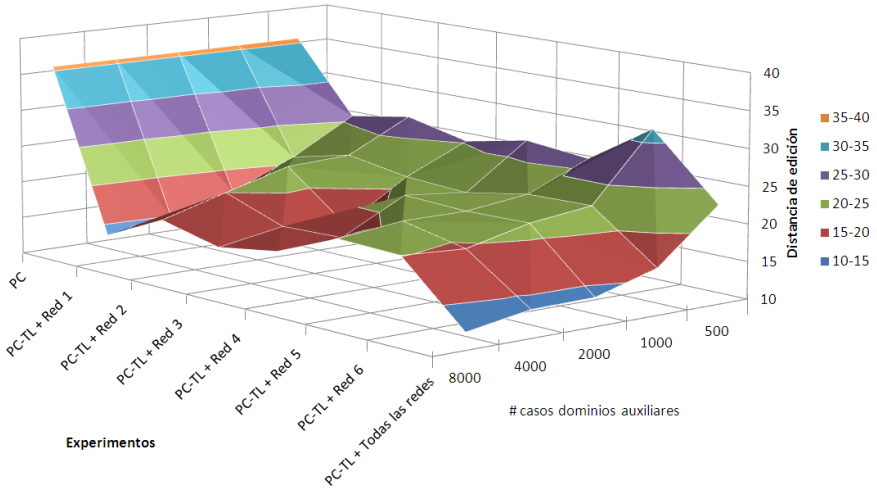


Figura 6.7: Comportamiento del método propuesto (PC-TL) cuando se utilizan redes auxiliares cuyas características de similitud con la red objetivo varían mostrándose en el eje Z (experimentos), para la red Alarm. El eje X muestra la variación en el tamaño de la red auxiliar variando de $500 - 8000$ casos. El eje Y muestra el error medido en la distancia de edición con la verdadera estructura sobre cada prueba.

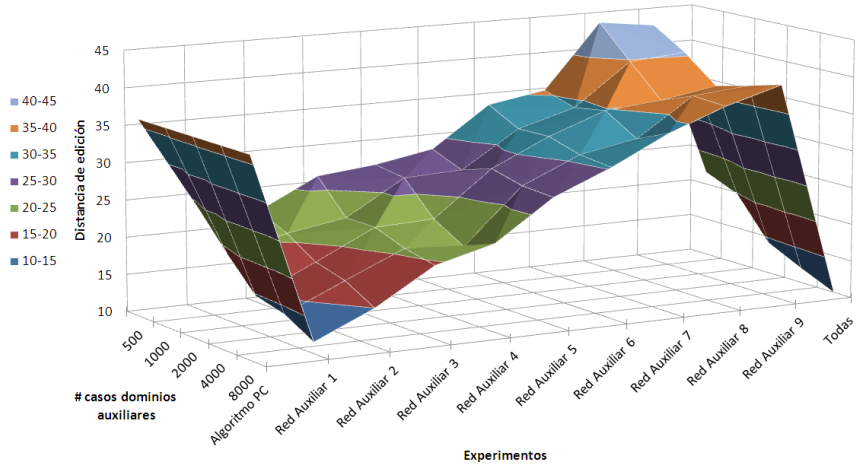


Figura 6.8: Comportamiento del método propuesto (PC-TL) cuando las redes auxiliares solo varían en la cantidad de enlaces con la estructura de la red objetivo original, para la red Alarm. El eje X muestra la proporción de enlaces eliminados en la red auxiliar, el eje Z muestra la cantidad de casos utilizados en los dominios auxiliares. El eje Y muestra el error medido en distancia de edición obtenido para cada prueba. La cantidad de ejemplos utilizados para la red objetivo se mantuvo fijo a 100 casos.

objetivo en la cantidad de enlaces presente. Se varió la cantidad de enlaces presentes en las redes auxiliares desde $10\,\% - 90\,\%$ menos de enlaces que los presentes en la red objetivo. Y se ejecutó el algoritmo utilizando 100 instancias generadas para la red objetivo y de $500 - 8,000$ instancias generadas para las redes auxiliares.

En las figuras 6.6, 6.7 y 6.8 podemos observar en todas ellas que el método propuesto pierde la habilidad de recuperar la verdadera estructura conforme los dominios auxiliares van perdiendo la similitud con el dominio objetivo. Ésta pérdida de habilidad se presenta cuando los dominios auxiliares pierden similitud, pero como se puede ver en los experimentos realizados, no se necesitan grandes requerimientos en este aspecto para obtener buenos resultados. En la figura 6.6 podemos observar que el método propuesto aprovecha las similitudes provenientes en todos los dominios auxiliares mejorando la recuperación estructural cuando se utilizan varios conjuntos de datos similares. Puede verse con mayores detalles en la figura 6.8 cuando se obtienen dominios auxiliares con gran similitud se mejora la recuperación de la verdadera estructura, y

combinando una mayor cantidad de ejemplos para los dominios auxiliares, se obtiene mejores resultados.

6.4. Discusión y análisis.

Al estudiar los resultados obtenidos para el método de aprendizaje estructural propuesto podemos observar que el comportamiento sobre la recuperación estructural del método está muy relacionado con la cantidad de datos utilizados para el dominio objetivo y auxiliares. Primero, mientras aumenta la cantidad de datos utilizados para el dominio objetivo, el método propuesto tiende a converger a los mismos resultados que el algoritmo PC [SGS93]. Sin embargo, cuando la diferencia en la cantidad de datos entre el dominio objetivo y los auxiliares aumenta, se puede observar que el error en la recuperación estructural empieza a aumentar. Si se cumple que los dominios son similares, se tiene una mejor aproximación del modelo buscado, sin embargo, como observamos en los experimentos en la red Alarm e Insurance (los resultados para la red Boblo e Insurance se muestran en el apéndice A), en prácticamente todos los casos, los dominios auxiliares ayudan a mejorar el modelo cuando se tienen pocos datos. En redes como la red Boblo, la mejora en el aprendizaje utilizando el método propuesto es escasa, comparándola con las otras redes del experimento. Si bien, esto no necesariamente representa que el método falle, sino podría ser que ante dominios más complejos sea necesaria una mayor transferencia de aprendizaje o cantidad datos. Para transferir el aprendizaje selectivamente, el algoritmo determina en cada prueba de independencia condicional sobre el dominio objetivo al dominio más relacionado mediante similitudes en las pruebas de independencia. Como es necesaria una mayor cantidad de datos para conseguir una fiabilidad uniforme conforme aumenta la complejidad de las pruebas de independencia condicional, se determina el dominio más relacionado utilizando únicamente en esta etapa pruebas de independencia de orden cero. El grado de transferencia de información es obtenida combinando medidas de similitud (mediante la ecuación 4.4) obtenidas mediante el acuerdo en las relaciones de dependencias/independencia condicionales entre el dominio objetivo contra los dominios auxiliares y transfiriendo el aprendizaje en las partes de la red donde se pre-

sente mayor similitud y menor fiabilidad en las pruebas de independencia condicional aplicadas en el dominio objetivo.

La figura 6.9 muestra algunos ejemplos obtenidos al realizar una prueba de un intento en el aprendizaje estructural utilizando los dominios auxiliares de la tabla 6.2. Como podemos ver en la figura, cuando se tiene poca cantidad de datos disponible para el dominio objetivo, el algoritmo propuesto tiene una mejor aproximación a la verdadera estructura de la red. Cuando existen dos dominios con diferente grado de similitud con el dominio objetivo, la recuperación de la red objetivo no parece tener diferencia en las pruebas presentadas a este nivel de similitud, como es el caso de los ejemplos presentados, donde se utilizan dos redes similares (las utilizadas en la tabla 6.2).

El aprendizaje paramétrico a partir de datos es una tarea muy difícil de resolver debido a la gran cantidad de parámetros que se requieren calcular. Para estimar los parámetros correspondientes a las tablas de probabilidad requeridas en el aprendizaje de redes bayesianas se requiere una enorme cantidad de datos, usualmente no disponibles. Como un método de mejorar la estimación de estas tablas se han propuesto diversos métodos basados en una combinación de probabilidades a partir de varias fuentes de datos (normalmente provenientes del mismo dominio). Los métodos propuestos se basan en el método de combinación lineal (ver sección 5.3) mientras se mantiene una similar eficiencia computacional. Ambos métodos aplican una medida de peso a las tablas de probabilidad provenientes del dominio objetivo mientras se combinan con los datos de los dominios auxiliares. El primer método (DBLP) combina estos datos centrándose en los datos objetivo mientras el segundo método descarta aquellos datos más alejados en distancia de los datos del dominio objetivo y los combina con los dominios auxiliares. Los dos métodos propuestos no parecen tener mucha diferencia de acuerdo a los resultados obtenidos, sin embargo, como se muestra en la figura A.12, mejoran la recuperación de una red más compleja como la red Insurance en ciertas partes de las pruebas realizadas. Al observar los resultados obtenidos, se puede notar que los métodos propuestos en esta tesis presentan un buen compromiso en el costo computacional y recuperación de los parámetros de las redes bayesianas.

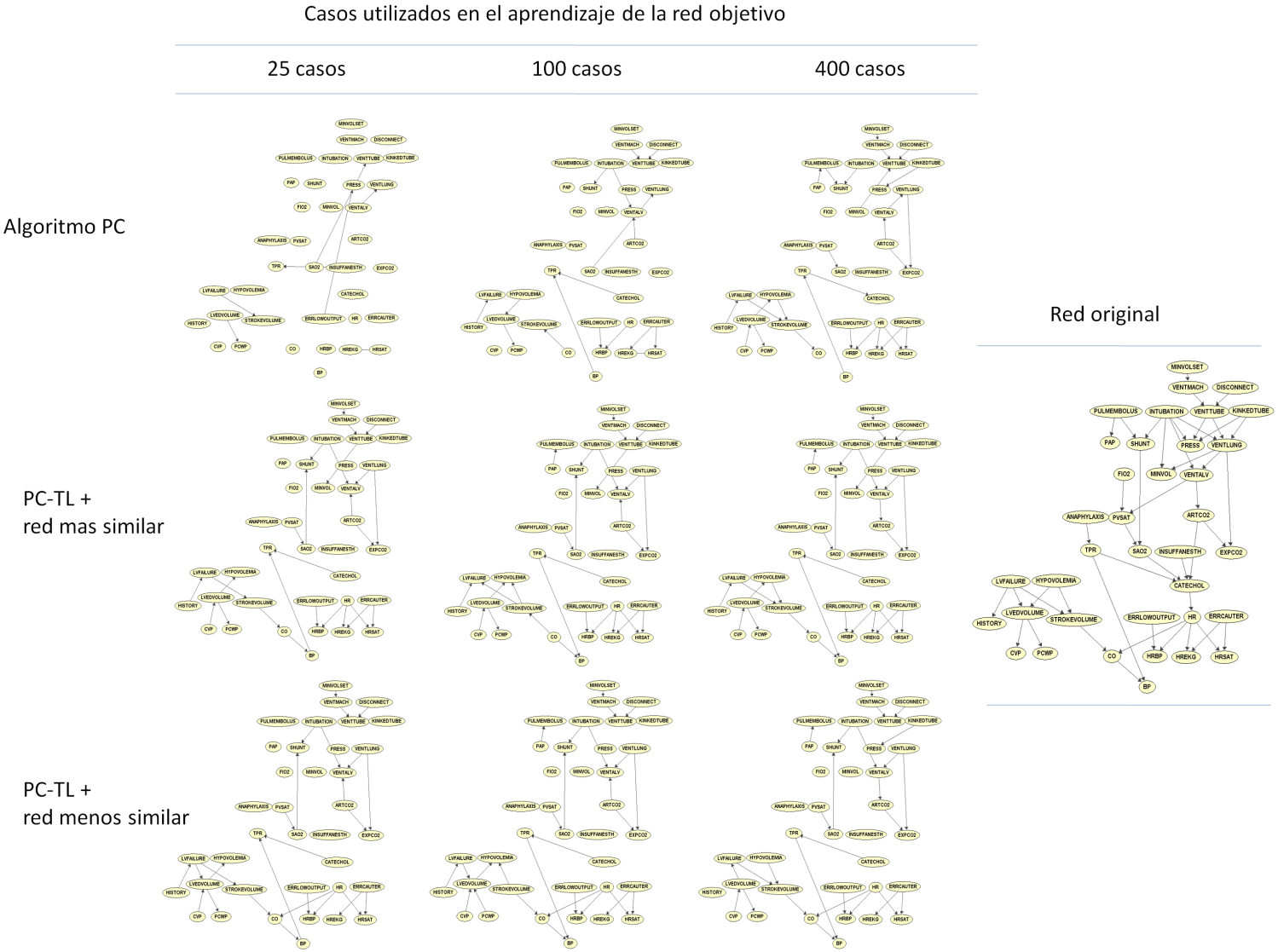


Figura 6.9: Algunas estructuras aprendidas por el algoritmo PC y el algoritmo propuesto (PC-TL), utilizando las redes auxiliares descritas en la tabla 6.2. El numero de casos para el dominio auxiliar se ha fijado a $1,000$.

Capítulo 7

Conclusiones y Trabajo Futuro

7.1. Resumen

En esta tesis se desarrollaron métodos de aprendizaje estructural y paramétricos de redes bayesianas enfocados en el aprendizaje de modelos con pocos datos mediante la utilización de aprendizaje por transferencia desde tareas relacionadas. Para esto se estudiaron los distintos métodos comúnmente utilizados para el aprendizaje de redes bayesianas en el capítulo 2, así como también algunos trabajos que emplean información desde diversas fuentes de datos para mejorar el modelo a aprender mediante técnicas de aprendizaje por transferencia en el capítulo 3. Se propuso, para el aprendizaje estructural, la aplicación de aprendizaje por transferencia durante la fase de construcción del modelo en un método de aprendizaje basado en criterios de independencias. El método propuesto presenta un enfoque novedoso en aprendizaje por transferencia aplicado en redes bayesianas, mostrando una mejora en la recuperación de la estructura. De forma similar, para el aprendizaje paramétrico, se propusieron dos métodos de aprendizaje paramétrico basados en la combinación de los parámetros objetivo con los parámetros de las tareas relacionadas. El primer método, llamado Combinación Lineal Basada en Distancia (DBLP), considera la fiabilidad de las probabilidades de todas las TPC´s participantes de todas las redes bayesianas, y modificando la TPC de interés

moviéndose en el sentido de las opiniones más próximas entre ellas, y alejándose de las más lejanas a la misma. Mientras el segundo método, llamado Combinación Lineal Local (LoLP), sólo descarta aquellos valores de probabilidad que se encuentran lejanos a una determinada distancia (dada por la ecuación 5.5) de la medida de probabilidad dada por la TPC de interés y los combina con el promedio de las probabilidades restantes, pesada por la cantidad de datos en el dominio de interés. Se probaron los métodos propuestos y se compararon mediante la distancia de edición y el error cuadrático medio para los métodos de aprendizaje estructural y paramétricos respectivamente con la finalidad de observar el comportamiento con bases de datos creadas artificialmente a partir de redes bayesianas comúnmente utilizadas en la literatura, y se encontró, para el caso del método de aprendizaje estructural propuesto, mejoras significativas en la recuperación de la estructura original, y para el caso de los métodos de aprendizaje paramétrico, mejoras en la recuperación de los parámetros, comparables al método de combinación lineal utilizado como base, pero superiores a la utilización de sólo los datos disponibles para el problema (es decir, sin la utilización de información de dominios relacionados).

En el caso del costo computacional, debido a la carga de utilizar más fuentes de información, éste se incrementa en función del número de tareas relacionadas involucradas en el aprendizaje. Para el caso del método de aprendizaje estructural propuesto, involucra el cálculo inicial de encontrar las similitudes de la tarea objetivo sobre las tareas relacionadas, para posteriormente continuar con la fase de construcción de la red transfiriendo selectivamente desde las tareas más relacionadas únicamente. Para el caso de los métodos de aprendizaje paramétrico, se utiliza una combinación lineal sobre todas las tareas relacionadas manteniendo un costo similar a los métodos base.

7.2. Conclusiones

El método de aprendizaje estructural propuesto se puede considerar una extensión al algoritmo PC [SGS93], sin embargo la utilización de información proveniente de dominios similares en los cálculos de las pruebas de independencia condicional fueron

incluidos permitiendo que la recuperación de las redes objetivo fueran mejoradas, y de lo cual se concluye lo siguiente:

- La exactitud de las pruebas de independencia condicional de la tarea objetivo es apoyada utilizando aprendizaje por transferencia selectiva desde los dominios más relacionados, repercutiendo en una significativa mejora en la recuperación de la estructura cuando se tienen bases de datos relativamente pequeñas.
- La recuperación de la estructura puede ser mejorada notablemente mientras mayor sea la similitud de las tareas relacionadas con el modelo objetivo buscado.

En esta tesis se hizo una comparación del método propuesto con el algoritmo PC (utilizado como base), sin embargo existen algunos métodos que incorporan algunas estrategias para mejorarlo [AGOM03]. El algoritmo propuesto muestra la utilización de la técnica de aprendizaje por transferencia al aprendizaje estructural de redes bayesianas, la cual puede incorporar estrategias convencionales para aumentar aún más la precisión en la recuperación de los modelos. Además se presentaron dos propuestas que permiten incluir información de las TPC de dominios similares mejorando la exactitud en la recuperación de los parámetros comparados contra el método base, y demostrando un mejor acercamiento a la verdadera distribución de los parámetros cuando se utiliza información de dominios similares.

Como se ha mostrado en este trabajo, los métodos de aprendizaje utilizando información de dominios similares aumenta la precisión en la recuperación de la verdadera estructura y mejora la precisión en la recuperación de los parámetros numéricos de las TPC de las redes bayesianas construidas, comparándolos con los métodos tradicionales que no utilizan técnicas de aprendizaje por transferencia, en particular, cuando se tienen pocos datos en el dominio de interés.

7.3. Trabajo futuro

Existen varias ramas por las cuales el presente trabajo se puede extender a fin de mejorar el aprendizaje estructural y paramétrico de redes bayesianas:

- Utilizar una combinación de varios dominios relacionados para la transferencia de información.
- Utilizar una medida adaptable de transferencia de información mientras se construye la red.
- Proponer una política de transferencia de información cuando las bases de datos objetivo sean muy pequeñas.

El primer punto involucra la utilización de varios dominios similares al mismo tiempo en la función de combinación final, esto ayudaría al método propuesto en situaciones en donde se encuentren similitudes parciales no presentes al calcular las pruebas de independencia condicional con orden cero, pero que se pudieran encontrar en pruebas de orden superior. Para el punto dos, la búsqueda de similitudes se restringe a las pruebas de independencia de orden cero por lo que se mantiene una medida fija de transferencia a lo largo del proceso de aprendizaje. Utilizar una medida adaptable, es decir, modificable durante el proceso de aprendizaje podría incrementar la exactitud de la recuperación al encontrarse durante este proceso mayor o menor evidencias de similitud, por lo que mayor cantidad de transferencia de información. Finalmente el último punto, parece ser la principal desventaja del algoritmo propuesto (al utilizar bases de datos muy pequeñas) debido a que como la transferencia se basa en similitudes en las pruebas de independencia condicional, cuando la cantidad de datos es insuficiente se obtienen independencias condicionales en las pruebas lo que conlleva a construir redes poco conectadas, por lo que redes menos conectadas serán las que se utilizan como fuente de transferencia.

El aprendizaje paramétrico de redes bayesianas es muy complicado ya que se requiere una enorme cantidad de datos a fin de tener estimaciones precisas de los parámetros. Los métodos paramétricos propuestos utilizaron sencillos cálculos para la obtención de los parámetros basados en el conteo del número de ocurrencias de los datos, sin embargo en un trabajo futuro se tiene pensado probar la utilización de métodos de estimación de parámetros mediante funciones de densidad (gausianas, etc). Cuando las estructuras gráficas de la red poseen una gran cantidad de padres por nodo, se

requiere una gran cantidad de datos para una mejor estimación. Se piensa que la investigación en una función que permita descomponer en parte más pequeñas estas estructuras podría ayudar a mejorar la estimación de parámetros de la red objetivo. La precisión perdida a causa de la función de descomposición podría ser mejorada utilizando el conocimiento disponible en dominios auxiliares.

Referencias

[AdC96] S. Acid y L. M. de Campos, “BENEDICT: An algorithm for learning probabilistic belief networks,” en “Proceedings of the IPMU-96 Conference,” páginas 979–984, 1996. [Citada en p. 16]

[AGOM03] J. Abellan, M. Gomez-Olmedo y S. Moral, “Some Variations on the PC Algorithm,” 2003. [Citada en p. 30, 83]

[AZ05] Rie Kubota Ando y Tong Zhang, “A Framework for Learning Predictive Structures from Multiple Tasks and Unlabeled Data,” noviembre 2005. [Citada en p. 37]

[Bax97] Jonathan Baxter, “A Bayesian/Information Theoretic Model of Learning to Learn via Multiple Task Sampling,” *Machine Learning*, Vol. 28(1):páginas 7–39, 1997. [Citada en p. 3, 5, 32]

[Bax00] J. Baxter, “A Model of Inductive Bias Learning,” *Journal of Artificial Intelligence Research*, Vol. 12:páginas 149–198, 2000, ISSN 1076-9757. [Citada en p. 34]

[BDS03] Ben-David y Schuller, “Exploiting Task Relatedness for Multiple Task Learning,” en “COLT: Proceedings of the Workshop on Computational Learning Theory, Morgan Kaufmann Publishers,” 2003. [Citada en p. 34, 37]

[BH03] Bart Bakker y Tom Heskes, “Task Clustering and Gating for Bayesian Multitask Learning,” mayo 2003. [Citada en p. 34, 37]

[BSCC89] I. A. Beinlich, H. J. Suermondt, R. M. Chavez y G. F. Cooper, “The ALARM monitoring system: A case study with two probabilistic inference techniques for belief networks,” en “Proceedings of the Second European Conference on Artificial Intelligence in Medicine, London,” Springer Verlag, Berlin, agosto 1989. [Citada en p. 68]

[Bun91] W. Buntine, “Theory refinement on Bayesian networks.” *UAI ’91*, páginas 52–60, 1991. [Citada en p. 28]

[Bun96] W. Buntine, “A guide to the literature on learning probabilistic networks from data,” *Ieee Trans. On Knowledge And Data Engineering*, Vol. 8:páginas 195–210, 1996. [Citada en p. 1, 23]

[Car96] Rich Caruana, “Algorithms and Applications for Multitask Learning,” en “ICML,” páginas 87–95, 1996. [Citada en p. 32]

[Car97] Rich Caruana, “Multitask Learning,” *Machine Learning*, Vol. 28(1):páginas 41–75, 1997. [Citada en p. 3, 5, 32, 33, 34, 38]

[CC96] Chih-Shyang Chang y Arbee L. P. Chen, “Aggregate Functions over Probabilisitic Data,” *Inf. Sci.*, Vol. 88(1-4):páginas 15–45, 1996. [Citada en p. 54]

[CCT96] Arbee L. P. Chen, Jui-Shang Chiu y Frank Shou-Cheng Tseng, “Evaluating Aggregate Operations Over Imprecise Data,” *IEEE Transactions on Knowledge and Data Engineering*, Vol. 8(2):páginas 273–284, abril 1996. [Citada en p. 54]

[CGH97] Enrique Castillo, Jose Manuel Gutierrez y Ali S. Hadi, *Expert Systems and Probabilistic Network Models*, Springer, New York, 1 edición, 1997, ISBN 0-387-94858-9. [Citada en p. 25]

[CH92] G. F. Cooper y E. Herskovits, “A Bayesian method for the induction of probabilistic networks from data.” *Machine Learning*, Vol. 9:páginas 309–347, octubre 1992. [Citada en p. 2, 3, 15, 26, 27, 30]

[CHM97] Gregory Cooper, David Heckerman y Christopher Meek, “A Bayesian Approach to Causal Discovery,” Informe Técnico MSR-TR-97-05, Microsoft Research (MSR), febrero 1997. [Citada en p. 29]

[CL68] C. K. Chow y C. N. Liu, “Approximating discrete probability distributions with dependence trees.” *IEEE Trans. on Inf. Theory*, Vol. IT-14(3):páginas 462–467, mayo 1968. [Citada en p. 23, 24]

[DD99] Denver Dash y Marek J. Druzdzel, “A Hybrid Anytime Algorithm for the Construction of Causal Models From Sparse Data,” octubre 11 1999. [Citada en p. 30]

[DSM00] Jose Del Sagrado Martínez, *Fusión topológica y cuantitativa de redes causales*, Tesis Doctoral, Universidad de Granada, 2000. [Citada en p. 52, 55]

[Elv02] Elvira, “Elvira: An Environment for Creating and Using Probabilistic Graphical Models,” en José A. Gámez y Antonio Salmerón, editores, “First European Workshop on Probabilistic Graphical Models, 6-8 November - 2002 - Cuenca (Spain), Electronic Proceedings,” 2002. [Citada en p. 62, 66]

[EMP05] Theodoros Evgeniou, Charles A. Micchelli y Massimiliano Pontil, “Learning Multiple Tasks with Kernel Methods,” abril 2005. [Citada en p. 37]

[FY96] N. Friedman y Z. Yakhini, “On the Sample Complexity of Learning Bayesian Networks,” en Eric J. Horvitz y Finn Verner Jensen, editores, “Proceedings of the 12th Conference on Uncertainty in Artificial Intelligence,” páginas 274–282, Morgan-Kaufmann, 1996. [Citada en p. 44]

[GZ86] Christian Genest y James V. Zidek, “Combining probability distributions: A critique and an annotated bibliography,” *Statistical Science*, Vol. 1(1):páginas 114–148, 1986. [Citada en p. 54, 59]

[HC90] E. Herskovits y G. Cooper, “Kutató: An Entropy-Driven System for Construction of Probabilistic Expert Systems from Databases,” en Piero P. Bonissone, Max Henrion, Laveen Kanal y John F. Lemmer, editores, “Proceedings of the Sixth Conference on Uncertainty in Artificial Intelligence,” páginas 117–125, Elsevier, 1990. [Citada en p. 2, 24, 25]

[Hec95] David Heckerman, “A Tutorial on Learning With Bayesian Networks,” Informe Técnico MSR-TR-95-06, Microsoft Research, marzo 1995, revised November 1996. [Citada en p. 26]

[HGC95] David Heckerman, Dan Geiger y David M. Chickering, “Learning Bayesian networks: The combination of knowledge and statistical data,” en “Machine Learning,” páginas 197–243, 1995. [Citada en p. 2]

[Jeb04] Tony Jebara, “Multi-task feature and kernel selection for SVMs,” en Carla E. Brodley, editor, “ICML,” Vol. 69 de *ACM International Conference Proceeding Series*, ACM, 2004. [Citada en p. 37]

[KP07] Samuel Kaski y Jaakko Peltonen, “Learning from Relevant Tasks Only,” en Joost N. Kok, Jacek Koronacki, Ramon López de Mántaras, Stan Matwin, Dunja Mladenic y Andrzej Skowron, editores, “ECML,” Vol. 4701 de *Lecture Notes in Computer Science*, páginas 608–615, Springer, 2007, ISBN 978-3-540-74957-8. [Citada en p. 35, 38]

[LGC07] Jun Won Lee y Christophe G. Giraud-Carrier, “Transfer Learning in Decision Trees,” en “IJCNN,” páginas 726–731, IEEE, 2007. [Citada en p. 35, 36]

[MM06] L. Mihalkova y R. Mooney, “Transfer learning with markov logic networks,” 2006. [Citada en p. 36]

[Nea03] R. E. Neapolitan, *Learning Bayesian Networks.*, Prentice Hall, 2003. [Citada en p. 14]

[NMC07] Alexandru Niculescu-Mizil y Rich Caruana, “Inductive Transfer for Bayesian Network Structure Learning,” en “Proceedings of the 11th International Conference on AI and Statistics (AISTATS ‘07),” 2007. [Citada en p. 37, 38, 62]

[Pea88] J. Pearl, *Probabilistic reasoning in intelligent systems:networks of plausible inference*, Morgan Kaufmann, San Francisco, CA., 1988. [Citada en p. 3, 10, 16, 18, 19, 30, 52]

[Qui86] J. R. Quinlan, “Induction of decision trees,” *Machine Learning*, Vol. 1(1):páginas 81–106, 1986. [Citada en p. 35]

[RD03] Matt Richardson y Pedro Domingos, “Learning with Knowledge from Multiple Experts,” en Tom Fawcett y Nina Mishra, editores, “Machine Learning, Proceedings of the Twentieth International Conference (ICML 2003), August 21-24, 2003, Washington, DC, USA,” páginas 624–631, AAAI Press, 2003, ISBN 1-57735-189-4. [Citada en p. 36]

[Ris86] J. Rissanen, “Stochastic complexity and modeling,” *Annals of Statistics*, Vol. 14:páginas 1080–1100, 1986. [Citada en p. 25]

[Rob77] R. W. Robinson, “Counting unlabeled acyclic digraphs,” en C. H. C. Little, editor, “Combinatorial Mathematics V,” Vol. 622 de *Lecture Notes in Mathematics*, páginas 28–43, Springer, Berlin, 1977. [Citada en p. 16]

[RP87] George Rebane y Judea Pearl, “The Recovery of Causal Poly-Trees from Statistical Data,” en Laveen N. Kanal, Tod S. Levitt y John F. Lemmer, editores, “UAI,” páginas 175–182, Elsevier, 1987. [Citada en p. 23, 24]

[SGS93] Peter Spirtes, Clark Glymour y Richard Scheines, *Causation, prediction, and search*, Springer-Verlag, Berlin, 1993. [Citada en p. 19, 21, 29, 42, 78, 82]

[Suz96] Joe Suzuki, “Learning Bayesian Belief Networks Based on the MDL Principle: An Efficient Algorithm Using the Branch and Bound Technique,” en Lorenza Saitta, editor, “Proceedings of the Thirteenth International Conference on Machine Learning,” páginas 462–470, Morgan-Kaufmann, 1996. [Citada en p. 2]

[SV93] Moninder Singh y Marco Valtorta, “An Algorithm for the Construction of Bayesian Network Structures from Data,” en David Heckerman y Abe Mamdani, editores, “Proceedings of the Ninth Conference on Uncertainty in Artificial Intelligence,” páginas 259–265, Morgan-Kaufmann, 1993. [Citada en p. 16]

[SV95] M. Singh y M. Valtorta, “Construction of Bayesian network structures from data: a brief survey and an efficient algorithm,” *International Journal of Approximate Reasoning*, Vol. 12:páginas 111–131, 1995. [Citada en p. 29]

[SZ06] Jiang Su y Harry Zhang, “Full Bayesian network classifiers,” en William W. Cohen y Andrew Moore, editores, “Machine Learning, Proceedings of the Twenty-Third International Conference (ICML 2006), Pittsburgh, Pennsylvania, USA, June 25-29, 2006,” Vol. 148 de *ACM International Conference Proceeding Series*, páginas 897–904, ACM, 2006, ISBN 1-59593-383-2. [Citada en p. 44]

[Thr96] Sebastian Thrun, “Is Learning The n-th Thing Any Easier Than Learning The First?” en David S. Touretzky, Michael C. Mozer y Michael E. Hasselmo, editores, “Advances in Neural Information Processing Systems,” Vol. 8, páginas 640–646, The MIT Press, 1996. [Citada en p. 3, 5, 32]

[TO96] Sebastian Thrun y Joseph O’Sullivan, “Discovering structure in multiple learning tasks: the TC algorithm,” en “Proc. 13th International Conference on Machine Learning,” páginas 489–497, Morgan Kaufmann, 1996. [Citada en p. 34]

[vDvdGT03] Steven van Dijk, Linda C. van der Gaag y Dirk Thierens, “A Skeleton-Based Approach to Learning Bayesian Networks from Data,” en Nada Lavrač, Dragan Gamberger, Ljupčo Todorovski y Hendrik Blockeel, editores, “Lecture Notes in Computer Science, Volume 2838: Proceedings of the Seventh Conference on Principles and Practice of Knowledge Discovery in Databases,” páginas 132–143, Springer, 2003. [Citada en p. 30]

[XLCK07] Ya Xue, Xuejun Liao, Lawrence Carin y Balaji Krishnapuram, “Multi-Task Learning for Classification with Dirichlet Process Priors,” *Journal of Machine Learning Research*, Vol. 8:páginas 35–63, 2007. [Citada en p. 34]

[ZGY05] Jian Zhang, Zoubin Ghahramani y Yiming Yang, “Learning Multiple Related Tasks using Latent Independent Component Analysis,” en “NIPS,” 2005. [Citada en p. 34, 37]

Apéndice A

Apéndice

En este apéndice se presentan los resultados obtenidos cuando se aplica los mismos experimentos presentados en el capítulo 6 a las redes Boblo e Insurance.

A.1. Red Boblo

La red Boblo (BOvine BLOod), que se muestra en la figura A.1, es un sistema de ayuda para la verificación del parentesco del ganado a través de la identificación del tipo de sangre.

EXPERIMENTO TIPO I

Para realizar los experimentos de este tipo, se han utilizado redes auxiliares con las características provista en la tabla A.1.

Se realizaron los mismos experimentos descritos para la red Alarm en la sección 6.3. Las figuras A.2, A.3 muestran el comportamiento del método propuesto de aprendizaje estructural cuando se varía la cantidad de casos para el dominio objetivo (manteniendo constante la cantidad de casos para los dominios auxiliares) y posteriormente para

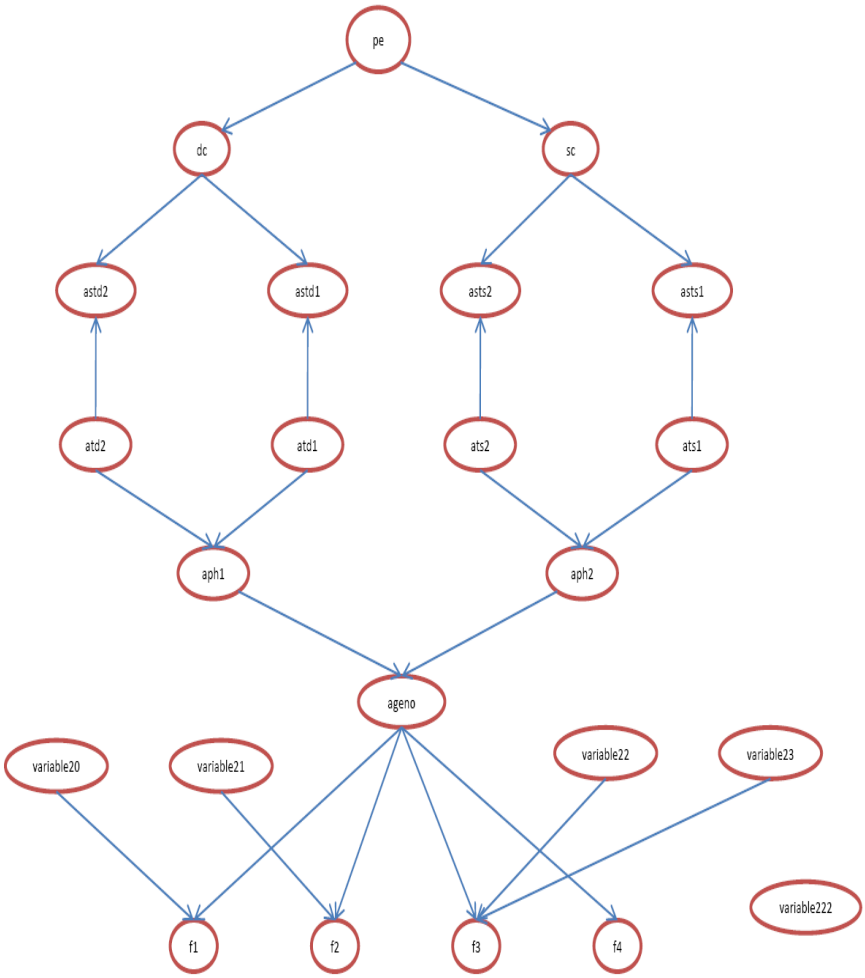


Figura A.1: Estructura original de la red Boblo

		Arcos añadidos	Arcos eliminados	Arcos revertidos	Distancia de Edición	Ruido aplicado a las TPC (%)	Error Cuadrático Medio (MSE)
Boblo	Red Mas Similar	1	0	0	1	5%	0.011
	Red Menos Similar	4	4	0	8	20%	0.029

Tabla A.1: Características de las redes auxiliares utilizadas en los experimentos tipo I para la red Boblo.

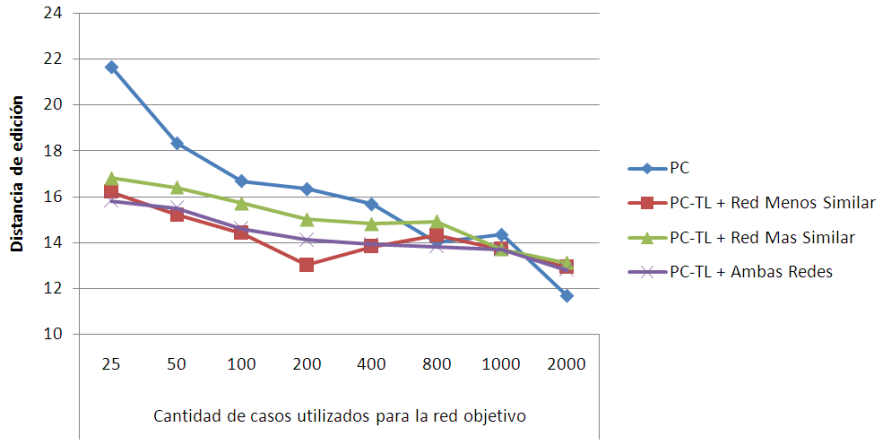


Figura A.2: Comparación entre el algoritmo PC y PC-TL considerando diferente número de casos y utilizando dos conjuntos de datos auxiliares. La figura muestra el comportamiento medido como la reducción en la distancia de edición mientras se incrementa el número de casos para la red objetivo mientras se mantiene fijo (a $1,000$ casos) para las redes auxiliares en la red Boblo.

los dominios auxiliares (manteniendo constante la cantidad de casos para el dominio objetivo).

Similarmente, utilizando como base las estructuras encontradas por el método de aprendizaje estructural propuesto en el experimento anterior, se probaron los métodos de aprendizaje paramétrico mostrándose los resultados en las figuras A.4 y A.5. La figura A.4 muestra los resultados cuando se incrementa la cantidad de casos para el dominio objetivo, utilizando dos dominios auxiliares con diferente grado de similitud con el dominio objetivo. La figura A.5 muestra los resultados cuando se incrementa la cantidad de casos para los dominios auxiliares utilizando, nuevamente, dos dominios auxiliares con diferente grado de similitud con el dominio objetivo.

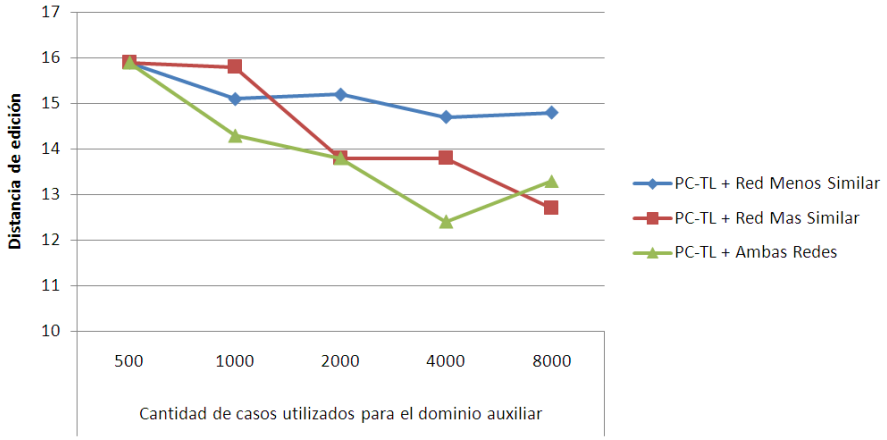


Figura A.3: Comparación entre el algoritmo PC y PC-TL considerando diferente número de casos y utilizando dos conjuntos de datos auxiliares. La figura muestra el comportamiento medido como la reducción en la distancia de edición mientras se incrementa el número de casos para las redes auxiliares mientras se mantiene fijo(a $1,000$ casos) para la red objetivo en la red Boblo.

Al observar los resultados del método propuesto de aprendizaje estructural en las figuras A.2 y A.3 podemos notar nuevamente la mejora en la recuperación de la estructura especialmente cuando se dispone de relativamente pocos casos. En las figuras A.2 y A.3 podemos observar una mejor recuperación de la estructura cuando la similitud de los dominios auxiliares con el dominio objetivo es mayor, y si adicionalmente se dispone de una mayor cantidad de casos utilizados en los dominios auxiliares se obtiene una mejor recuperación en la estructura del dominio objetivo.

Nuevamente al observar los resultados obtenidos en las figuras A.4 y A.5 podemos notar la mejora en la recuperación de los parámetros (con respecto al método base) principalmente cuando se dispone de relativamente pocos casos. Como se puede observar en las figuras, ocurre más favorablemente cuando la cantidad de casos para la red objetivo es relativamente baja comparándolo con los casos del dominio objetivo y mientras mayor sea la cantidad en ésta diferencia, se tienen mejores resultados en la recuperación de los parámetros.

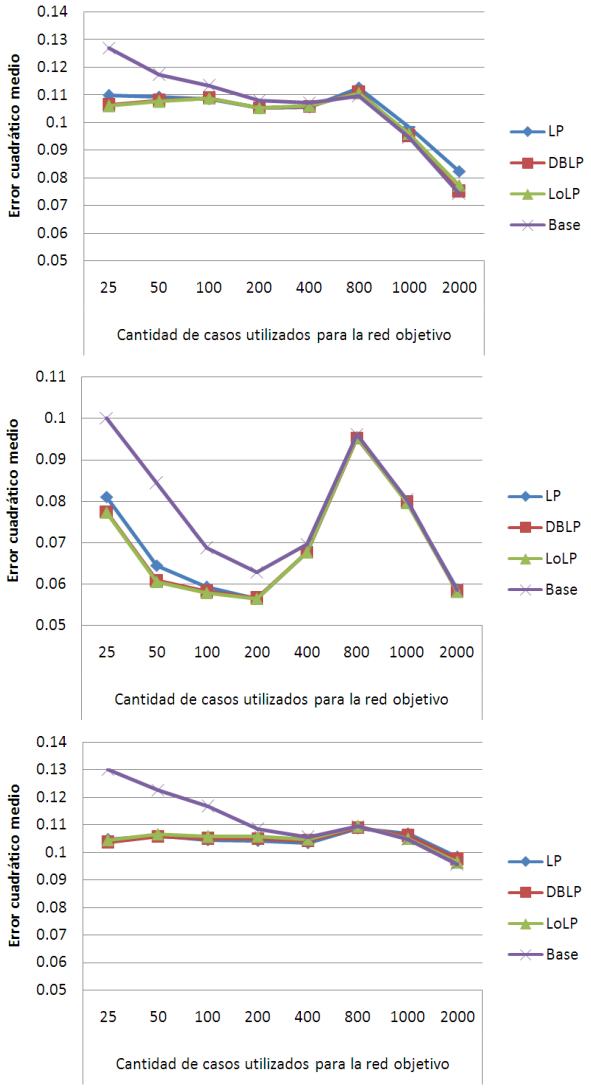


Figura A.4: Comparación entre el aprendizaje paramétrico cuando se incrementa el número de casos para la red objetivo en la red Boblo. La primera figura muestra la reducción en el error utilizando una red auxiliar muy similar, la figura central utilizando una red menos similar mientras que la última figura se muestra utilizando las dos redes auxiliares previas.

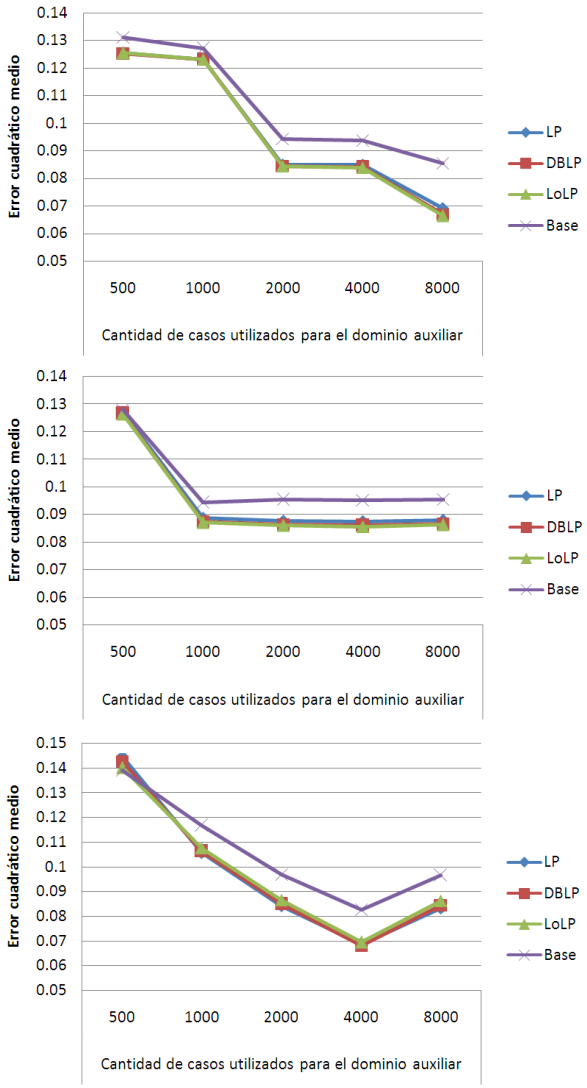


Figura A.5: Comparación entre el aprendizaje paramétrico cuando se incrementa el número de casos para las redes auxiliares en la red Boblo. La primera figura muestra la reducción en el error utilizando una red auxiliar muy similar, la figura central utilizando una red menos similar mientras que la última figura se muestra utilizando las dos redes auxiliares previas.

EXPERIMENTO TIPO II

En estos experimentos las figuras A.6 y A.7 muestran el comportamiento cuando se varía la similitud de los dominios auxiliares en la forma de la cantidad de enlaces añadidos y eliminados mientras se varía la cantidad de casos utilizados para el dominio objetivo y auxiliares. Mientras en la figura A.8 se muestra cuando la cantidad de enlaces en las estructuras de los modelos de los dominios auxiliares va disminuyendo partiendo de la verdadera estructura y adicionalmente mostrándose el comportamiento cuando aumenta la cantidad de casos únicamente para los dominios auxiliares.

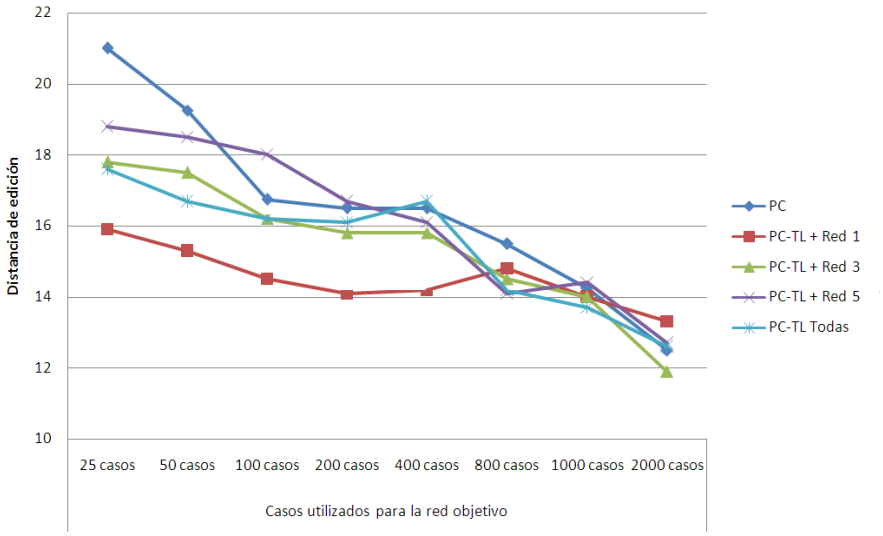


Figura A.6: Comportamiento del algoritmo PC-TL cuando se utilizan redes auxiliares cuyas características de similitud con la red objetivo varían, para la red Boblo. El eje X muestra la variación en el tamaño de la red objetivo, variando de $25 - 2000$ casos. El eje Y muestra el error medido en la distancia de edición con la verdadera estructura sobre cada prueba. La cantidad de casos utilizados para los dominios objetivos a sido fijado a 1000 casos.

Ambas pruebas utilizaron las redes auxiliares cuyas características se presentan en las tablas A.2 y A.3.

De las figuras A.6, A.7 y A.8 podemos observar que la recuperación de la estructura del modelo por parte del algoritmo propuesto PC-TL no ha podido tomar grandes ventajas de la similitud con los dominios auxiliares, pero aun así, el algoritmo propuesto a podido

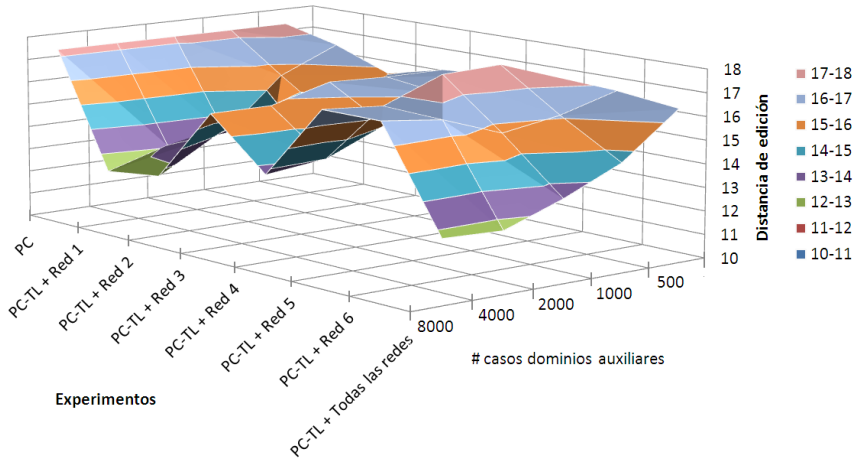


Figura A.7: Comportamiento del algoritmo PC-TL cuando se utilizan redes auxiliares cuyas características de similitud con la red objetivo varían mostrándose en el eje Z (experimentos), para la red Boblo. El eje X muestra la variación en el tamaño de la red auxiliar variando de $500 - 8000$ casos. El eje Y muestra el error medido en la distancia de edición con la verdadera estructura sobre cada prueba.

		Arcos añadidos	Arcos eliminados	Arcos revertidos	Distancia de Edición	Error Cuadrático Medio (MSE)
Red Boblo	Red Similar 1	1	1	0	2	0.015
	Red Similar 2	4	4	0	8	0.047
	Red Similar 3	7	7	0	14	0.059
	Red Similar 4	6	6	0	12	0.021
	Red Similar 5	10	10	0	20	0.082
	Red Similar 6	11	11	0	22	0.090

Tabla A.2: Características del conjunto 1 de redes utilizadas en los experimentos tipo II para la red Boblo.

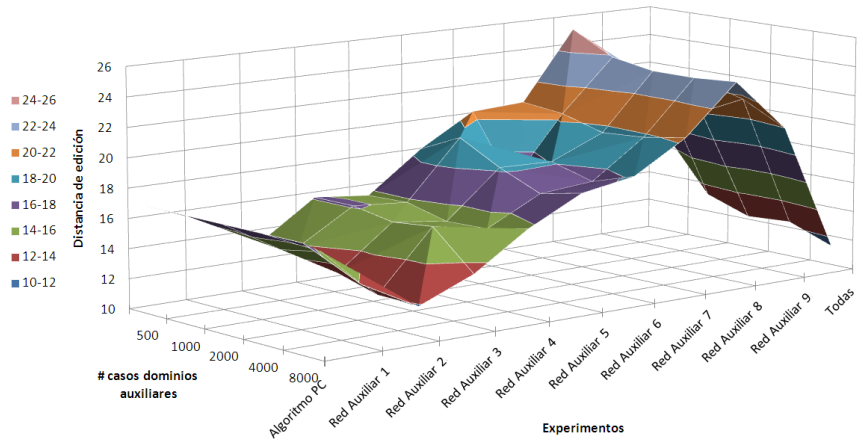


Figura A.8: Comportamiento del algoritmo cuando las redes auxiliares solo varían en la cantidad de enlaces con la red objetivo original, para la red Boblo. El eje X muestra la proporción de enlaces eliminados en la red auxiliar, el eje Z muestra la cantidad de casos utilizados en los dominios auxiliares. El eje Y muestra el error medido en distancia de edición obtenido para cada prueba. La cantidad de ejemplos utilizados para la red objetivo se mantuvo fija a 100 casos.

		Arcos añadidos	Arcos eliminados	Arcos revertidos	Distancia de Edición	Error Cuadrático Medio (MSE)
Red Boblo	Red Similar 1	2	0	0	2	0.023
	Red Similar 2	4	0	0	4	0.032
	Red Similar 3	7	0	0	7	0.051
	Red Similar 4	9	0	0	9	0.070
	Red Similar 5	12	0	0	12	0.097
	Red Similar 6	14	0	0	14	0.118
	Red Similar 7	16	0	0	16	0.111
	Red Similar 8	19	0	0	19	0.155
	Red Similar 9	21	0	0	21	0.151

Tabla A.3: Características del conjunto 2 de redes utilizadas en los experimentos tipo II para la red Boblo.

		Arcos añadidos	Arcos eliminados	Arcos revertidos	Distancia de Edición	Ruido aplicado a las TPC (%)	Error Cuadrático Medio (MSE)
Insurance	Red Mas Similar	3	2	0	5	5%	0.012
	Red Menos Similar	12	10	0	22	20%	0.052

Tabla A.4: Características de las redes auxiliares utilizadas en los experimentos tipo I para la red Insurance.

superar al algoritmo PC cuando se disponen pocos datos para el dominio objetivo. La recuperación ha sido mas fuerte cuando los dominios tienen gran similitud entre si, pero se decrementa en la medida que la similitud entre el dominio objetivo y los auxiliares disminuye. Los resultados nos hacen pensar que una mayor o menor recuperación en la estructura del dominio objetivo obedece en parte a la complejidad del modelo mismo que se requiere recuperar.

A.2. Red Insurance

La red Insurance, que se muestra en la figura A.9, es un sistema para clasificación en aplicaciones de seguros de automóvil.

Experimento tipo I

Para realizar los experimentos, se han utilizado redes auxiliares con las características provista en la tabla A.4.

Se realizaron los mismos experimentos descritos para la red Alarm en la sección 6.3. Las figuras A.10, A.11 muestran el comportamiento del método propuesto de aprendizaje estructural cuando se varía la cantidad de casos para el dominio objetivo (manteniendo constante la cantidad de casos para los dominios auxiliares) y posteriormente para los dominios auxiliares (manteniendo constante la cantidad de casos para el dominio objetivo).

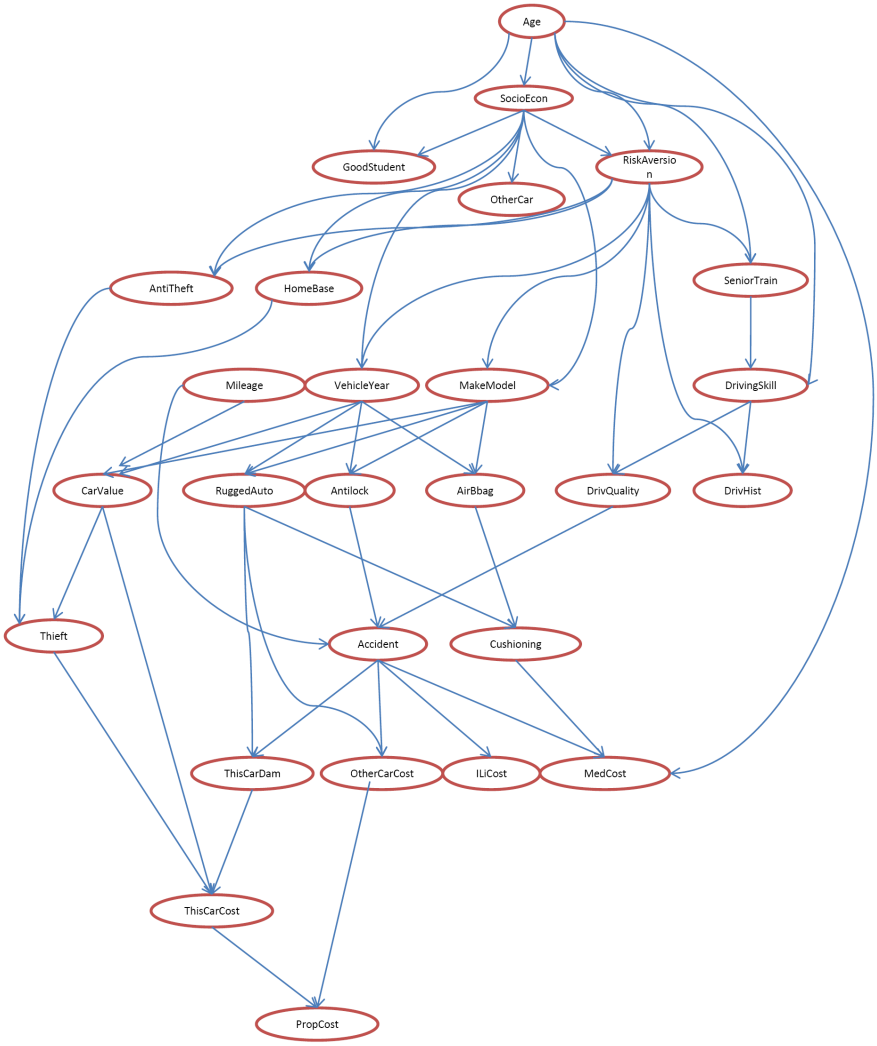


Figura A.9: Estructura original de la red Insurance.

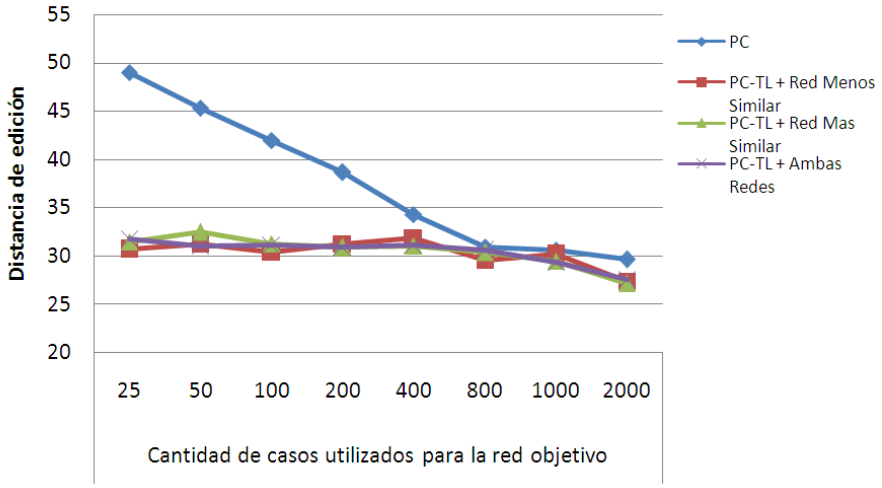


Figura A.10: Comparación entre el algoritmo PC y PC-TL considerando diferente número de casos y utilizando dos conjuntos de datos auxiliares. La figura muestra el comportamiento medido como la reducción en la distancia de edición mientras se incrementa el número de casos para la red objetivo mientras se mantiene fijo para las redes auxiliares en la red Insurance.

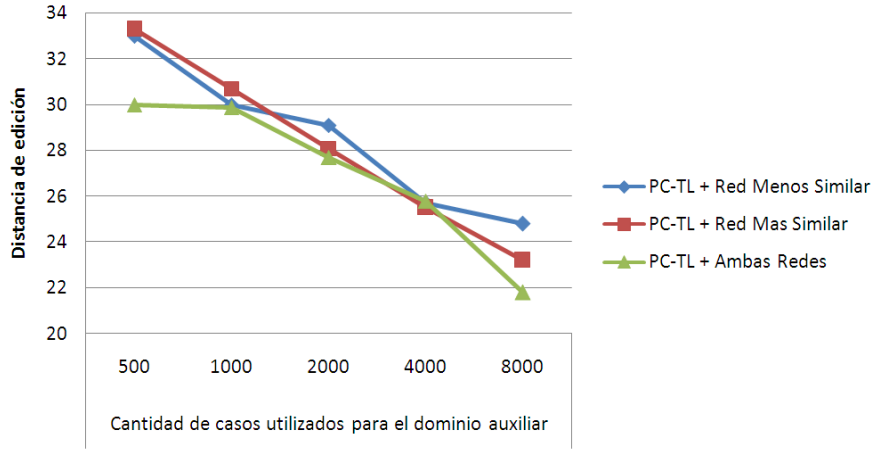


Figura A.11: Comparación entre el algoritmo PC y PC-TL considerando diferente número de casos y utilizando dos conjuntos de datos auxiliares. La figura muestra el comportamiento medido como la reducción en la distancia de edición mientras se incrementa el número de casos para las redes auxiliares mientras se mantiene fijo para la red objetivo en la red Insurance.

Similarmente, utilizando como base las estructuras encontradas por el método de aprendizaje estructural propuesto en el experimento anterior, se probaron los métodos de aprendizaje paramétrico mostrándose los resultados en las figuras A.12 y A.13. La figura A.12 muestra los resultados cuando se incrementa la cantidad de casos para el dominio objetivo, utilizando dos dominios auxiliares con diferente grado de similitud con el dominio objetivo. La figura A.13 muestra los resultados cuando se incrementa la cantidad de casos para los dominios auxiliares utilizando, nuevamente, dos dominios auxiliares con diferente grado de similitud con el dominio objetivo.

Al observar los resultados del método propuesto de aprendizaje estructural en las figuras A.10 y A.11 podemos notar nuevamente la mejora en la recuperación de la estructura especialmente cuando se dispone de relativamente pocos casos. En las figuras A.10 y A.11 podemos observar una mejor recuperación de la estructura cuando la similitud de los dominios auxiliares con el dominio objetivo es mayor, y si adicionalmente se dispone de una mayor cantidad de casos utilizados en los dominios auxiliares se obtiene una mejor recuperación en la estructura del dominio objetivo.

Nuevamente al observar los resultados obtenidos en las figuras A.12 y A.13 podemos notar la mejora en la recuperación de los parámetros (con respecto al método base) principalmente cuando se dispone de relativamente pocos casos. Como se puede observar en las figuras, ocurre más favorablemente cuando la cantidad de casos para la red objetivo es relativamente baja comparándolo con los casos del dominio objetivo y mientras mayor sea la cantidad en ésta diferencia, se tienen mejores resultados en la recuperación de los parámetros.

EXPERIMENTO TIPO II

En estos experimentos las figuras A.14 y A.15 muestran el comportamiento cuando se varía la similitud de los dominios auxiliares en la cantidad de enlaces añadidos y

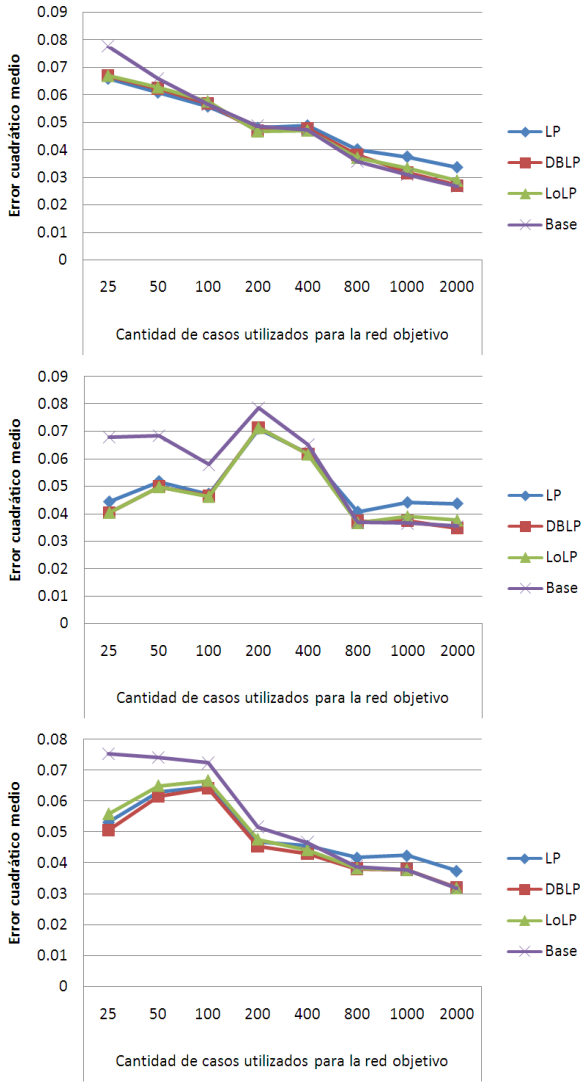


Figura A.12: Comparación entre el aprendizaje paramétrico cuando se incrementa el número de casos para la red objetivo en la red Insurance. La primera figura muestra la reducción en el error utilizando una red auxiliar muy similar, la figura central utilizando una red menos similar mientras que la ultima figura se muestra utilizando las dos redes auxiliares previas.

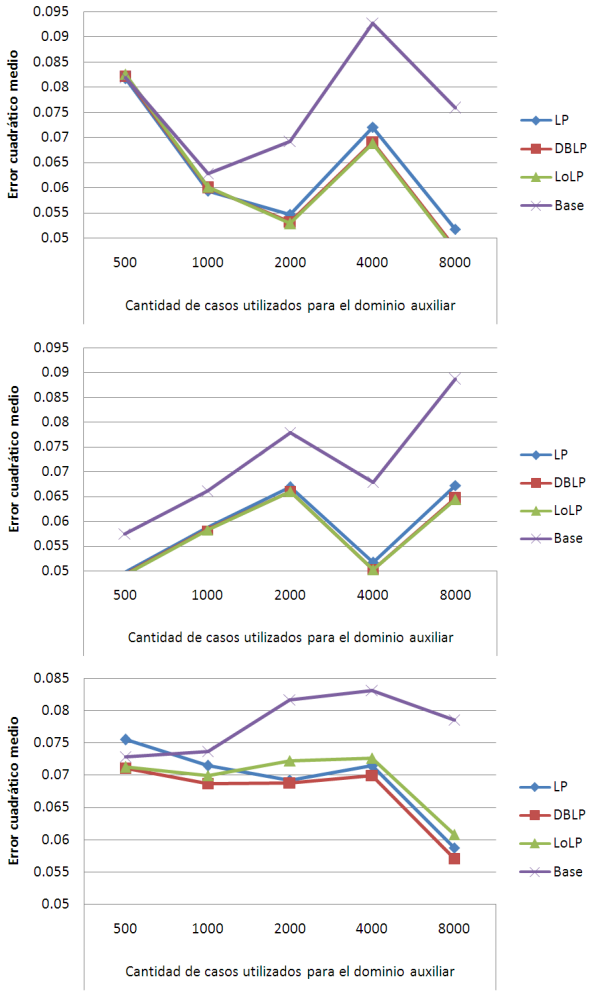


Figura A.13: Comparación entre el aprendizaje paramétrico cuando se incrementa el número de casos para las redes auxiliares en la red Insurance. La primera figura muestra la reducción en el error utilizando una red auxiliar muy similar, la figura central utilizando una red menos similar mientras que la ultima figura se muestra utilizando las dos redes auxiliares previas.

eliminados. Mientras en la figura A.16 se muestra cuando la cantidad de enlaces va disminuyendo partiendo de la verdadera estructura.

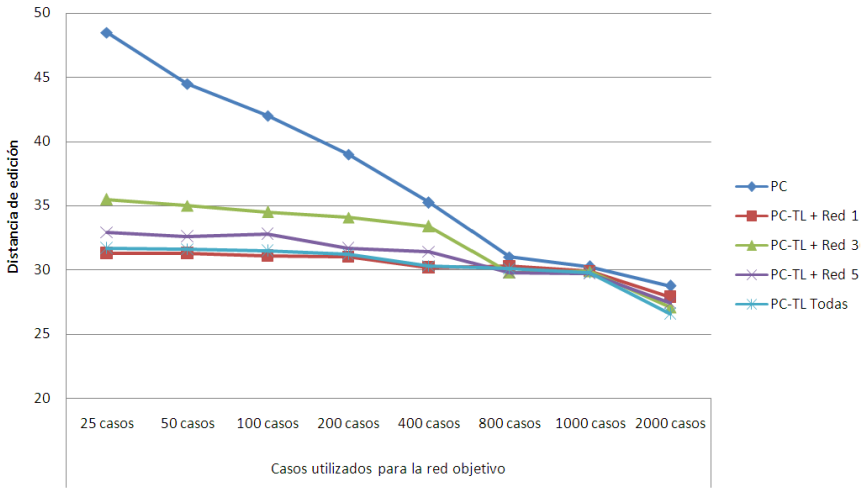


Figura A.14: Comportamiento del algoritmo PC-TL cuando se utilizan redes auxiliares cuyas características de similitud con la red objetivo varían, para la red Boblo. El eje X muestra la variación en el tamaño de la red objetivo, variando de $25 - 2000$ casos. El eje Y muestra el error medido en la distancia de edición con la verdadera estructura sobre cada prueba. La cantidad de casos utilizados para los dominios objetivos a sido fijado a 1000 casos.

Ambas pruebas utilizaron las redes auxiliares cuyas características se presentan en las tablas A.5 y A.6.

De las figuras A.14, A.15 y A.16 podemos observar que la recuperación por parte del algoritmo de aprendizaje estructural propuesto ha podido tomar ventaja de la similitud con los dominios auxiliares. La recuperación ha sido más fuerte cuando los dominios tienen gran similitud entre sí, obteniendo mejores resultados que los presentados para la red Boblo presentado en éste apéndice.

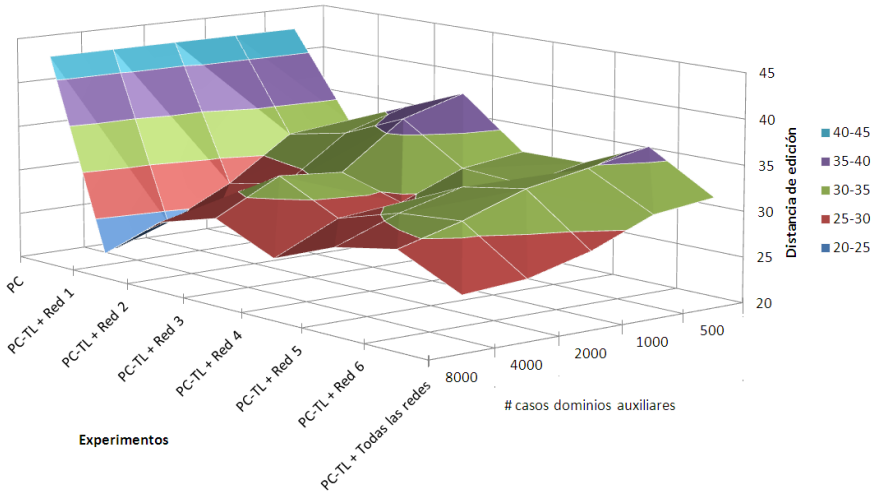


Figura A.15: Comportamiento del algoritmo PC-TL cuando se utilizan redes auxiliares cuyas características de similitud con la red objetivo varían mostrándose en el eje Z (experimentos), para la red Boblo. El eje X muestra la variación en el tamaño de la red auxiliar variando de $500 - 8000$ casos. El eje Y muestra el error medido en la distancia de edición con la verdadera estructura sobre cada prueba.

		Arcos añadidos	Arcos eliminados	Arcos revertidos	Distancia de Edición	Error Cuadrático Medio (MSE)
Red Insurance	Red Similar 1	5	5	0	10	0.015
	Red Similar 2	10	10	0	20	0.021
	Red Similar 3	12	12	0	24	0.025
	Red Similar 4	15	15	0	30	0.022
	Red Similar 5	16	16	0	32	0.060
	Red Similar 6	17	17	0	34	0.039

Tabla A.5: Características del conjunto 1 de redes utilizadas en los experimentos tipo II para la red Insurance.

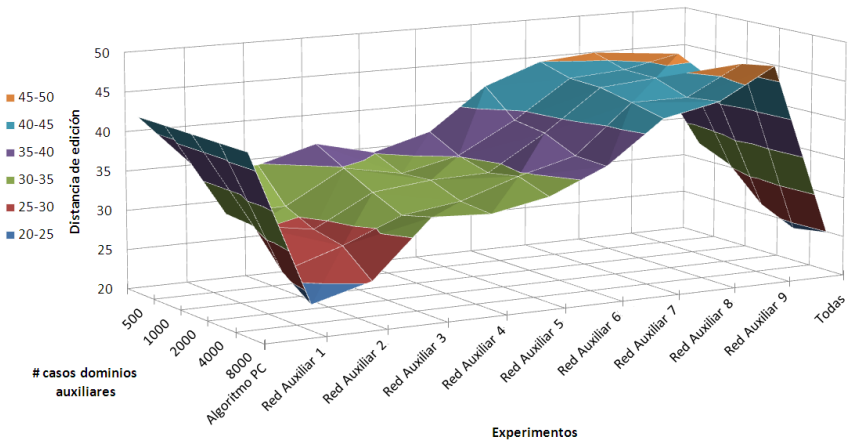


Figura A.16: Comportamiento del algoritmo cuando las redes auxiliares solo varían en la cantidad de enlaces con la red objetivo original, para la red Boblo. El eje X muestra la proporción de enlaces eliminados en la red auxiliar, el eje Z muestra la cantidad de casos utilizados en los dominios auxiliares. El eje Y muestra el error medido en distancia de edición obtenido para cada prueba. La cantidad de ejemplos utilizados para la red objetivo se mantuvo fija a 100 casos.

		Arcos añadidos	Arcos eliminados	Arcos revertidos	Distancia de Edición	Error Cuadrático Medio (MSE)
Red Insurance	Red Similar 1	5	0	0	5	0.018
	Red Similar 2	10	0	0	10	0.042
	Red Similar 3	15	0	0	15	0.062
	Red Similar 4	20	0	0	20	0.066
	Red Similar 5	26	0	0	26	0.051
	Red Similar 6	31	0	0	31	0.055
	Red Similar 7	36	0	0	36	0.059
	Red Similar 8	41	0	0	41	0.062
	Red Similar 9	46	0	0	46	0.072

Tabla A.6: Características del conjunto 2 de redes utilizadas en los experimentos tipo II para la red Insurance.