BAYESIAN NETWORK MODELING OF OFFENDER BEHAVIOR FOR CRIMINAL PROFILING

by

Kelli A. Crews Baumgartner

Department of Mechanical Engineering and Materials Science
Duke University

Date: ______________________________

Approved:

____
Silvia Ferrari, Ph.D, Advisor

____
Michael Lavine, Ph.D

____
Daniel G. Cole, Ph.D

____
Xiaobai Sun, Ph.D

A thesis submitted in partial fulfillment of the
requirements for the degree of Master of Science
in the Department of Mechanical Engineering and Materials Science
in the Graduate School of
Duke University

2005

Abstract

A Bayesian network (BN) combines probability and graph theory to map static relationships of the many variables comprising a stochastic system. This modeling techniques has been applied in engineering, particularly in mine detection and sensor management projects. Currently, Bayesian networks is applied to model human behavior, in particular criminal behavior. A Bayesian network (BN) model of criminal behavior is obtained linking the action of an offender on the scene of the crime to his or her psychological profile. Structural and parameter learning algorithms are employed to discover inherent relationships that are embedded in a database containing crime scene and offender characteristics from homicide cases solved by the British police from the 1970s to the early 1990s. A technique has been developed to reduce the search space of possible BN structures by modifying the greedy search K2 learning algorithm to include *a priori* conditional independence relations among nodes. The new algorithm requires fewer training cases to build a satisfactory model, which can be of great benefit in applications where additional data may not be readily available, such as criminal profiling. Once the BN model is constructed, an inference algorithm is used to predict the offender profile from the behaviors observed at the crime scene. The overall predictive accuracy, which refers to the total number of correct predictions, of the model obtained by the modified K2 algorithm is found to be 79%, showing a 15% improvement over the original K2 algorithm. In fact, the predictive accuracy is found to increase with the confidence level provided by the BN. Thus, the confidence level provides the user with a measure of reliability for each variable predicted in any given case. These results show that a BN model of criminal behavior could provide a valuable decision tool for reducing the number of suspects in a homicide case, based on the evidence at the crime scene.

Acknowledgements

I would like to thank my advisor Dr. Silvia Ferrari for providing me with the opportunity to work on this project and for her continued guidance.

I would also like to thank our co-collaborator on this research Dr. C. Gabrielle Salfati, a professor at the the John Jay College of Criminal Justice. She provided the database of cases for this research. This research has been submitted to the 44th IEEE Conference on Decision and Control to be held on December 12-15, 2005. The co-authors are Dr. Silvia Ferrari and Dr. Gabrielle Salfati. Additional acknowledgments go to our Italian and FBI collaborators: Roberta Bruzzone, Marco Strano, and Anthony Palermo.

Furthermore, I would like to thank my husband and parents for their continued support of my education and career.

Contents

List of Tables

List of Figures

List of Symbols and Abbreviations

Variable	Definition
α	$1/\mathcal{S}^h_{total}$
$\bar{N_{ij}}$	$\bar{N_{ij}} = \sum\limits_{k=1}^{ri} N_{ijk}$
$\emptyset$	empty set
$\mathcal{B}$	Domain of all possible BNs for the variables $\mathcal{X}$; $\mathcal{B} = (\mathcal{S}, \Theta)$
$\mathcal{B}^h$	A particular hypothesized BN from $\mathcal{B}$, where $\mathcal{B}^h \in \mathcal{B}$
$\mathcal{D}$	Set of samples of $\mathcal{X}$, also referred to as cases. Each case is an instantiation of $\mathcal{X}$, so for j total cases $\{C_1, ..., C_j\}$, C_i is an instantiation of $\mathcal{X}$
$\mathcal{F}$	The space of all possible instantiations of $X_i \in \mathcal{X}$ for $i = (1, ..., n)$
$\mathcal{P}$	Probability distributions for all $X_i \in \mathcal{X}$ w.r.t $\mathcal{S}$ and $\mathcal{F}$
$\mathcal{S}$	Domain of all the possible structures for $\mathcal{B}$
$\mathcal{S}^h$	Hypothesized structure for $\mathcal{B}^h$; $\mathcal{S}^h \in \mathcal{S}$
$\mathcal{S}^h_{total}$	Total number of $\mathcal{S}^h \in \mathcal{S}$
$\mathcal{X}$	Domain of variables
μ_i	Children of variable X_i
Ω	The DAG probability space variable; $\Omega = (\mathcal{X}, \mathcal{S})$, which is the model's variables ($\mathcal{X}$) and all the possible structures/arcs ($\mathcal{S}$)
π_i	Parents of variable X_i, pa(X_i)
Θ	Domain of all CPTs, where $\Theta = (\theta_1, ..., \theta_n)$
θ_i	$\theta_i \in \Theta$ is the CPT for each variable X_i
C_i	An instantiation of $\mathcal{X}$ for $i = (1, ..., j)$ with j total cases
F	Frequency Algorithm
K_t	Total number of predictions
$K_{C,CL}$	Number of correctly predicted variables for a specified CL

K_{CL}	Number of predicted variables for a specified CL
K_w	Number of variables inferred incorrectly
K_{ZMP}	Number of ZMP variables
m	Number of marginal probabilities in x
N_{ijk}	Number of cases in $\mathcal{T}$ for X_i is state $x_{i,k}$, where $k = (1, ..., r_i)$ and π_i is instantiated as w_{ij}
p	Number of actual *presents* for the variables in x
q_i	The number of unique instantiations for π_i
r_i	Number of possible states for variable X_i, $(x_{i,1}, ..., x_{i,ri})$
$w_{i,j}$	The j^{th} element of w_i
w_i	The particular instantiation of π_i
x	Interval from [0,1] and increments of 0.05
X_i	Each variable in $X_i \in \mathcal{X}$ for i=(1,...,n), where n is the number of variables
X_j^O	Output (OFF) variables for $j = 1, ..., k$
X_l^I	Input (CS) variables for $l = 1, ..., d$
$x_{i,j,h}$	The particular instantiation of variable i, state j and case h
$x_{i,j}$	The particular state j for state i
y	Ratio of p/m
BN	Bayesian network
CL	Confidence level, i.e. the marginal probability
CLA	The percent accuracy for a specified confidence level
CP	Criminal profile/profiling
CPT	Conditional probability table
d	Number of independent variables (CS variables= 36 total independent variables)
DAG	Directed acyclic graph

g	Scoring metric
HFM	Resulting model when the high frequency variables have been removed
i.i.d.	independent and identically distributed
IPA	An individual variable's predictive accuracy
k	n-d=k
K2′	The modified K2 structural algorithm that inhibits node connections between known variables
m	Max number of cases in $\mathcal{D}$ ($\mathcal{D} = 200$)
MDS	Multidimensional Scaling
MLE	Maximum likelihood parameter estimation
n	Number of total variables (n=57)
n	Number of variables (57 total)
nd	Non-descendants (this is in the Markov property)
OPA	The overall predictive accuracy for a model
r	Overall max r_i for i=(1,...,n), where r=2 as all variables are discrete and binary
u	Number of max parents allowed (set at 10 for this research)
ZMP	Zero Marginal Probability nodes, this is the abbreviation for the variables that have insufficient training data, thus unable to make a prediction

Chapter 1

Introduction

1.1 Background and Motivation

The scientific study of human behavior focuses on modeling underlying behavioral dynamics as a function of environment and mental processes. The behavioral model consists of a mathematical representation of the internal and external forces influencing an individual's decisions and actions in the form of a wide range of stimuli, e.g., the environment, other people, and personal psyche. The forces involved in a behavioral model are interpreted differently than forces in a traditional engineering model as behavioral forces do not follow general physical principles. Thus, the process of obtaining a human behavior model relies on a dynamically evolving system not based on a set of first principles. Thus, to obtain the actions and decisions of an individual from modeling is best done through empirical analysis of data. The empirical data available and its level of organization is growing rapidly due to recent and ongoing contributions from the information technology field. The specific behavior model addressed in this thesis is to study a genre of criminals to obtain a criminal profile.

The empirical research on criminal profiling, also known as offender profiling, so far has been limited both in scope and impact. The offender profile consists of determining the behavioral, cognitive, and emotional characteristics [27] from the signature behaviors left behind by the offender at the crime scene. The goal of criminal profiling is to concentrate a criminal investigation by narrowing the number of possible suspects and to recommend interrogation techniques [7, 18, 27]. In this thesis, a Bayesian network (BN) approach is developed for modeling an offender's behavior at the crime scene, with the purpose of predicting the offender's profile in unsolved cases.

It has been suggested [20] that offender profiling is not only possible, but also a psychologically straightforward process. Recent research has shown that it is much more complex than just a “multilevel series of attributions, correlations and predictions” [19]. In the early 1980’s, the FBI Behavioral Science Unit, the originator of the modern profile, undertook an original empirical study in the field of criminal profiling, focusing on sexual murders. This research sought to show that a correlation existed between the level of behavioral sophistication of the crime and corresponding offender characteristics. Based on the analysis of 36 incarcerated sexual murderers within North America, the outcome of this research explored the “organized/disorganized” behavior dichotomy. The *organized* offender represented the methodical, premeditated crime with corresponding offender characteristics of maturity, and resourcefulness, while the *disorganized* offender represented an opportunist likely to suffer from some mental disorder [14, 20]. Thus, a criminal profile is deduced by the investigator through categorizing the particular crime as either organized or disorganized. This research showed promise drawing conclusions of an offender from the crime scene, but advocates recommended a consistent routine for investigators to follow in the process of their investigation [31].

Later independent replications of the FBI research [15] revealed that a more realistic and utilitarian interpretation of crime behaviors was needed to develop beyond the simple dichotomy. Dr. David Canter (between the years of 1985 and 1994) focused on the search for feasible psychological principles that could be used to generate profiles and assist criminal investigations [1, 7]. This research expanded the simple organized/disorganized model into five basic aspects of the criminal transaction between the offender and victim: interpersonal coherence, significance of time and place, criminal characteristics, criminal behavior, and forensic awareness. It was suggested that future research should aim to construct a model for scientific and objective interpretation of crime behaviors and associated characteristics [14, 15]. Other critiques noted the lack of substantiation when a profile is created, such as an empirical measure to the level of confidence accompanying the predicted offender behaviors [27]. For example, an offender profiled as older in age through the categorical profiling technique is not supported by a numerical confidence level. This research did not show a distinct criminal profile but instead a categorization of similar behaviors. This did support, however, the basic assumption that similar crimes are done by similar offenders.

More recently, empirical crime scene analysis using statistical methods to understand the link between crime scene actions by an offender and his/her characteristics has shown promise [25]. The first study was based on 82 British single offender-single victim solved homicides [25], and the follow-up study was based on a larger sample (247) of single offender-single victim solved homicides [22]. Both used a statistical analysis of Multidimensional Scaling (MDS) procedure to classify cases according to specific behavioral themes: the *expressive* theme, composed of behaviors that center on the victim as a certain person, and the *instrumental* theme, centered on the benefits they obtained from the offender (e.g., either sexual or material gain). MDS is a non-metric multidimensional scaling procedure which plots the association coefficients that are calculated for each variable. Each of the points on a plot represents a crime scene behavior, and the proximity of points measures the strength of the relationships between the variables they represent. Points plotted close together have a stronger association than those plotted further apart [29]. Thus, similarly themed actions will co-occur in the same region of the plot, and variables that do not occur together will be plotted farther apart. The final results of this study classifies homicides as expressive and instrumental. A total of 62% of the cases were seen to exhibit a majority of the crime scene characteristics in a single theme, and 74% of all offenders could also be classified as either expressive or instrumental. Over half (55%) of all the 247 cases exhibited the same theme in both their crime scene actions and in the offender background characteristics [22].

A follow-up study [24] aimed at investigating the patterns of co-occurrences of crime scene actions correlating to certain offender behaviors. Based on the same 247 samples of the 2000 study, one of the conclusions was that crime scene variables that are present in more than 50% of the samples, *high frequency* variables, should not be considered when differentiating between the cases. High frequency behaviors are interpreted as typical behaviors within the cases and do not contribute to an insightful and unique view of the offender.

1.2 Research Objectives

Profiling is challenging due to many variables involved and the high degree of uncertainty surrounding a criminal act and the corresponding investigation. Probabilistic graphs are suitable modeling techniques because they are inherently distributed and stochastic. In this work, the system variables comprising the BN are offender behaviors and crime scene evidence, which are initialized by experts through their professional experience (expert knowledge). The mathematical relationships naturally embedded in a set of crimes [20, 24, 28] are learned through training from a database containing solved criminal cases. The BN model is to be applied when only the crime scene evidence is known to obtain a useable offender profile to aid law enforcement in the investigations. A criminal profile is predicted with a certain quantitative confidence.

The BN approach presented here seeks to build on the ideas of behavior correlations in order to obtain a usable criminal profile when only crime scene evidence is known from the investigation. This thesis proposes a systematic approach for deriving a multidisciplinary behavioral model of criminal behavior. The proposed offender behavioral model is a mathematical representation of a system comprised of an offender's actions and decisions at a crime scene and the offender's personal characteristics. The influence of the offender traits and characteristics on the resulting crime scene behaviors is captured by a probabilistic graph or BN that maps cause-and-effect relationships between events, and lends itself to inductive logic for reasoning under uncertainty [5]. The use of BNs for criminal profiling (CP) may allow investigators to take into consideration various aspects of the crime and discover behavioral patterns that might otherwise remain hidden in the data. The various aspects of a crime include a victimology assessment (victim's characteristics, e.g., background characteristics, age, gender, and education), crime scene analysis (evidence from the crime scene, e.g., time and place the crime occurred), and a medical report (autopsy report, e.g., type of non-deadly and deadly lesions and signs of self defense).

The BN approach to criminal profiling is demonstrated by learning from a series of crime scene and offender behaviors. The learning techniques employed in this modeling research are evaluated on a set of validation cases not used for training by defining a prediction accuracy based on the most likely value of the output variables (offender profile) and its corresponding confidence level.

1.3 Thesis Organization

The topics addressed in this thesis begin with a background of Bayesian networks and an introduction of notation to be used throughout the thesis. The following Criminal Profile Modeling chapter is divided into five sections. Section 3.1 formulates the criminal profiling problem with respect to this criminal profile research, and Sections

3.2-3.5 describe the offender variables and the set of cases used in this research. Chapter 4 details the learning and prediction methods implemented to obtain the offender model, while Chapter 5 outlines the application of BN learning and inference and the overall model evaluation. Additional details are given in the Appendices.

Chapter 2

Bayesian Networks

2.1 Introduction to Bayesian Networks

Bayesian networks (BNs) combine probability and graph theory in order to extract knowledge of a given system from empirical data by mapping cause-and-effect relationships among all relevant variables. By using conditional probabilities, they can capture the extent to which variables are likely to affect each other, even if the underlying mechanisms are unknown. The causal relationships between variables and events are learned from an ensemble of known cases where all the variables are known or measured. Then, the BN model obtained can be used in new cases to infer missing variables from known ones. Through this inference mechanism, the BN computes the likelihood, or probability, that an unknown variable will take any of its possible values [17].

A BN is a directed graph consisting of a set of variables or *nodes* (or events) and a set of *directed* arc or edges between variables [12]. The nodes together with the directed arcs form a *directed acyclic graph* (DAG). A variable represents a set of countable states of affairs. It can be an event, a proposition, or a mathematical quantity. Each arc represents a causal dependency among the nodes it connects. Each arc has a strength associated with it that is stored in a *conditional probability table* (CPT) attached to each node. Figure 2.1 shows a simple example of Bayesian network, depicting four variables ("Cloudy", "Sprinkler on", "Rain", and "Wet grass") and their causal influence. For the grass to be wet (W), it is caused by rain (R) and the sprinkler (S), both of which are affected by a cloudy day (C). The CPT attached to the node S corresponds to the variable "sprinkler on". Suppose the relevant possibilities for S are that on a given day it was on at one point ($S = True$) or that it was not on ($S = False$), and that the presence or absence of clouds (C) influences the value of S. Then, C is said to be a parent of S, and the strength of their relationships is expressed by the CPT attached to the node S. This table contains the conditional probability $p(S|C)$ for all possible values of the variables C and S. For instance, the probability that the sprinkler was on given that it was not a cloudy day, $P(S = T|C = F)$, is 0.9. If it is observed that the grass is wet, $W = T$, it can be *inferred*, or predicted, the cause was either the sprinkler, rain, or both. With the observation of $W = T$, the unknown variables S and R will be predicted with a certain predictive probability, or *confidence*. From here, it is possible to infer the presence or absence of a cloudy day.

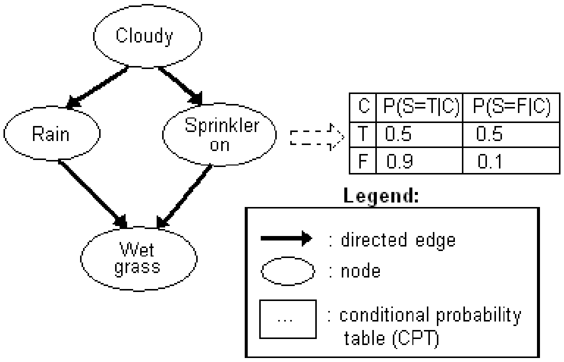

Figure 2.1: Example of a Bayesian network in which the events “cloudy”, “sprinkler on”, “rain”, and “wet grass” are displayed in the form of a DAG where T=True and F=False, source: [17]

While the experts’ knowledge and experience can be used to initialize the graph, the actual arcs (relationships) and probabilities are learned from an ensemble of real cases, where all the variables are known from observation. Later, as new cases become available, they also can be incorporated to refine the graph’s structure and CPTs. Using basic probability theory, one can use the BN model to understand the relationships between the variables. For example, if two nodes are disconnected, they are independent of each other (e.g., sprinkler on and rain from Figure 2.1). This is also referred to as d-separation. If two variables are found to be weakly connected, the arc between them will have very small probabilities. Another example is that if two nodes are connected only through a third node (e.g., cloudy and wet grass in Figure 2.1), they become independent when the intermediate variable or variables (e.g., sprinkler on and rain) is known. Another important property addressing conditional independence is the *directed Markov property*, which states that a variable is conditionally independent of its non-descendants given its parents [5].

It can be seen from this simple examples that the BN structure can be applied to that of behavioral patterns to gain insight into what factors influence certain human behaviors. Another important feature is that, once the structure and CPTs are learned, the BN can be used for inference. This means that if a new case is being investigated and some variables are unknown as they are unobservable, an inference algorithm can be used to obtain a prediction as to the most likely value of the variable, as well as the level of uncertainty associated with it.

2.2 Bayesian Network Notation and Theory

In this thesis, capital letters denote variables and lowercase letters denote the *states* or instantiations of the variables (i.e., X_i is said to be in its j^{th} instantiation when $X_i = x_{i,j}$). A variable or *node* in a BN corresponds to each item in a domain $\mathcal{X} = (X_1, ..., X_n)$ for $n > 1$ discrete variables in the probability space $\{\Omega, \mathcal{F}, \mathcal{P}\}$. The probability space of a BN refers to a structure or graph $\Omega = \{\mathcal{X}, \mathcal{S}\}$, where $\mathcal{S}$ is the set of directed arcs (denoted by arrows) between the variables $\mathcal{X} = (X_1, ..., X_n)$. The variables and *directed* edges of Ω together comprise a *graph*, referred to as a *directed acyclic graph* (DAG) [12]. The BN parameter $\mathcal{F}$ is the space of all possible instantiations of X_i, for $i = 1, ..., n$. $\mathcal{P}$ is the probability distribution for all X_i with respect to $\mathcal{S}$ and $\mathcal{F}$.

A Bayesian network is a directed graphical model combining probability and graph theory. Let $\mathcal{B}$ be the set of all possible BNs, $\mathcal{B} = (\mathcal{S}, \Theta)$, where $\mathcal{S}$ is the DAG with parameters $\Theta = (\theta_1, ..., \theta_n)$ and $\Theta \in \mathcal{P}$. An example of the directed graphical structure of a simple BN is in Figure 2.2. The parameter $\theta_i \in \Theta$ is the *conditional probability table* (CPT) attached to node X_i. A CPT lists in tabular form the conditional probabilities of each state of X_i with respect to each of its parents, $P(X_i|\pi_i)$, where π_i represents the parents of X_i. If a node has no parents ($\pi_i = (\emptyset)$) , the CPT for θ_i is simply a prior probability distribution $P(X_i)$. Every X_i has a CPT that is either initialized by a user from prior knowledge or learned from the set of training cases, described in detail in Chapter 4. A *sample* over $\mathcal{X}$ is an observation for every variable in $\mathcal{X}$. A database $\mathcal{D}$ is a compilation of d samples of $\mathcal{X}$, $\mathcal{D} = \{C_1, ..., C_d\}$. $\mathcal{D}$ is said to have no *missing values* when all values of all variables are known. An assumption is made that each individual sample C_i is independent and identically sampled (i.i.d.) with an underlying unknown distribution.

A BN is a mathematical model based on the acquired data and the implementation of Bayes rule for inference when a variable (or variables) is (are) unknown given observations for the other variables [12, 10]. Bayes' rule of dependence can be utilized to calculate the posterior probability distribution of X_i given the instantiations of X_i's children, represented as μ_i, as follows

$$P(X_i|\mu_i) = \frac{P(\mu_i|X_i)P(X_i)}{P(\mu_i)}. \tag{2.1}$$

The prior probability of X_i, $P(X_i)$, is the known probability distribution over the states of X_i, $(x_{i,1}, ..., x_{i,ri})$, and is considered a known relationship either by previous experience, testing, or observation. The likelihood function, $P(\mu_i|X_i)$, contains the conditional probabilities of the instantiated children variables connected to X_i. Similar to the prior probability, the likelihood function is obtained from prior observations or subjectively estimated by the user through experience. In this case, it becomes the product of the likelihood probabilities of the instantiated variables $P(\mu_i|X_i) = \prod_{j=1}^{p} P(\mu_{i(j)}|X_i)$, where $\mu_{i(j)}$ is the instantiation of the j^{th} child of X_i. The marginalization of the observed variables, $P(\mu_i)$, accounts for the relationship between the instantiated variables and all possible states of X_i as follows

$$P(\mu_i) = \sum_{k=1}^{ri} P(X_i = x_{i,k}) \prod_{j=1}^{p} P(\mu_{i(j)}|X_i), \tag{2.2}$$

where $\mu_{i(j)}$ is the j^{th} instantiated variable of X_i's p total children. The posterior probability of $X_i = x_{i,k}$, denoted by $P(X_i = x_{i,k}|\mu_i)$, is also known as the marginal probability of $x_{i,k}$ and represents its confidence as a probability for which to occur given the evidence. Predicting an unknown variable to be in a certain state based on the evidence of the observed variables from Bayes' Theorem is *inference.* X_i is inferred from μ_i using (2.1).

A simple BN consisting of three-nodal diverging causal relationships in Figure 2.2 is used to illustrate Bayes' rule for purposes of inferring an unknown variable based on evidence. In causal BNs generally the "causes" are the parent nodes and the "effects" are the children nodes. The parent node X_1 has r_1 states $(x_{1,1}, ..., x_{1,r1})$, and the children nodes X_2 and X_3 have the possible set of instantiations $(x_{2,1}, ..., x_{2,r2})$ and $(x_{3,1}, ..., x_{3,r3})$, respectively. Thus, the parent variable π_i becomes $\pi_2 = \pi_3 = (X_1)$ as X_1 is the parent of variables X_2 and X_3, and likewise the children variable μ_j becomes $mu_1 = (X_2, X_3)$. If it is observed that $X_2 = x_{2,r2}$ and $X_3 = x_{3,r3}$, the posterior probability distribution of X_1 given the evidence, (2.1) is as follows

$$P(X_1|x_{2,r2},x_{3,r3}) = \frac{P(x_{2,r2},x_{3,r3}|X_1)P(X_1)}{P(x_{2,r2},x_{3,r3})}. \tag{2.3}$$

The prior probability of X_1, $P(X_1)$, is the known probability distribution over the states of X_1, $(x_{1,1},...,x_{1,r1})$. The likelihood function, $P(x_{2,r2},x_{3,r3}|X_1)$, is the conditional probabilities of the instantiated variables X_2 and X_3 connected to X_1, which becomes the product of the likelihood probabilities of the instantiated variables $P(x_{2,r2},x_{3,r3}|X_1) = \prod_{j=2}^{3} P(x_{j,rj}|X_1)$. The marginalization of the observed variables, $P(x_{2,r2},x_{3,r3})$, accounts for the relationship between the instantiated variables and all possible states of X_1, and (2.4) becomes

$$\begin{aligned} P(x_{2,r2},x_{3,r3}) =& P(x_{1,1})P(x_{2,r2}|x_{1,1})P(x_{3,r3}|x_{1,1}) + ...+ \\ & P(x_{1,r1})P(x_{2,r2}|x_{1,r1})P(x_{3,r3}|x_{1,r1}). \end{aligned} \tag{2.4}$$

The distribution $P(X_1|x_{2,r2},x_{3,r3})$ is consistent with the the probability distribution axiom $\sum_{v=1}^{r1} P(X_1 = x_{1,v}|x_{2,r2},x_{3,r3}) = 1$. The *inference* of X_1 is the prediction of the state of X_1 from its posterior distribution from the observation of X_2 and X_3. The posterior probability of $X_i = x_{i,j}$, denoted by $P(X_i = x_{i,j}|\mu_i)$, is also known as the marginal probability of $x_{i,j}$ and represents its confidence as a probability for which it occurs given the evidence. Hence, the state of X_1 is predicted from the maximum marginal probability, as this is seen as the most likely state given the uncertainty.

The exact computation of the marginal probabilities of a BN is often too computationally expensive [2, 8]. Constructing an inference engine allows for a more tractable procedure to calculate the marginal probabilities in the BN [5]. Efficient inference engines identify the conditional independencies between the variables in a system in order to simplify computation, described in detail in the previous section. Typically,

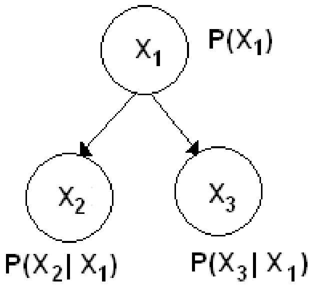

Figure 2.2: Example of simple diverging connect BN

identifying these relationships from conditional independent properties (i.e., directed Markov property and *d*-separation) simplifies the inference procedure. Here, it also is exploited to simplify structural learning to obtain the so-called K2′ algorithm, discussed in Chapter 4.

Potential conditional independencies among nodes must be identified so that inference of unknown variables can be completed when evidence becomes available. For this reason, the inference engine is compiled through steps of graphical manipulations that transform a DAG into a junction tree. The junction tree is a moralized, triangulated, and undirected graphical representation of the original BN structure [5, 4] in which all of the conditional independencies among the variables are recognized. An arc (also an edge) refers to the parent/child relationship between two variables in the form of an arrow, leading from the parent to the child variable. An *undirected* arc is simply a line relating two variables. A graph is *moralized* when undirected arcs are added to all co-parents not previously joined, and all current directed arcs become undirected. *Triangulation* refers to the acyclic property in which additional undirected edges are added between nodes to assure that there are no cycles. The final step in building the junction tree for graphical manipulation is to identify and join the BN structure's cliques, i.e., the unique path between two or more variables. Once the junction tree compilation is complete, all conditional relationships among variables are connected by an undirected edge. If two variables are not connected, then they are conditionally independent and the instantiation of one does not affect the inference of the other.

A graphical model whose edges are initially undirected is an undirected graphical models, which is also called Markov Random Fields (MRFs) or Markov networks [16, 13]. A major difference between directed and undirected graphical models is the differing method for identifying the variables' conditional independencies. An undirected graphical model has a simple definition of independence in which two variables are deemed conditionally independent if the are separated by a known variable. However, directed graphical models, also called Bayesian networks or belief networks (BNs), take into account the directionality of the arcs. For a directed graphical model, two variables are deemed conditionally independent if two variables are separated by a known variable in an equivalent undirected graphical model, which is an undirected and moralized version of the directed graph. To obtain this equivalent structure, the directed edges are first replaced with undirected edges, and undirected edges are then added between parents who share a common child (i.e., "moralize" the graph) to prevent identifying incorrect independence statements.

A good example of an undirected graphical structure is Figure 2.3a, where X_1 and X_4 are conditionally independent from each other given X_2, as are X_2 and X_3 given X_1. However, conditional independence is not recognized as easily for the directed graphical structure depicted in Figure 2.3b, which becomes the undirected and moralized structure in Figure 2.3c. Figure 2.3c differs from the original undirected graph in Figure 2.3a by the additional edge relating X_4 and X_1. The conditional independencies for the directed graph are acquired from the undirected and moralized graphical model. The variables X_2 and X_3 are conditionally independent if X_1 is known, same as the undirected graph in Figure 2.3a. However, X_4 and X_1 are conditionally *dependent* for a directed graphical structure, which is different than the undirected graph in Figure 2.3a. Although directed models have a more complicated notion of independence than undirected models, they also have several advantages including simpler training methods and the ability to encode deterministic relationships. Directed models are more popular with the AI and statistics communities, while undirected models are more popular with the physics and vision communities [16].

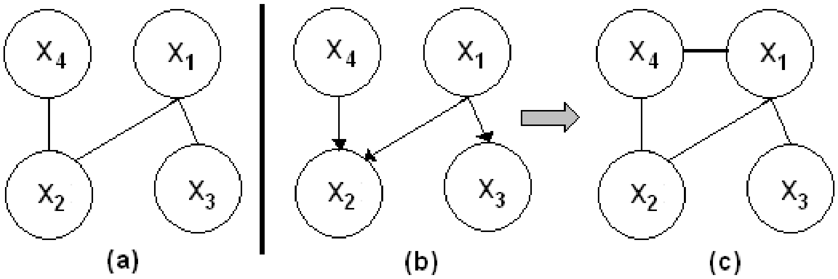

Figure 2.3: Example of an undirected graphical model (a), a directed graphical model (b), and the equivalent undirected and moralized graphical model (c)

Chapter 3

Criminal Profile Modeling

3.1 Problem Formulation

Currently, a *criminal profile* (CP) is obtained from an investigator's or forensic psychologist's interpretation linking crime scene characteristics and an offender's behavior to his or her characteristics and psychological profile. This research seeks an efficient and systematic discovery of non-obvious and valuable patterns between variables from a large database of solved cases via a Bayesian network (BN) modeling approach. The BN structure can be used to extract behavioral patterns and to gain insight into what factors influence these behaviors. Thus, when a new case is being investigated and the profile variables are unknown because the offender has yet to be identified, the observed crime scene variables are used to infer the unknown variables based on their connections in the structure and the corresponding numerical (probabilistic) weights. The objective is to produce a more systematic and empirical approach to profiling, and to use the resulting BN model as a decision tool.

A graphical model of offender behavior is learned from a database of solved cases. The database for this research is from the British police forces and were completed by the investigator at the conclusion of an investigation. The resulting CP model obtained through training is then tested by comparing its predictions to the actual offenders' profiles. The database $\mathcal{D}$ containing d solved cases $\{C_1, ..., C_d\}$, where C_i is an instantiation of $\mathcal{X}$, is randomly partitioned into two independent datasets: a *training set* $\mathcal{T}$ and a *validation set* $\mathcal{V}$, such that $\mathcal{D} = \mathcal{T} \cup \mathcal{V}$. The variables $\mathcal{X}$ are partitioned as follows: the *inputs* are the *crime scene* (CS) variables X^I (evidence) for $X^I = (X^I_1, ..., X^I_k)$, and the *outputs* are the *offender* (OFF) variables comprising the criminal profile X^O for $X^O = (X_1^O, ..., X_m^O)$, where $(X^I, X^O) \in \mathcal{X}$.

The BN model is learned from $\mathcal{T}$, as explained in Chapter 4, and it is tested by performing inference to predict the offender variables (OFF) in the validation cases $\mathcal{V}$. An offender profile is estimated based on crime scene evidence, with a prediction being the *most likely* value of a particular offender variable. During the testing phase, the predicted value of X_i^O, denoted by $x_{i,a}^P$ where a=1 or 2 for a binary variable, is compared to the observed state $x_{i,b}^O$ obtained from the validation set $\mathcal{V}$, where b=1 or 2. An example of an offender variable is “gender”, with states “male” and “female”. The overall performance of the BN model is evaluated by comparing the true (observed) states $x_{i,b}^O$ to the predicted output variable values $x_{i,a}^P$ in the validation cases. This process tests the generalization properties of the model by evaluating its efficiency over $\mathcal{V}$.

The basic schematic of the training software, including the validation process, is shown in Figure 3.1, where $\mathcal{B}^h$ is the proposed BN and $\mathcal{B}^{opt}$ is the trained (or optimized) BN. The software is intended to aid law enforcement in the investigation of violent crimes. Because the cases are unsolved and only the crime scene inputs are known, the criminal profiling software consists of a trained BN model that has been previously trained and validated with $\mathcal{D}$. Also, the model has the potential to be updated by means of an incremental training algorithm when additional cases are solved by the police. Thus, $\mathcal{B}^{trained}$ consistently reflects the model of an evolving criminal profile over time.

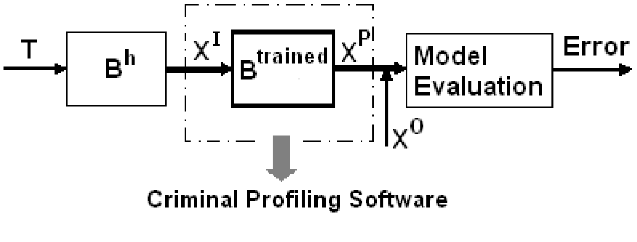

Figure 3.1: Diagram of the CP model training and validation software

3.2 Variables

The relevant categories of variables that have emerged from the criminal profiling research as selected by investigators, criminologists, and forensic psychologists are described as follows:

- **Crime Scene Analysis (CSA):** CSA variables are systematic observations made at the crime scene by the investigator. Examples of CSA variable pertain to where the body was found (e.g., neighborhood, location, environment characteristics), how the victim was found (e.g., the body was well-hidden, partially hidden, or intentionally placed for discovery), and the correlation between where the crime took place and where the body was found (e.g., the body was transported after the murder).

- **Victimology Analysis (VA):** VA variables consist of the background characteristics of the victim independent of the crime. For example, VA variables include the age, sex, race, education level, and occupation of the victim.

- **Forensic Analysis (FA):** FA variables rely on the medical examiner's report that deals with the autopsy. Examples of this are time of death, cause of death, type of non-lethal wounding, wound localization, and type of weapon that administered the wounds.

The set of CP variables used in this research were acquired from police reports of homicide crime scenes and were defined in previous research [28, 23, 26, 22, 24]. The selection criteria for variable selection [24] are: (*i*) behaviors are clearly observable and not easily misinterpreted, (*ii*) behaviors are reflected in the crime scene, e.g., type of wounding, and (*iii*) behaviors indicate how the offender acted toward and interacted with the victim, e.g., victim was bound/gagged, or tortured. 36 crime scene

(CS) variables describing the observable crime scene and 21 offender (OFF) variables describing the actual offender were selected based on the above criteria. Examples of the CS variables are multiple wounding to one area, drugging the victim, and sexual assault. Examples of the offender variables include prior offenses, relationship to the victim, prior arrests, etc. The variables all have binary values representing whether the event was present or absent.

3.3 Database of Solved Cases

A set of single offender/single victim homicides was collected by psychologists from solved homicide files of the British police forces around the UK from the 1970s to the early 1990s. This same data was also used in criminal profiling research [22, 24]. In order to examine the aggressive behavioral patterns of a particularly violent offense, the criteria for case selection is: single offender/single victim homicide cases; a mixture of *domestic* (where the victim and offender were known to each other (e.g., family member, spouse, co-worker) and *stranger* (the offender is unknown to the victim, thus they had no previous links to each other) cases; offenders are adults at least 17 years of age, as defined by the court system. Excluded from the sample were cases when the cause of death was not aggressive or extremely intentional. Homicides by reckless driving are not included due to the lack of interpersonal interaction between the offender and victim. Also excluded was murder clearly done by professional hitmen and euthanasia.

3.4 Sample Demographics

In these 247 sample cases, the majority of the victims were female (56%) with a mean age of 41 years, ranging from 0 to 93. Male victims (44%) had a mean age of 39 years, ranging from 0 to 82. The offenders in this sample were predominantly male (89%) with a mean age 32 years ranging from 16 to 79. The female offenders (11%) had ages ranging from 17 to 70, with a mean age of 33 years. Only 15% of the cases were considered sex crimes and only 9% of the offenders had prior sexual convictions. As for the victim/offender relationships, 10% of the victims were related to their offender (either by blood or otherwise) and 43% of the victims had a previous sexual relationship with the offender (excluding cases of prostitution). A total of 83% of the offenders knew the victim in some capacity prior to the offense.

3.5 Sampling

In order to study the BN learning and inference capabilities, a more extensive list of crime scene and offender characteristics, including multiple-valued variables, a simulation is built to produce an artificial CP database. A BN is used to simulate a set of cases where the crime scene and offender variables can be chosen by the user. An initial structure $\mathcal{S}_o$ relating the variables and the corresponding initial probabilistic parameters Θ_o are declared by expert criminologists and investigators (our collaborators, see Acknowledgements) based on their prior knowledge, through experience, or by sampled statistics cited in literature from actual cases (similar to the sample demographics in Section 3.4). Cases are simulated by *feedforward* sampling, where variables are sampled one at a time in order from *top-level* variables (variables without parents), to the *mid-level* variables (variables with both parents and children), ending with the *bottom-level* variables (children variables with parents only). For each variable, the discrete conditional prior probabilities in vector form, $[P(x_{i,1}|\pi_i), P(x_{i,2}|\pi_i), ..., P(x_{i,r_i}|\pi_i)]$, where r_i is the maximum state for X_i and π_i disappears if X_i is a top-level variable, represent ranges of occurrence for each state. A value v_i is drawn from a uniform continuous distribution between $[0, 1]$, and the conditional prior probability vector as a vector of ranges becomes

$[P(x_{i,1}|\pi_i), P(x_{i,1}|\pi_i) + P(x_{i,2}|\pi_i), ..., \sum_{j=1}^{r_i} P(x_{i,j}|\pi_i)]$, which refers to

$$X_i = \begin{cases} x_{i,1} \texttt{ if } 0 < v < P(x_{i,1}|\pi_i) \\ x_{i,2} \texttt{ if } P(x_{i,1}|\pi_i) \leq v < \sum_{j=1}^{2} P(x_{i,j}|\pi_i) \\ \vdots \\ x_{i,ri} \texttt{ if } \sum_{j=1}^{r_i} P(x_{i,j}|\pi_i) \leq v < 1 \end{cases} \tag{3.1}$$

To simulate a set of cases for the system represented by the three-nodal model in Figure 2.2, the variables are ordered as (X_1, X_2, X_3), where X_1 is the parent of X_2 and X_3, and $X_1 = (x_{1,1}, x_{1,2})$, $X_2 = (x_{2,1}, x_{2,2})$ and $X_3 = (x_{3,1}, x_{3,2})$. Starting with X_1, it has three possible states with the prior probabilities $P(x_{1,1}) = 0.2$, $P(x_{1,2}) = 0.5$, and $P(x_{1,3}) = 0.3$, which becomes a range vector $[0.2, 0.7, 1]$ referring to

$$X_1 = \begin{cases} x_{1,1} \texttt{ if } 0 \leq v_1 < 0.2 \\ x_{1,2} \texttt{ if } 0.2 \leq v_1 < 0.9 \\ x_{1,3} \texttt{ if } 0.9 \leq v_1 < 1 \end{cases} \tag{3.2}$$

If $v_1 = 0.11$ which makes $X_1 = x_{1,1}$, and the CPT for X_2 is listed in Table 3.1, then the conditional prior probability vector of ranges for a newly generated v_2 becomes

$$X_2 = \begin{cases} x_{2,1} \texttt{ if } 0 \leq v_2 < 0.2 \\ x_{2,2} \texttt{ if } 0.2 \leq v_2 < 1 \end{cases} \tag{3.3}$$

X_3 sampled following the same procedure as X_1 and X_2. This is repeated until the desired number of cases as specified by the user is reached. The Matlab function

Table 3.1: An example of a CPT for X_2 in Figure 2.2.

X_2	$P(x_{2,1}\|X_1)$	$P(x_{2,2}\|X_1)$
$X_1 = x_{1,1}$	0.2	0.8
$X_1 = x_{1,2}$	0.9	0.1

utilized for the sampling exercise is *sample_bnet* in the *Bayes Net Toolbox* [17].

Chapter 4

Learning Methods

Since the recent development of efficient inference algorithms [11, 8], BNs have become a common representation tool in computer science. They also are useful for control and decision making because they can model stochastic processes from data. A BN allows for deterministic interpretation of events in which predictions of intervention are made with some unknown information. A set of probabilistic Bayesian networks $\mathcal{B}$ can be constructed given a database containing the instantiation of a set of variables and an implicit assumption about the variables' characteristics and interactions with each other. A learning framework is used to obtain the network that "best" describes the database.

Ideally, if $\mathcal{B} = (\mathcal{S}, \Theta)$ denotes the set of all possible BNs with nodes $\mathcal{X}$ reflecting the variables in $\mathcal{D}$, then the compatibility of all DAGs with $\mathcal{T}$ would be compared pairwise. The compatibility of each hypothesized structure, $\mathcal{S}^h \in \mathcal{S}$, with the training data is assessed by a so-called scoring metric that assigns a value, or *score*, to each $\mathcal{S}^h$ given by the conditional probability of $P(\mathcal{S}^h|\mathcal{T})$ [2, 11, 10]. The *best* score is the maximum conditional probability of $\mathcal{S}^h$ given the training data $\mathcal{T}$, i.e., $\max P(\mathcal{S}^h|\mathcal{T})$. Since the calculation $P(\mathcal{S}^h|\mathcal{T})$ is computationally infeasible, it is recognized that because $P(\mathcal{D})$ is independent of $\mathcal{S}^h$, a more feasible calculation is the joint probability $P(\mathcal{S}^h, \mathcal{T})$ [2] (See Appendix B for the proof that $P(\mathcal{S}^h|\mathcal{T}) \propto P(\mathcal{S}^h, \mathcal{T})$). Thus, the scoring metric becomes a joint probability calculation, where the joint probability distribution is given by

$$P(\mathcal{S}, \mathcal{T}) = \int_{\Theta} f(\mathcal{T}|\mathcal{S}, \Theta) f(\Theta|\mathcal{S}) P(\mathcal{S}) d\Theta, \tag{4.1}$$

where $f(\mathcal{T}|\mathcal{S},\Theta)$ is the conditional probability density function over $\mathcal{T}$ given $\mathcal{S}^h$ and Θ^h; $f(\Theta|\mathcal{S})$ is the conditional probability density function over Θ^h given $\mathcal{S}^h$; $P(\mathcal{S})$ is the prior probability of $\mathcal{S}^h$. With the following assumptions [2], the computation of (4.1) becomes tractable: (*i*) all variables are discrete; (*ii*) all structures are equally likely, $P(\mathcal{S}) \sim i.i.d.$ Uniform(α); (*iii*) all cases in $\mathcal{D}$ occur independently given a BN model; (*iv*) all variables are known with no cases that have missing variables; (*v*) no prior knowledge of the numerical properties to assign to $\mathcal{B}^h$ with structure $\mathcal{S}^h$ before observing $\mathcal{T}$. With assumptions (*i*-*v*), the scoring metric becomes a joint-probability scoring metric [2] that can be simplified as follows (see Appendix B)

$$P(\mathcal{S}^h,\mathcal{T}) = P(\mathcal{S}^h)\cdot\prod_{i=1}^{n}\prod_{j=1}^{q_i}\frac{(r_i-1)!}{(\bar{N}_{ij}+r_i-1)!}\prod_{k=1}^{r_i}N_{ijk}!, \tag{4.2}$$

where n discrete variables in $\mathcal{X}$ each have r_i possible states $(x_{i,1},...,x_{i,r_i})$, q_i is the number of unique instantiations for π_i, N_{ijk} is the number of cases in $\mathcal{T}$ where $X_i = x_{i,k}$, and $\bar{N}_{ij} = \sum_{k=1}^{r_i} N_{ijk}$. $\mathcal{S}^h$ is encoded as a discrete variable whose state corresponds to the set of possible network structures in $\mathcal{B}$ and assesses the probabilities $P(\mathcal{S}^h)$. Since (4.2) depends on the relative compatibility of the hypothesized structure with the data and the goal is to find $\mathcal{S}^h$ with maximum score, the scoring metric is maximized with respect to $\mathcal{S}^h$.

The number of possible structures grows exponentially as a function of the number of nodes [21]. Thus, a more feasible search algorithm is needed to systematically limit the search space in order to find a suitable local optimized structure, $\mathcal{S}^{trained}$, for a domain of variables $\mathcal{X}$. Incremental search methods have been developed to minimize the search field using a scoring function similar to (4.2). A typical search algorithm works by adding an arc where one is absent, eliminating an arc if one is present, scoring the new structure, and then continuing to the next structure. Arcs can be removed, reversed, or added, so long as they satisfy the acyclic property. Following the notation introduced in [9], all eligible changes to a graph (i.e., arc additions, reversals, or eliminations) are denoted by E and a specific graph, in the set E, is denoted by e.

A popular search method is the greedy search algorithm [2, 9], which exploits an assumption of node ordering for $\mathcal{X}$, and allows only causal arcs in the forward path eliminating arc reversals from the search space. Typically in BN learning, variables are assigned a particular order (e.g., $\{X_1, X_2, ...\}$. Node ordering does not always prevent a variable from being the child of a succeeding variable (e.g., X_2 can be the parent of X_1). However, in the K2 algorithm, a stricter interpretation of node ordering is implemented in order to decrease the search space of $\mathcal{S}^h$. It is assumed that directed edges only can be replaced from preceding to succeeding variables. This procedure allows the designer to use expert knowledge to eliminate arc reversals between selected variables. Hence, if X_1 precedes X_2 an arc reversal from X_2 to X_1 is excluded *a priori*. The greedy search algorithm begins with an initial graph structure $\mathcal{S}_o$, which is either known, empty, or random [9], and searches for the maximum $\Delta(e)$ for all $e \in E$, where $\Delta(e)$ is the change in the log score of the modified network. The log score is implemented because of its monotonically increasing characteristic that is computationally more efficient. This algorithm does not guarantee to find the structure with the highest probability, but it systematically reduces the computationally infeasible search space and, at the same time, maximizes the scoring function. Random restarts are introduced to avoid local maxima.

A greedy search algorithm, referred to as the heuristic search K2 algorithm [2], is one method explored in this research. The following simplifying assumptions are added to (*i*-*v*): (*vi*) ordering of nodes, and (*vii*) limited number of parents per node.

These assumptions lead to a simplified score, from (4.2),

$$g = \log \left(\prod_{j=1}^{q_i} \frac{(r_i - 1)!}{(\bar{N}_{ij} + r_i - 1)!} \prod_{k=1}^{r_i} N_{ijk}! \right) . \tag{4.3}$$

The complexity of the K2 algorithm is significantly less than the complexity of an exhaustive search. The function g in (4.3) is $\mathcal{O}(mur)$, where m is the maximum number of cases in $\mathcal{T}$, u is the maximum number of parents allowed per node, and $r = \max_{1 \leq i \leq n} r_i$. When this function is called at most $n - 1$ times, it requires $\mathcal{O}(munr)$ computation time. Each of the total nodes n is limited to a maximum of u parents leading to a computation time of $\mathcal{O}(un)$. The resulting complexity of the K2 algorithm with a bound on the maximum number of parents is $\mathcal{O}(mu^2n^2r)$ [2].

The second learning method used in this research further reduces the computational complexity of (4.2) (while still maintaining a suitable search space) by introducing an additional assumption of input independence. The purpose of learning a BN is to use the trained BN to infer variables that are non-observable from the values of the observable variables. If it is known prior to learning that a set of nodes *always* will be instantiated (always observed) during the inference process, independence among these variables can be established. These conditional independence relationships are illustrated by the BN in Figure 4.1. Since X_4 has influence on X_1 which in turn has influence on X_2 and X_3, then evidence on X_2 and X_3 will effect the inference of both X_1 and X_4. However, if X_1 is known, this instantiation blocks communication to its parent and children respectively: X_4 is said to be *d-separated* from X_2 and X_3 [12]. Similarly, if it is known prior to learning that X_1 and X_4 are always instantiated and never inferred, then regardless of the connection between the X_1 and X_4, these variables are always graphically conditionally independent of each other. This statement is derived from the property of admittance of d-separation in BNs, which states that if two variables X_4 and X_2 are d-separated in a BN with evidence e for X_1, then $P(X_2|X_4, X_1) = P(X_2|X_1)$ [12]. Inhibiting certain node connections prior to learning eliminates a subset of potential BNs and, thus, increases the efficiency of the greedy search algorithm. This conditional independence assumption is insufficient if the data is incomplete. Hence, it should be used only for those nodes that will be instantiated by the observations. Further explanation of validating inhibiting nodal connections among observed variables is found in Appendix B.

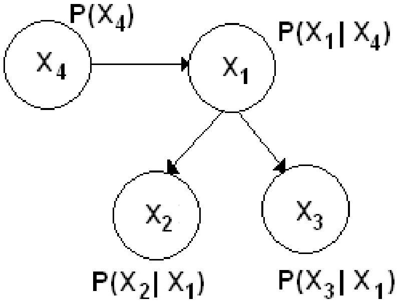

Figure 4.1: Inserting variable X_4 as a parent to another input variable showing independence between X_1 and X_4 if they are both instantiated

In this thesis, the modified K2 algorithm, where a particular set of arcs between variables (i.e., input) is blocked *a priori*, is referred to as K2′. The complexity of the K2′ algorithm is significantly less than the complexity of the K2. The K2 and K2′ require the same time $\mathcal{O}(mur)$ for computing the function g in (4.3) $n-1$ times, leading to time $\mathcal{O}(munr)$. The computational expense for analysis of the maximum of u parents in K2 looping over all $\mathcal{X}$ (n times) is $\mathcal{O}(un)$. With the conditional independence assumption among input variables, the computation time is reduced from $\mathcal{O}(un)$ to $\mathcal{O}(uk)$, where $k = n - d$, and d is the number of variables that are independent of each other. Thus, the overall complexity of K2′, $\mathcal{O}(mu^2nkr)$ time, significantly decreases as the number of independent variables, d, increases. The K2 algorithm function is called in Matlab by the *learn_struct_k2* function in the Bayesian Network Toolbox [17] and is compared to the K2′ algorithm in support of the above assumption. In addition to reducing the computational expense, the K2$'$ algorithm shows an improvement in model accuracy for the validation data $\mathcal{V} \in \mathcal{D}$.

The *maximum likelihood parameter estimation* (MLE) [9] procedure is implemented to obtain Θ^h for a given $\mathcal{S}^h$ and $\mathcal{T}$. MLE begins with a mathematical expression known as the *likelihood function* of the sample data. The density function $f(\mathcal{T}|\Theta^h)$ is the probability distribution for the set of training cases given the set of parameters (CPTs for the n variables). The assumption is made that the cases in $\mathcal{T}$ are i.i.d., and the resulting density for $\mathcal{T}$ is

$$f(\mathcal{T}|\Theta^h, \mathcal{S}^h) = \prod_{i=1}^{t} f(C_i|\Theta^h, \mathcal{S}^h) = \mathcal{L}(\Theta^h|\mathcal{T}, \mathcal{S}^h) \tag{4.4}$$

The likelihood ($\mathcal{L}(\cdot)$) for a set of parameters given a set of training cases $\mathcal{T}$ and hypothesized structure $\mathcal{S}^h$ is the probability of obtaining $\mathcal{T}$ given Θ^h and $\mathcal{S}^h$. The values of Θ^h that maximize the sample likelihood are known as the Maximum Likelihood Estimators MLE's. Thus the goal of MLE is to find a particular $\Theta^h \in \Theta$ that maximizes $\mathcal{L}$, $\Theta^{trained} = \max_{\Theta^h} \mathcal{L}(\Theta|\mathcal{T}, \mathcal{S}^h)$. This maximization function becomes $\log(\mathcal{L}(\Theta^h|\mathcal{T}, \mathcal{S}^h))$ because it is an equivalent and analytically easier calculation. The MLE function is acquired in Matlab from the *learn_params* function in the Bayesian Network Toolbox [17].

Chapter 5

BN Implementation for CP Modeling of Offender Behavior

5.1 Application of BN Learning and Inference to CP

A Bayesian network CP model can be used to estimate offender variables, also providing confidence levels for these predictions, when only the crime scene evidence is observed. Since BN arcs represent relationships between variables, a structure learned from data can be used to discover links between variables and quantify their significance. Thus, a trained BN model is able to determine a criminal profile based upon the crime scene evidence.

Prior to training, the crime scene and offender behavior variables, $\mathcal{X}$, are selected based on expert knowledge (Section 3.2), and the initial structure $\mathcal{S}_o$ is initialized as an empty set (i.e., variables not connected by arcs) assuming no prior knowledge about the node connections, as seen in Figure 5.1a. The training data is used to build the probabilistic model by cycling through the set of possible BN, $\mathcal{B}^h \in \mathcal{B}$ for the purposes of inference when the offender variables are unknown. The learning efficiency of the K2 and K2$'$ algorithms are compared when the number of training cases is limited. The K2$'$ algorithm inhibits connections between the k input nodes X_i^I, for $i = (1, ..., k)$, which reduces the overall computational complexity of the system in order to concentrate training. After the structure is learned, the parameters are learned from the the maximum likelihood parameter estimation (MLE) procedure, which is valid because $\mathcal{T}$ is a complete dataset. The trained model $\mathcal{B}^{trained} \in \mathcal{B}$

obtained is the $\mathcal{B}^h$ that best describes $\mathcal{D}$ and maximizes the scoring metric (4.2). Each algorithm's performance is compared in Chapter 5.2.

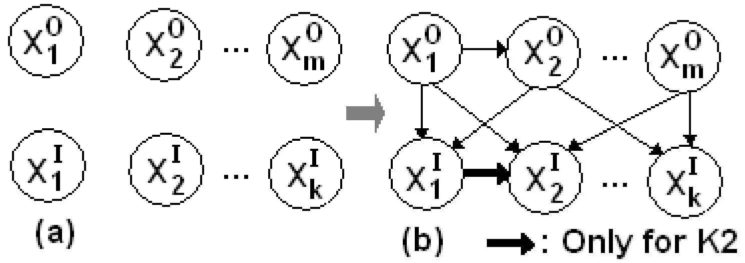

Figure 5.1: The initial BN structure is an empty set with no connections (a) an example of a final structure (b), i.e., where (4.2) is presented, is learned from (a) and $\mathcal{T}$

For this research, the topology of the BN follows the causal representation in that the offender profile is the cause for the resulting crime scene. Also, observations are made from the crime scene with the purpose of predicting the offender profile. Therefore, the inputs are the crime scene variables and the outputs are the offender variables (parent nodes), as is illustrated in Figure 5.2 for m outputs and k inputs.

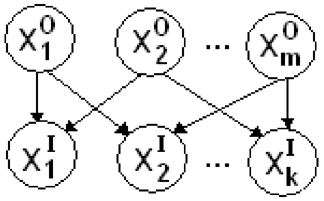

Figure 5.2: Example of the BN structure with offender variables (outputs) that are parent to the crime scene variables (inputs).

The database $\mathcal{D}$ of single offender/single victim homicides used in this research contains 247 cases and are divided into $\mathcal{T}$ (200 cases) and $\mathcal{V}$ (47 cases). The variables in $\mathcal{X}$ are partitioned into 36 *crime scene* (CS) input variables $(X_1^I, ..., X_{36}^I)$ (evidence), and into 21 *offender* (OFF) output variables $(X_1^O, ..., X_{21}^O)$. The outputs comprise the criminal profile to be inferred from the input crime scene evidence. The maximum number of parents allowed per node (u) is set to 10. All variables $X_i^{I,O}$ are binary

($r_i = 2$), with the value $x_{i,j}^{I,O}$ representing whether an event is either present ($x_{i,1}^{I,O} = 1$) or absent ($x_{i,2}^{I,O} = 2$). Examples of five input (crime scene) variables and eight output (offender) variables are listed in Table 5.1, while all 57 variables are listed in Appendix A. For example, if the victim was found to be blindfolded (X_3^I) for a particular case, then X_3^I=1.

Table 5.1: Definition of five crime scene (input) variables and 7 offender (output) variables.

Variable:	**Definition**
X_1^I:	Foreign object penetration
X_2^I:	Face not deliberately hidden
X_3^I:	Victim was blindfolded (at one point)
X_4^I:	Wounds caused by a blunt instrument
X_5^I:	Suffocation (other than strangulation)
X_1^O:	Young offender between 17-21 years
X_2^O:	Criminal record of theft
X_3^O:	Criminal record of fraud
X_4^O:	Criminal record of burglary
X_5^O:	Relationship with victim
X_6^O:	Unemployed at the time of offense
X_7^O:	Male
X_8^O:	Familiar with area of offense occurrance

5.2 Results

5.2.1 Probabilistic Graphical Model of Criminal Behavior

Once a suitable model is attained for purposes of predicting an offender profile, another benefit of the probabilistic BN model is in the graphical display of the relationships learned for a given system. A slice of the K2′ model is shown in Figure 5.3, to illustrate an example of relationships between 5 of the 36 crime scene input variables, $(X_1^I, ..., X_5^I)$, and 8 of the 21 output offender variables, $(X_1^O, ..., X_8^O)$, listed in Table 5.1. Each arc is accompanied by a probabilistic weight in the form of a CPT. Examples of this are Tables 5.2 and 5.3.

An example of an observation drawn from the structure and CPTs comprising the trained BN, a pattern can be found linking the action of deliberately hiding the victim's face (X_2^I) to the offender's gender (X_7^O), as seen in Figure 5.3. The CPT for variable X_2^I is shown in Table 5.2, with the values in the CPT being viewed as a probabilistic degree of influence supporting the state of the unknown variable based on the evidence. The influence between X_7^O and X_2^I is interpreted as strongly supporting a male offender $(X_7^O = x_{7,1}^O = 1)$ if the face is hidden (0.98 compared to 0.75). Instead, if the evidence shows that the victim's face is not hidden, the gender of the offender is more likely female $(X_7^O = x_{7,2}^O = 2$, 0.25 compared to 0.05). However, the BN in Figure 5.3 also shows that when inferring the gender of the offender, X_7^O, the evidence on wounding from a blunt instrument, X_4^I, must also be taken into account. The CPT for X_4^I, shown in Table 5.3, shows that a blunt object being used in the offense supports a male offender (0.26 compared to 0), and vice versa. Of course, through inference in the BN, the influence of all observable crime scene variables on the offender profile is taken into account simultaneously. But these examples show how the learned BN structure also portrays the relationships discovered from the data, and thus can be easily utilized by a multidisciplinary team interested in understanding human behavior.

Table 5.2: Conditional probability table (CPT) for the offender variable X_7^O (male offender) influencing the crime scene variable X_2^I (victim's face not hidden).

	$P(X_2^I = x_{2,1}^I \mid X_7^O)$	$P(X_2^I = x_{2,2}^I \mid X_7^O)$
$X_7^O = x_{7,1}^O$	0.98	0.05
$X_7^O = x_{7,2}^O$	0.75	0.25

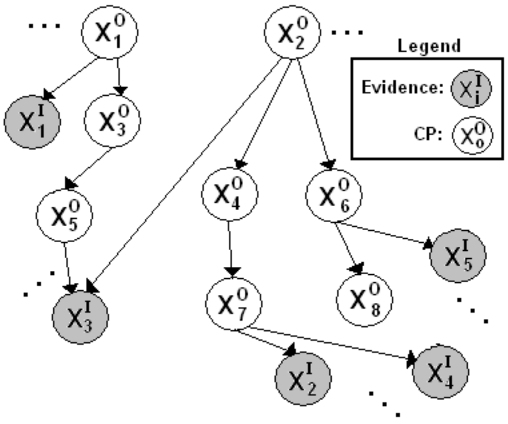

Figure 5.3: A slice from the actual full BN structure that is learned from data by the K2′ algorithm (CPTs are not shown for simplicity).

Table 5.3: Conditional probability table (CPT) for the offender variable X_7^O (male offender) influencing the crime scene variable X_4^I (wounds caused by a blunt instrument).

	$P(X_4^I = x_{4,1}^I \vert X_7^O)$	$P(X_4^I = x_{4,2}^I \vert X_7^O)$
$X_7^O = x_{7,1}^O$	0.26	0.74
$X_7^O = x_{7,2}^O$	0.0	1.0

5.2.2 BN Predictions and Accuracy

When a BN model of offender behavior on the crime scene is learned from solved cases, it is implemented on a set of solved validation cases in order to test the trained model's performance. Performance is tested through probabilistic inference. Inference is the process of updating the probability distribution of a set of possible outcomes based upon the relationships represented by the BN model and the observations of one or more variables. With the updated probabilities, a prediction can be made from the most likely value of each inferred variable. Thus, in order to test the trained model, only the crime scene evidence is inserted into the model, with the predicted offender profile being compared to the actual offender characteristics. Because this is a probabilistic model, a certain confidence accompanies the offender variable predictions.

To complete inference, an inference engine must first be compiled through the steps of graphical manipulations described in Chapter 2.2. This entails identifying all of the conditional independencies among the variables in a structure. A structure is described as a joint density over all of the n variables and can be calculated as,

$$P(X_1, ..., X_n) = \prod_i P(X_i|\pi_i), \tag{5.1}$$

where the variable X_i has n possibilities and π_i represents the instantiation of the parents of X_i. From the directed Markov property stated in Chapter 2.2, the recursive factorization of (5.1) is simplified when, given the evidence, the conditional independence relationships among the variables are identified. In this research, the Matlab functions utilized are *jtree_inf_engine* to build the junction tree; *enter_evidence* to insert evidence; *marginal_nodes* to complete the inference on the specified nodes for the respective junction tree and evidence, and are found in *Bayes Net Toolbox* for

Matlab [17].

An inferred variable refers to a posterior (or predictive) probability distribution, where each of the individual probabilities (also called the marginal probabilities) represent the probability (or confidence) of the particular state given the evidence. Following the probability distribution property of $\sum_{j=1}^{r_i} P(X_i = x_{i,j}|\pi_i) = 1$, the state of a variable is predicted by choosing the state with the maximum marginal probability.

The predictive accuracy of a model is determined by comparing the overall correct predictions of the offender profile. In every one of the 47 validations cases, 21 output variables are predicted, leading to a total of 987 predictions. Because the variables are all binary, a uniformly-random prediction procedure would produce ∼50% predictive accuracy (PA). The predictive accuracy is defined as the frequency at which output variables are inferred correctly over the 47 validation cases, $\mathcal{V}$. A predicted variable is said to be inferred correctly, or its prediction is said to be correct, when the true (observed) state $x_{i,b}^{O}$ is equal to the predicted value $x_{i,a}^{P}$. The overall model predictive accuracy (OPA) is the percentage of correct predictions over the total number of predictions (987). The predictive accuracy of an individual node (IPA) is computed by considering the correct predictions of that node value over the total number of validation cases (47). The K2′ structural learning algorithm seeks uses fewer training cases through additional conditional independence assumptions to obtain a useful CP model, and is compared to its predecessor, the K2 algorithm. The overall complexity of K2′ is $\mathcal{O}(mu^2nkr) = \mathcal{O}(4.79 \times 10^7)$ and is reduced with respect to the K2 algorithm, with complexity $\mathcal{O}(mu^2n^2r) = \mathcal{O}(1.3 \times 10^8)$.

The results in Table 5.4 show that the predictive accuracy of the K2 and K2′ algorithms is better than 50%. This suggests that this BN method may have value in predicting offender profiles in unsolved cases. Also, the K2′ algorithm has a better predictive accuracy than the K2 algorithm. Additionaly, Table 5.4 shows a comparison of the overall performance for the models obtained by the two algorithms. The improved accuracy brought about by the K2′ indicates that the conditional independence relations assumed between the crime scene variables correctly reflect the crime situation.

Table 5.4: Overall performance efficiency for K2 and K2′ algorithms for 987 total predictions.

Algorithm:	**K2**	**K2′**
OPA (%):	64.1%	79.0%
Correct Predictions (number of nodes):	633	780

Further comparison of the K2 and K2′ models involves the confidence levels of each prediction. When compared to other expert systems, such as Neural Networks, probabilistic networks have the added advantage that their predictions are based on posterior probability distributions for the states of each variable, also known as marginal probabilities. The marginal probability $P(x_{i,j}^P|e)$ is computed for each state of an inferred node X_i, and can be seen as the confidence level of a prediction stating that $X_i = x_{i,j}^P$. Table 5.5 shows that as the marginal probability for the predicted variable increases, so does the accuracy of the prediction. The accuracy of nodes predicted with a confidence level CL is denoted by CLA and is calculated by the following formula

$$CLA = \frac{K_{C,CL}}{K_{CL}} * 100, \tag{5.2}$$

where, $K_{C,CL}$ is the total number of correct predictions (subscript C) with a specified confidence level (subscript CL), and K_{CL} is the total number of nodes in the specified confidence level. For example, from Table 5.5 if the designated confidence level is $\geq 70\%$, K_{CL} is the number of nodes with a marginal probability $\geq 70\%$ ($K_{CL} = 573$ for K2 and $K_{CL} = 725$ for K2′), and $K_{C,CL}$ is the number of correctly predicted variables with the $\geq 70\%$ confidence level ($K_{C,CL} = 493$ for K2 and $K_{C,CL} = 618$

for K2′). Thus, the overall number of variables predicted correctly from Table 5.4 is $K_{C,\geq 50\%}$. Table 5.5 also shows a comparison of the K2 and K2′ models with respect to the number of predictions with confidence levels ranging from $\geq 50\%$ to $\geq 95\%$. It is apparent that the K2′ model has significantly more variables that are predicted with a higher confidence level, although the CLA for both methods are similar. For CL=$\geq 70\%$, $CLA = 86\%$ for K2 and $CLA = 85.2\%$ for K2′, but the number of variables predicted correctly is significantly higher for K2′ ($K_{C,CL} = 618$) compared to K2 ($K_{C,CL} = 493$). In Appendix C, Figure C.3 shows a more in-depth view of the CL behavior of the K2 and K2′ algorithm models. Together, Table 5.5 and Figure C.3 support the claim of the higher the marginal probability, which translates to a higher confidence, then the higher the predictive accuracy.

Table 5.5: Confidence level of predictions for the K2 and K2′ algorithm models.

Algorithm:	**K2 ‖ K2′**		
Confidence Level, CL (%):	K_{CL}	$K_{C,CL}$	$CLA(\%)$
$\geq$ 50%	798 ‖ 987	633 ‖ 780	79.3% ‖ 79.0%
$\geq$ 60%	727 ‖ 866	600 ‖ 713	82.5% ‖ 82.3%
$\geq$ 70%	573 ‖ 725	493 ‖ 618	86.0% ‖ 85.2%
$\geq$ 80%	400 ‖ 573	361 ‖ 501	90.3% ‖ 82.5%
$\geq$ 90%	168 ‖ 255	159 ‖ 232	94.6% ‖ 91.0%
$\geq$ 95%	83 ‖ 116	79 ‖ 107	95.2% ‖ 92.2%

Although the CLA of the models obtained by the K2 and K2′ algorithms are close, the number of inferred nodes with a marginal probability $\geq 50\%$ is consistently higher with the K2′ model. Only 798 out of 987 predictions had a predicted marginal probability ≥ 0.5 for the K2 model, because 189 variables had a predictive marginal probability of zero. In this research, the occurrence $P(X_j = x_i|e) = 0$ for $i = 1, 2$, which results in $\sum_{i=1}^{r=2} P(X_j = x_i|e) \neq 1$, is referred to as a “Zero Marginal Probability”

(ZMP) node. ZMP nodes have been observed when inference is performed in a BN with an inadequate number of training cases. Where, the number of training cases required depends not only on the network size and database $\mathcal{D}$, but also on the number of variables that need to be inferred in an unsolved case. As can be deduced from Table 5.5, the number of ZMP nodes is 189, or 19% of all predictions. This idea of ZMP leads to a more accurate calculation of the model's PA (recorded in Table 5.4):

$$PA = \frac{K_t - (K_w + K_{ZMP})}{K_t} \cdot 100, \tag{5.3}$$

where K_w is the number of variables inferred incorrectly, K_{ZMP} is the number of ZMP variables, and K_t is the total number of predictions ($K_t = 987$ for OPA or $K_t = 47$ for IPA). $K_{C,CL}$ for CL$\geq$ 50% is related to K_w and K_{ZMP} by $K_{C,\geq 50\%} = K_w + K_{ZMP}$.

The decrease in prediction efficiency caused by ZMP is overcome by (i) using more training cases, (ii) decreasing the number of system variables, or (iii) decreasing the number of variable relationships. However, the number of cases available is usually not up to the programmer, and it is not good practice to eliminate variables, since important relationships could be lost. The solution (iii) to decrease the search space through additional simplifying assumptions is typically the most useful. In this work, the search space is decreased through the K2$'$ algorithm, which reduces the number of possible variable relationships by exploiting conditional independence properties. Table 5.5 shows that the given training data (200 cases) is sufficient for learning the BN model through the K2$'$ algorithm but insufficient in learning for the K2 algorithm as seen by the presence of ZMP nodes. Although computational savings previously mentioned by the K2$'$ algorithm from the K2 algorithm may appear at first to be insignificant, it is enough to eliminate the ZMP nodes from the model for inference when the number of training cases is limited.

Another method for decreasing the number of ZMP variables while improving the predictive accuracy is to decrease the number of system variables (solution (*ii*)). The variables that are present in more than 50% of all of the cases are considered *high frequency* (HF) and are removed from the model, as discussed in Chapter 1. By removing the HF variables, the model size is systematically decreased by the idea that HF behaviors are interpreted as typical behaviors within the cases and do not add any more insight into the behavior of the offender. A total of four crime scene behaviors are recognized as HF variables and are listed in Table 5.6. Also listed in Table 5.6 is the the variables' frequency of occurrence, and is shown to be similarly distributed between $\mathcal{T}$ and $\mathcal{V}$ from $\mathcal{D}$.

Table 5.6: High frequency CS behaviors.

CS Behavior:	**Frequency (%) in $\mathcal{T}\|\mathcal{V}\|$D**
Face not hidden	88 ‖ 89.4 ‖ **88.4**
Victim found at the scene where he/she was killed	78.5 ‖ 80.9 ‖ **78.9**
Victim found face up	59 ‖ 70.2 ‖ **61.1**
Multiple wounds to the body	54 ‖ 44.7 ‖ **52.2**

The OPA with respect to the resulting models in which the HF variables have been removed (HFMs) are listed in Table 5.7. The results show that the HFM learned from the K2$'$ algorithm has a higher predictive accuracy than the HFM learned from the K2 algorithm, consistent with the results of Table 5.4. Comparing the HFMs listed in Table 5.7 to the original models listed in Table 5.4, the OPA increases slightly for the HFMs, but this improvement is considered negligible. However, the K2 HFM has 21 fewer ZMP variables than the original K2 model. This suggests that if the number of variable relationships cannot be decreased, i.e., the K2$'$ algorithm is not a solution, then the HFM is an alternative solution to reducing the number of ZMP nodes. It is important to note that combining solutions *(ii-iii)* does not improve model performance.

Table 5.7: Comparing the overall performance efficiency of the HFMs obtained by the the K2 and K2′ structural learning algorithms.

Algorithm:	**K2**	**K2′**
OPA (%):	66%	79.6%
$K_{C,\geq 50\%}$ **(number of nodes):**	652	786
K_{ZMP} **(number of nodes)**	168	0

5.2.3 Frequency of Occurrence

Earlier analysis showed that K2 and K2′ are more efficient than a model with no prior information, which would have a 50% OPA. Further analysis compares the K2′ algorithm to a non-intelligent method called *frequency*, F. The frequency of occurrence of a variable is the number of times the variable was present in a dataset. In this instance, f represents the frequency of presence for a variable over $\mathcal{T}$, and the frequency of non-occurrence, $\bar{f}$, represents the frequency of absence for a variable over a dataset, or $\bar{f} = 1 - f$. For example, 93 out of 200 training cases involve an offender with a prior theft conviction (X_2^O=1), which leads to $f = 0.465$ and $\bar{f} = 0.535$. To incorporate the idea of frequency for the prediction of variables in $\mathcal{V}$, f and $\bar{f}$ are acquired from $\mathcal{T}$ for each offender variable. These probabilities can be interpreted as this method's confidence levels. For X_2^O, the variable is more often absent with $\bar{f} = 0.535$, so the variable is predicted to be absent for each of the 47 validation cases. Thus, because X_2^O is absent in 25 of the 47 cases in $\mathcal{V}$, or 0.51, the IPA for the naïve method F for X_2^O is 51%. Table 5.8 shows the offender profile for the eight offender variables listed in Table 5.1 based on frequency of occurrence from $\mathcal{T}$, with their "confidence levels" in parentheses.

Comparing the two methods in similar fashion to K2 and K2′ in Table 5.5, Table 5.9 compares K2′ and F with respect to the number of predictions with confidence levels ranging from $\geq 50\%$ to $\geq 95\%$. However, unlike the results in Table 5.5 that

Table 5.8: The offender profile for eight offender variables from the frequency of occurrence from $\mathcal{T}$ is the same over all $\mathcal{V}$. The third column represents 1=Present, 2=Absent.

Variable:	**Definition**	**F Offender Profile**
X_1^O:	Young offender between 17-21 years	2 (0.81)
X_2^O:	Criminal record of theft	2 (0.54)
X_3^O:	Criminal record of fraud	2 (0.67)
X_4^O:	Criminal record of burglary	2 (0.67)
X_5^O:	Relationship with victim	2 (0.64)
X_6^O:	Unemployed at the time of offense	1 (0.52)
X_7^O:	Male	1 (0.90)
X_8^O:	Familiar with area of offense occurrence	1 (0.86)

clearly supported K2′ over K2, the results in Table 5.9 do not appear to strongly support one algorithm over the other. Due to the absence of ZMP nodes, the OPA is the same as the CLA for $CL \geq 50\%$, which means that the OPA for K2′ and F are approximately the same ($OPA_{K2'} = 79.0\%$, $OPA_F = 79.3\%$). By inspection, F predicts more variables than K2′ when the lower bound is $CL \geq 60\%$, while K2′ predicts more variables than F when the lowerbound of CL increases beyond 60%.

Table 5.9: Confidence level of predictions for the F and K2′ algorithm models.

Algorithm:	**K2′ ‖ F**		
Confidence Level, CL (%):	K_{CL}	$K_{C,CL}$	$CLA(\%)$
$\geq 50\%$	987 ‖ 987	780 ‖ 784	79.0% ‖ 79.3%
$\geq 60\%$	866 ‖ 893	713 ‖ 740	82.3% ‖ 82.9%
$\geq 70\%$	725 ‖ 658	618 ‖ 568	85.2% ‖ 86.3%
$\geq 80\%$	573 ‖ 470	501 ‖ 423	87.4% ‖ 90.3%
$\geq 90\%$	255 ‖ 188	232 ‖ 172	91.0% ‖ 91.5%
$\geq 95\%$	116 ‖ 47	107 ‖ 46	92.2% ‖ 97.9%

Because Table 5.9 does not appear to strongly support the usefulness of K2′, another method is used to differentiate between the two. *Information Entropy*, H

is a quantitative measure of the certainty/uncertainty in a probability distribution describing a system. The amount of information is related to the confidence of the prediction, and is calculated by [3]

$$H = -\sum_{i=1}^{r_i} p_i \log(p_i), \tag{5.4}$$

where $\sum p_i = 1$. From (5.4), $\max(H)$ occurs when $p_i = \frac{1}{r_i}$, or a uniform distribution, and $\min(H)$ occurs when $p_i = 1$ from the property $H = -1\log(1) = 0$ when $p = \{0, ..., 1\}$ [3]. Thus, less entropy means more predictions in the long run. Applying this idea to K2$'$ and F, the best measure is to calculate H over the entire model. H for a model is calculated by the chain rule for entropies,

$$H(X_1, X_2, ..., X_n) = \sum_{i=1}^{k} H(X_i | X_{i-1}, ..., X_1), \tag{5.5}$$

where k is the number of offender variables and the crime scene variables can be thought of as fixed. The independence bound on entropy is

$$H(X_1, X_2, ..., X_n) \leq \sum_{i=1}^{k} H(X_i). \tag{5.6}$$

It is apparent from (5.6) that the independent model is either equal to or less than the same model with the addition of conditionally dependent relationships. So, an improvement that can be incorporated to improve the certainty of a model is to include dependency relationships among the variables. The H calculation for F is simple due to the independence of the offender variables, but is much more difficult for $K2'$ due to all the variable dependencies. Thus, H is calculated with respect to the marginal probabilities $K2'$ and the frequency probabilities for F, and becomes an average of entropies over all of the posterior distributions. The respective calculations of H become $H_F = \sum_{i=1}^{k} H(X_i)$ and $H_{K2'} \leq \sum_{i=1}^{k} H(X_i)$, which lead to an average $H_F = 0.49$ and $H_{K2'} = 0.45$. The initial H for K2′ is not significantly better than F, however, it suggests that there is more certainty involved in the predictions made from the K2′ model rather than F. Proofs of various H relationships are in Appendix B.

Another aspect in which to compare the two models is to show the range of the confidence levels for each variable over all the predictions. Obviously, F has only one confidence level for each variable over all the validation cases. However, the confidence level changes for the K2′ model as it is dependent on the evidence variables for each case. Figure 5.4 shows that for almost all of the variables, the confidence levels for each prediction vary for the K2′ algorithm. Even though the OPA for K2′ and F are equivalent, the K2′ model is much more advantageous as the confidence level takes into account the crime scene evidence and the other offender variables, thus is more descriptive than the F confidence level. It is also shown from Figure 5.4 which offender variables are not improved by training, as evident to the constant confident level across $\mathcal{V}$. The two variables with a fairly constant confidence level are X_{14}^{O} and X_{16}^{O} (definitions are in Appendix A), which can be considered as not improving with training. This could be due to the fact that there is not a strong relationship in the data supporting these variables or because there was insufficient training cases to learn the behavior.

Variables that have a $\sim 50\%$ frequency of occurrence are considered more difficult to predict because they are present and absent at approximately the same rate. Thus, when one of these variables are predicted with a strong confidence level, this is seen as a benefit to the BN modeling. An example of a sample case that is inferred by the BN model from the crime scene evidence is compared to the offender profile by the frequency of occurrence and the actual offender profile. The F offender profile is

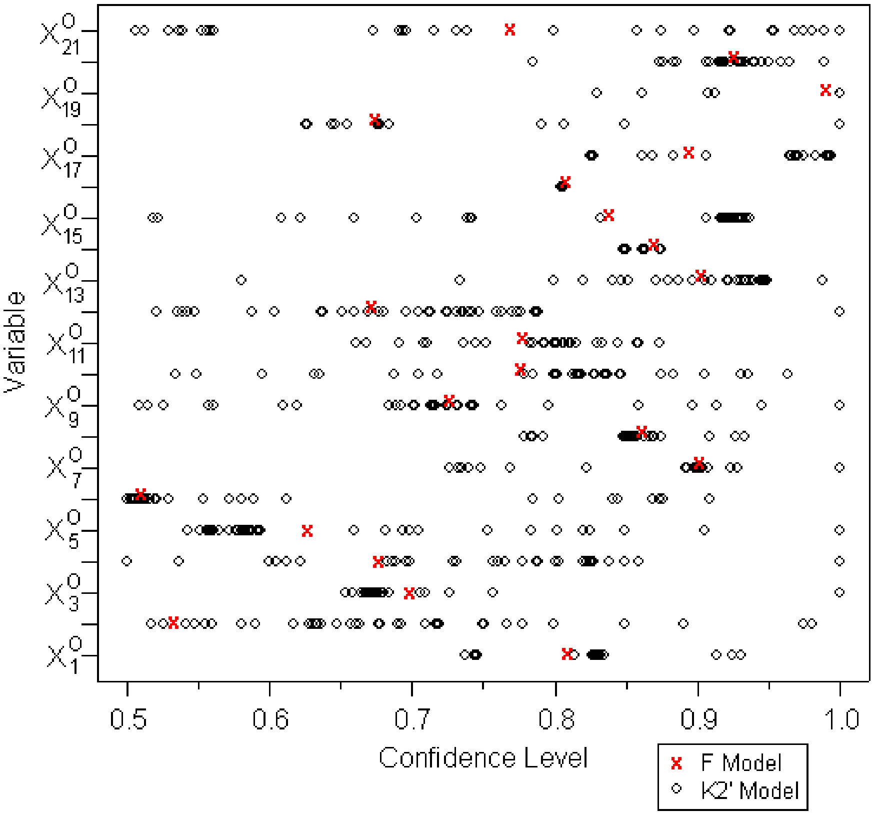

Figure 5.4: The range of the confidence level of predictions for the K2′ and F for each variable over $\mathcal{V}$

the same for all cases (also in Table 5.8), however the confidence levels for the BN model obtained by the K2′ algorithm are the predictive probabilities from inference. From this case, variables X_2^O, X_4^O, X_5^O, and X_6^O all have a frequency of occurrence of approximately 50%. However, X_2^O, X_4^O, and X_6^O are all predicted correctly with a fairly high confidence level, which is $\geq 75\%$. In addition to this, X_3^O, X_5^O, and X_6^O were all predicted incorrectly by the F method but correctly by the BN model.

Table 5.10: A sample of an offender profile obtained by both F and BN models to compare to the actual

Variable:	F **CP**	**K2′ CP**	**Actual CP**
X_1^O:	2 (0.81)	2 (0.93)	2
X_2^O:	2 (0.54)	2 (0.75)	2
X_3^O:	2 (0.67)	1 (0.73)	1
X_4^O:	2 (0.67)	2 (0.82)	2
X_5^O:	2 (0.64)	1 (0.56)	1
X_6^O:	1 (0.52)	2 (0.87)	2
X_7^O:	1 (0.90)	1 (0.89)	1
X_8^O:	1 (0.86)	1 (0.78)	1

5.2.4 Internal Stability

Finally, the K2′ model is tested for internal stability with respect to the validation data. Internal stability refers to the consistency of the predictions made by the model regarding the frequency of a particular marginal probability. This analysis is done by first obtaining a matrix **M**, whose dimensions are 21 × 47 (number of offender variables by number of validation cases), and each entry is the marginal probability for the state "present" for the respective variable and case. For example, the marginal probabilities for X_1^O in case 1 are $(0.6633, 0.3367)$, where the first entry is "present" (X_1^O=1), the second entry is "absent" (X_1^O=2), which makes **M**(1,1)=0.6633. Next, the entries of **M** are grouped into intervals (x) ranging from 0 to 1 in increments of

0.05, where the first interval is [0,0.05) and the last interval is [1]. The number of marginal probabilities in each x is m, and the number of actual "presents" known from $\mathcal{V}$ is p, where the ratio of p/m is y. This probability is plotted versus x for K2$'$ in Figure 5.5. The internal stability plot for the K2 model is in Appendix C but is similar to Figure 5.5. Perfect internal stability occurs essentially when $x = y$ and is represented by the solid line. If there are 10 entries of $\mathbf{M}$ in the range $0 \leq x < 0.05$ ($m = 10$), it is desired that p be very small which makes y very small, as $p = 0 \Longrightarrow x = y = 0$. It can be seen in Figure 5.5 that the K2$'$ model is consistent with $x = y$, thus is considered internally stable.

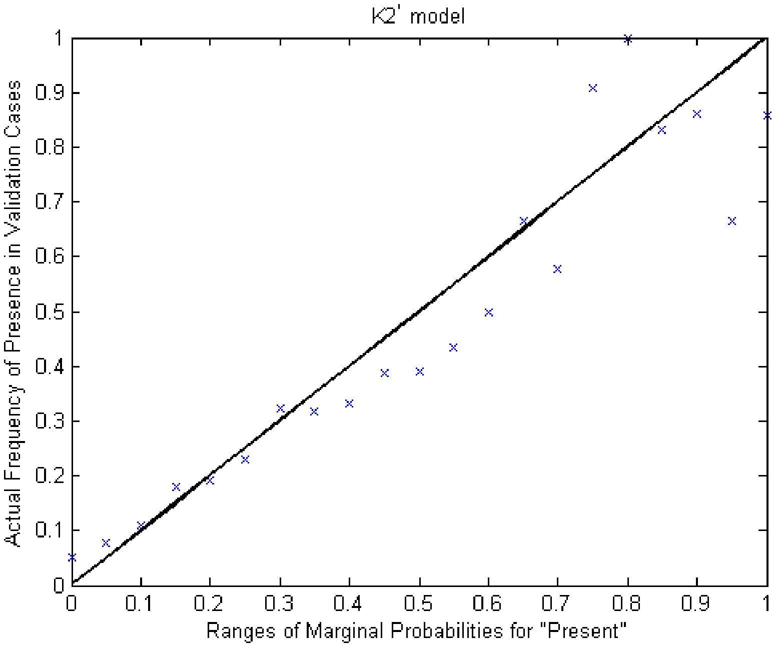

Figure 5.5: Displays the internal stability of the K2$'$ algorithm.

Chapter 6

Conclusions

This thesis presents an approach for deriving a network model of criminal profiling that draws on knowledge-based systems and on fields of criminology and offender profiling. Implementing Bayesian networks makes it possible to represent multidimensional interdependencies between all relevant variables that have been identified in previous research as playing a role in determining or reflecting the behavior of offenders at the crime scene. Hence, a valid network model can be used to predict unknown variables composing an offender profile based on the variables observed from the crime scene. In addition to the predicted criminal profile, confidence levels that denote the probability that the variables predicted are correct can be very valuable in narrowing the list of suspects, due to the fact that variables with the highest confidence can be given a higher priority over the others. In addition, structural learning algorithms and corresponding sensitivity analysis can be used to understand what are the most significant relationships among the CP variables.

The Bayesian network modeling approach to identify underlying patterns of criminal behavior from a database of solved cases implements a well-known structural learning algorithm, known as K2, and compares this to a modified version that exploits conditional independence relations among the input variables. The modified algorithm, referred to as K2$'$, is faster, more effective, and requires fewer number of training cases for learning a BN from data for the purpose of predicting a criminal profile. This thesis shows that additional conditional independence relationships can be effectively incorporated into the learning procedure to increase the final model performance. Inhibiting nodal connections systematically decreases the search space and is shown to improve the model performance considerably. Most importantly, the K2$'$ requires a smaller sample of training cases than the K2 algorithm, which may otherwise lead to ZMP predications. This attribute is particularly useful in applications where additional data is not easily acquired.

The learned structure is particularly useful for understanding human behavior.

The usefulness of the K2$'$ algorithm over the frequency approach was not as apparent as it was over the K2 algorithm. Intuitively, it is obvious that the more efficient method of acquiring a prediction is a model that incorporates the evidence and training from prior solved cases. However, the OPA for the K2$'$ and F models were the same, as was the CLA. It is important to note that there were more predictions made in F with a confidence level less than 70%, and more predictions made for K2$'$ with a confidence level greater than or equal to 70%. Higher confidence levels associated with predictions refer to more correct actual predictions, which supports the K2$'$ algorithm over F. A measure of the confidence level for each prediction is information entropy, which also supports K2$'$. The final analysis for the K2$'$ and F comparisons were the range of confidence levels for each variable over $\mathcal{V}$. This showed the advantage of K2$'$ over F in the fact that the confidence level for the predictions made by the K2$'$ model were in fact affected by the evidence. Also, by showing the ranges of the confidence levels, it is apparent which variables do not benefit from the training. The lack of training required for some variables may be due to the fact that there does not exist strong co-occurrence relationships in the data for these variables or because there was insufficient training cases to learn the behavior. In the long run, it can be concluded that the K2$'$ training algorithm is advantageous over F due to the incorporation of the crime scene evidence in the confidence of the predictions.

Future research should explore a possible collaboration of other intelligent systems, such as neural networks (NN). By combining the probabilistic features of a BN to the non-linear aspects of the NN, a more robust model may be learned from the data for purposes of prediction. Another collaborative recommendation is to combine the categorized instrumental/expressive CP model described in Section 1.3 by [25, 22, 24] to the BN technique described in this thesis. The research method being proposed is to first analyze each training and validation case through the MDS technique developed by [22, 24] to categorize the case and label as either *instrumental* or *expressive*. Once all of the training and validation cases have been divided into categories, the training cases will be used to train a respective BN. Thus, the outcome will be an *expressive* BN and an *instrumental* BN. Next, the validation cases will be used to validate their respective models. The error analysis will be the same as before, with each models' incorrectly predicted variables divided by the total possible predictions. One note that has to be looked into is there are some cases that have traits in more than one category. Depending on the forthcoming analysis, a multi-themed case may end up training both the expressive BN and instrumental BN, as the two categorical models are independent of each other. This highly supervised learning will optimize the learning capability of the BN to allow the intelligent system to more effectively predict an offender profile. In addition to the collaborative research efforts suggested here, another recommendation to advance the methods described in this thesis is to explore variables with more than two states, such as the offender variable "young offender" becoming "age" and having states that range from 17 to older than 65 years of age.

In conclusion, the preliminary results expressed in this thesis support the idea that underlying patterns exist between offenders and their crime, and that they can be learned from a set of solved cases. Future research will expand upon this methodology to systematically evaluate and improve automated criminal profiling techniques.

Appendix A

Crime Scene and Offender Variables

The following is a list of the definitions of the 36 crime scene (input) variables:

Variable	**Definition**
X_1^I:	Foreign object penetration
X_2^I:	Face not deliberately hidden
X_3^I:	Victim was blindfolded
X_4^I:	Wounds caused by blunt instrument
X_5^I:	Suffocation (other than strangulation)
X_6^I:	Vaginal penetration
X_7^I:	Anal penetration
X_8^I:	Face up (victim found as they fell)
X_9^I:	Victim partially undressed
X_{10}^I:	Victim naked
X_{11}^I:	Deliberate clothing damaged
X_{12}^I:	Bound (at one point)
X_{13}^I:	Stabbing injuries
X_{14}^I:	Manual injuries (hitting, kicking, strangled)
X_{15}^I:	Gunshot wounds

continued on next page...

Variable	**Definition**
X_{16}^{I}:	Wounds to the head
X_{17}^{I}:	Wounds to the face
X_{18}^{I}:	Wounds to the neck
X_{19}^{I}:	Wounds to the torso
X_{20}^{I}:	Wounds to the limbs
X_{21}^{I}:	Multiple wounds to one body area (MWOA)
X_{22}^{I}:	Multiple wounds distributed across different body parts (MWD)
X_{23}^{I}:	Weapon brought to scene
X_{24}^{I}:	Weapon from the scene
X_{25}^{I}:	Identifiable property stolen (identification property)
X_{26}^{I}:	Non-identifiable property stolen (non-valuable and unidentifiable)
X_{27}^{I}:	Valuable property stolen
X_{28}^{I}:	Body hidden (outside)
X_{29}^{I}:	Body transported
X_{30}^{I}:	Offender forensically aware
X_{31}^{I}:	Victim found at the same scene where they were killed
X_{32}^{I}:	Sexual crime
X_{33}^{I}:	Arson to crime scene/body
X_{34}^{I}:	Victim found in water

continued on next page...

Variable	**Definition**
X_{35}^{I}:	Victim drugged and/or poisoned
X_{36}^{I}:	Victim covered (i.e., inside rather than outside)

The following is a list of the definitions of the 21 offender (output) variables:

Variable	**Definition**
X_{1}^{O}:	Young offender between 17-21 years
X_{2}^{O}:	Criminal record of theft
X_{3}^{O}:	Criminal record of fraud
X_{4}^{O}:	Criminal record of burglary
X_{5}^{O}:	Relationship with victim
X_{6}^{O}:	Unemployed at the time of offense
X_{7}^{O}:	Male
X_{8}^{O}:	Familiar with area of offense occurrence
X_{9}^{O}:	Criminal record of violence
X_{10}^{O}:	Criminal record of committing damage
X_{11}^{O}:	Criminal record of disorderly conduct
X_{12}^{O}:	Record of imprisonment
X_{13}^{O}:	Sexual related criminal record
X_{14}^{O}:	Armed services, past or present

continued on next page...

Variable	**Definition**
X_{15}^{O}:	Knew victim
X_{16}^{O}:	History of abusiveness in past relationships
X_{17}^{O}:	Attempts of suicide
X_{18}^{O}:	Psychiatric disorders
X_{19}^{O}:	Related to victim
X_{20}^{O}:	Blood relative to victim
X_{21}^{O}:	Turned self into police

Appendix B

Additional Proofs

Theorem 1. The calculation of $P(\mathcal{S}^h|\mathcal{T})$ is equivalent to the calculation of $P(\mathcal{S}^h, \mathcal{T})$ [2].

Proof. To determine which hypothesized structure $\mathcal{S}^h$ best describes a given set of training cases $\mathcal{T}$, the obvious calculation is the posterior probability $P(\mathcal{S}^h|\mathcal{T})$, which quantifies the conditional belief of a particular $\mathcal{S}^h$ given $\mathcal{T}$. By obtaining a rank order [2] of the set of structures with respect to the probability value, i.e., order all $\mathcal{S}^h \in \mathcal{S}$ by largest probability (most compatible) to smallest probability (least compatible), the trained structure is identified. However, due to the intractability of $P(\mathcal{S}^h|\mathcal{T})$, it is recognized that $P(\mathcal{S}^h|\mathcal{T})$ relates to the joint probability, $P(\mathcal{S}^h, \mathcal{T})$, from the following conditional probability property [12]: $P(\mathcal{S}^h, \mathcal{T}) = P(\mathcal{S}^h|\mathcal{T})P(\mathcal{T})$. The result is an equivalence relationship in which the ratios for the pairs of hypothesized structures are rank ordered by their respective posterior probabilities from calculating the joint probabilities,

$$\frac{P(\mathcal{S}_i^h|\mathcal{T})}{P(\mathcal{S}_j^h|\mathcal{T})} = \frac{\frac{P(\mathcal{S}_i^h,\mathcal{T})}{P(\mathcal{T})}}{\frac{P(\mathcal{S}_j^h,\mathcal{T})}{P(\mathcal{T})}} = \frac{P(\mathcal{S}_i^h, \mathcal{T})}{P(\mathcal{S}_j^h, \mathcal{T})}. \tag{B.1}$$

From (B.1), the following property for rank ordering a set of structures holds

$$P(\mathcal{S}_1^h|\mathcal{T}) < P(\mathcal{S}_2^h|\mathcal{T}) \Leftrightarrow P(\mathcal{S}_1^h, \mathcal{T}) < P(\mathcal{S}_2^h, \mathcal{T}).$$

Theorem 2. The joint probability (4.1)

$$P(\mathcal{S}^h, \mathcal{T}) = \int_{\Theta^h} f(\mathcal{T}|\mathcal{S}^h, \Theta^h) f(\Theta^h|\mathcal{S}^h) P(\mathcal{S}^h) d\Theta^h$$

becomes (4.2)

$$P(\mathcal{S}^h, \mathcal{T}) = P(\mathcal{S}^h) \cdot \prod_{i=1}^{n} \prod_{j=1}^{q_i} \frac{(r_i - 1)!}{(\bar{N}_{ij} + r_i - 1)!} \prod_{k=1}^{r_i} N_{ijk}!$$

by the following assumptions [2]:

1. All variables are discrete.

2. All $\mathcal{S}^h$ are equally likely, $P(\mathcal{S}) \sim i.i.d.\ \ Uniform(\alpha)$

3. All cases in $\mathcal{T}$ occur independently given a BN model.

4. All variables are known with no cases that have missing variables.

5. No prior knowledge of the numerical properties to assign to $\mathcal{B}^h$ with structure $\mathcal{S}^h$ before observing $\mathcal{T}$.

Proof. The joint probability integral (4.1) is over all possible value assignments to $\Theta^h \in \Theta$, where Θ^h is a vector with values denoting the conditional probability assignments with respect to $\mathcal{S}^h$. The function $f(\mathcal{T}|\mathcal{S}^h, \Theta^h)$ is the conditional probability density function over $\mathcal{T}$ given $\mathcal{S}^h$ and Θ^h. Likewise, $f(\Theta^h|\mathcal{S}^h)$ is the conditional probability density function over Θ^h given $\mathcal{S}^h$. The term $P(\mathcal{S}^h)$ is the prior probability of $\mathcal{S}^h$. The joint probability of any particular instantiation of all n variables is as follows

$$P(X_i, ..., X_n) = \prod_{i=1}^{n} P(X_i = x_{i,j}|\pi_i). \tag{B.2}$$

Assumptions 1 and 2: By assuming $\mathcal{X} \in \mathcal{D}$ are discrete (assumption 1), the density function $f(\mathcal{T}|\mathcal{S}^h, \Theta^h)$ becomes a probability mass function $P(\mathcal{T}|\mathcal{S}^h, \Theta^h)$. If no prior knowledge is known about the likelihood of any particular structure (assumption 2), the prior probability of $P(\mathcal{S})$ is uniformly distributed with probability α, where

$\alpha = \frac{1}{\mathcal{S}^h_{total}}$, and $\mathcal{S}^h_{total}$ is the total number of $\mathcal{S}^h \in \mathcal{S}$. Thus, $P(\mathcal{S}^h_1) = P(\mathcal{S}^h_2) = ... = P(\mathcal{S}^h_s)$ for s total $\mathcal{S}^h \in \mathcal{S}$, and $P(\mathcal{S}^h)$ is considered a constant. Applying assumptions 1 and 2, (4.1) is rewritten

$$P(\mathcal{S}^h, \mathcal{T}) = P(\mathcal{S}^h) \int_{\Theta^h} P(\mathcal{T}|\mathcal{S}^h, \Theta^h) f(\Theta^h|\mathcal{S}^h) d\Theta^h. \tag{B.3}$$

Assumption 3: $P(\mathcal{T}|\mathcal{S}^h, \Theta^h)$ denotes the probability of reacquiring $\mathcal{T}$ given a structure $\mathcal{S}^h$ and Θ^h. By assuming cases occur independently given $\mathcal{B}^h = (\mathcal{S}^h, \Theta^h)$, the mass function over all of $\mathcal{T}$ becomes the product of the mass functions for each of the cases in $\mathcal{T}$ conditional on $\mathcal{B}^h$. For t total training cases, the following holds: $P(\mathcal{T}|\mathcal{S}^h, \Theta^h) = \prod_{h=1}^{t} P(C_h|\mathcal{B}^h)$. Applying assumption 3, (4.1) is rewritten

$$P(\mathcal{S}^h, \mathcal{T}) = P(\mathcal{S}^h) \int_{\Theta^h} \left[\prod_{h=1}^{t} P(C_h|\mathcal{B}^h) \right] f(\Theta^h|\mathcal{S}^h) d\Theta^h. \tag{B.4}$$

Assumption 4: The following notation is introduced to precede the next assumption. Currently, $x_{i,j}$ is the j^{th} instantiation for X_i. To further apply this notation to each case, $x_{i,j,h}$ is the j^{th} instantiation for X_i in C_h, $x_{variable,state,case}$. Every variable has a set of parents π_i that are instantiated as w_i. If X_i has no parents, π_i and w_i are empty sets, denoted as $\emptyset$. For example, three cases are generated by the BN depicted in Figure 4.1 and the parameters are depicted in Table B.1. X_1 has a parent list $\pi_1 = (X_4)$, and $w_1 = ((x_{4,1,1}), (x_{4,1,2}), (x_{4,1,3}))$ due to the instantiations of X_4 in the three cases. Let $w_{i,j}$ denote the j^{th} element of w_i, where the j^{th} element is the index function $\sigma(i, h)$, the instantiation of π_i in case h. For example, because in case 3 X_1 has the parent variable X_4 ($\pi_1 = X_4$) which is instantiated as $x_{4,1,3}$ and represented by the value w_1, then it follows that $\sigma(1, 3) = 1$ and $w_{1,\sigma(1,3)}$ is equal to $x_{4,1,1}$.

Table B.1: Three sample cases generated by the BN in Figure 4.1

Variable	**Case 1**	**Case 2**	**Case 3**	π_i	w_i
X_1	$x_{1,1,1}$	$x_{1,3,2}$	$x_{1,1,3}$	(X_4)	$((x_{4,1,1}),(x_{4,1,2}),(x_{4,1,3}))$
X_2	$x_{2,r2,1}$	$x_{2,1,2}$	$x_{2,2,3}$	(X_1)	$((x_{1,1,1}),(x_{1,3,2}),(x_{1,1,3}))$
X_3	$x_{3,3,1}$	$x_{3,1,2}$	$x_{3,r3,3}$	(X_1)	$((x_{1,1,1}),(x_{1,3,2}),(x_{1,1,3}))$
X_4	$x_{4,1,1}$	$x_{4,1,2}$	$x_{4,1,3}$	$(\emptyset)$	$(\emptyset)$

From assumption 4 which states all $\mathcal{X} \in \mathcal{D}$ are observed, the term $\prod_{h=1}^{t} P(C_h|\mathcal{B}^h)$ in (B.3) becomes

$$\prod_{h=1}^{t}\prod_{i=1}^{n} P(X_i = x_{i,j,h}|\pi_i = w_{i,\sigma(i,h)}, \Theta^h),$$

where n is the number of variables in $\mathcal{X}$. This expression computes the probability of each case from the conditional probabilities of the variables of the case based on the particular instantiations given proposed parents and Θ^h. Applying assumption 3, (B.4) becomes

$$P(\mathcal{S}^h, \mathcal{T}) = P(\mathcal{S}^h)\int_{\Theta^h}\left[\prod_{h=1}^{t}\prod_{i=1}^{n} P(X_i = x_{i,j,h}|\pi_i = w_{i,\sigma(i,h)}, \Theta^h)\right] f(\Theta^h|\mathcal{S}^h)d\Theta^h. \tag{B.5}$$

Recalling that each of the n variables in $\mathcal{X}$ has r_i possible state $(x_{i,1}, ..., x_{i,ri})$, then N_{ijk} is defined as the number of cases in $\mathcal{T}$ when both $X_i = x_{i,k}$ and $\pi_i = w_{i,j}$ for a maximum of q_i unique instantiations. The sum of N_{ijk} is defined to be $\bar{N}_{ij} = \sum_{k=1}^{ri} N_{ijk}$. Thus, (B.5) can be rewritten as

$$P(\mathcal{S}^h, \mathcal{T}) = P(\mathcal{S}^h)\int_{\Theta^h}\left[\prod_{i=1}^{n}\prod_{j=1}^{q_i}\prod_{k=1}^{r_i} P(X_i = x_{i,k}|\pi_i = w_{i,\sigma(i,j)}, \Theta^h)^{N_{ijk}}\right] f(\Theta^h|\mathcal{S}^h)d\Theta^h. \tag{B.6}$$

To simplify the notation, the following variable assignment is made: $\theta_{ijk} = P(X_i = x_{i,k}|\pi_i = w_{i,\sigma(i,j)}, \Theta^h)$. The conditional probability of θ_{ijk} is in the probability distribution of $(\theta_{ij1}, ..., \theta_{ijr_i})$, such that $\sum_{k=1}^{ri} \theta_{ijk} = 1$. The probability density function over the probability distribution for a given $x_{i,k}$ and $w_{i,j}$ is denoted as $f(\theta_{ij1}, ..., \theta_{ijr_i})$ and is called a *second-order probability distribution* [2, 6].

Assumption 5: Assumption 5 states that prior to observing $\mathcal{T}$, all $\mathcal{B}^h \in \mathcal{B}$ are equally likely to occur. This differs from assumption 2 as assumption 2 only included the indifference of the prior structure. Assumption 5 is two-fold in implying that $f(\theta_{ij1}, ..., \theta_{ijr_i})$ is independent and uniformly distributed for $1 \leq i \leq n, 1 \leq j \leq q_i$. By being independently distributed, $f(\Theta^h|\mathcal{S}^h)$ can be rewritten to be $f(\Theta^h|\mathcal{S}^h) = \prod_{i=1}^{n} \prod_{j=1}^{qi} f(\theta_{ij1}, ..., \theta_{ijr_i})$. This expression refers to the independence of $f(\theta_{ij1}, ..., \theta_{ijr_i})$ in that the values are not influenced by the values of other second-order probability distributions. Substituting this expression, (B.6) becomes

$$P(\mathcal{S}^h, \mathcal{T}) = P(\mathcal{S}^h) \int_{\theta_{ijk}} ... \int \left[\prod_{i=1}^{n} \prod_{j=1}^{q_i} \prod_{k=1}^{r_i} \theta_{ijk}^{N_{ijk}} \right] \left[\prod_{i=1}^{n} \prod_{j=1}^{qi} f(\theta_{ij1}, ..., \theta_{ijr_i}) \right] d\theta_{ij1}, ..., d\theta_{ijr_i}. \tag{B.7}$$

Equation (B.7) is currently considered an integral of products. By identifying the independent terms, the outer and middle products are factored outside of integral to convert (B.7) to a product of integrals

$$P(\mathcal{S}^h, \mathcal{T}) = P(\mathcal{S}^h) \prod_{i=1}^{n} \prod_{j=1}^{q_i} \int_{\theta_{ijk}} ... \int \left[\prod_{k=1}^{r_i} \theta_{ijk}^{N_{ijk}} \right] f(\theta_{ij1}, ..., \theta_{ijr_i}) d\theta_{ij1}, ..., d\theta_{ijr_i}. \tag{B.8}$$

Because $f(\theta_{ij1}, ..., \theta_{ijr_i})$ is uniformly distributed, this refers to the indifference of the values for $\theta_{ij1}, ..., \theta_{ijr_i}$. Similar to $P(^h) = constant$ in assumption 2, $f(\theta_{ij1}, ..., \theta_{ijr_i})$

is equal to some constant, κ_{ij}, for a certain i and j. $f(\theta_{ij1}, ..., \theta_{ijr_i}) = \kappa_{ij}$ follows the probability distribution property of

$$\int_{\theta_{ijk}} ... \int \kappa_{ij} d\theta_{ij1}, ..., d\theta_{ijr_i} = 1.$$

Substituting this relationship into (B.8), the joint probability relationship is rewritten

$$P(\mathcal{S}^h, \mathcal{T}) = P(\mathcal{S}^h) \prod_{i=1}^{n} \prod_{j=1}^{q_i} \kappa_{ij} \int_{\theta_{ijk}} ... \int \left[\prod_{k=1}^{r_i} \theta_{ijk}^{N_{ijk}} \right] d\theta_{ij1}, ..., d\theta_{ijr_i}. \tag{B.9}$$

The probability density function $f(\theta_{ij1}, ..., \theta_{ijr_i})$ is a special case of Dirichlet's distribution [6] and the multiple integral in (B.9) is Dirichlet's integral and has the following solution [30]

$$\int_{\theta_{ijk}} ... \int \left[\prod_{k=1}^{r_i} \theta_{ijk}^{N_{ijk}} \right] d\theta_{ij1}, ..., d\theta_{ijr_i} = \frac{\prod_{k=1}^{r_i} N_{ijk}!}{(\bar{N}_{ij} + r_i - 1)!}. \tag{B.10}$$

By solving (B.9) for κ_{ij} by substituting in (B.10) and $N_{ijk} = 0$, which in turn makes $\bar{N}_{ij} = 0$, the result is $\kappa_{ij} = (r_i - 1)!$. This leads to the scoring metric of (4.2)

$$P(\mathcal{S}^h, \mathcal{T}) = P(\mathcal{S}^h) \cdot \prod_{i=1}^{n} \prod_{j=1}^{q_i} \frac{(r_i - 1)!}{(\bar{N}_{ij} + r_i - 1)!} \prod_{k=1}^{r_i} N_{ijk}!. \tag{B.11}$$

Theorem 3. $H(X) \geq 0$ [3].

Proof. $0 \leq p(x) \leq 1$ implies $\log(1/p(x))) \geq 0$

Theorem 4. $H(X) \equiv 0$ when p=0 or 1 [3].

Proof. $H(X) = -p \log(p) - (1 - p) \log(1 - p) \equiv H(p)$. When p=0 or 1, the variable is not random and there is no uncertainty. Figure B.1 is a plot of H versus p_i, and shows the maximum H when $p_i = 0.5$ for a binary variable ($r_i = 2$) and a minimum

H when $p_i = 0$ or 1.

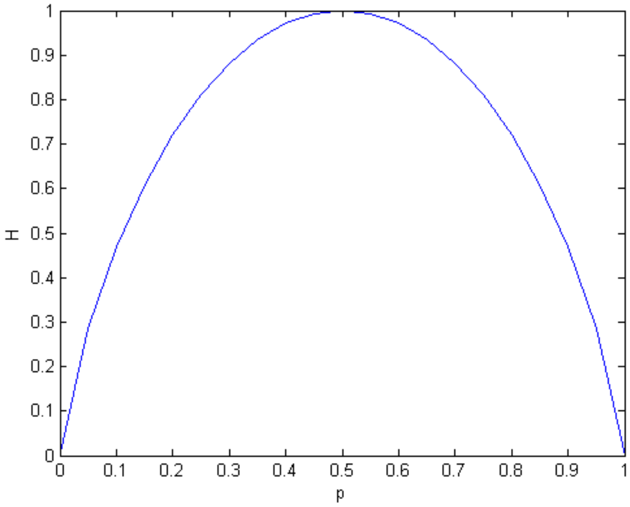

Figure B.1: Information entropy (H) versus probability (p) for $\log_2$

Theorem 5. *Chain rule for entropy*: Let $X_1, ..., X_n$ be drawn according to $P(x_1, ..., x_n)$, then $H(X_1, ..., X_n) = \sum_{i=1}^{n} H(X_i|X_{i-1}, ..., X_1)$ [3].

Proof. $H(X_1, ..., X_n) = H(X_1) + H(X_2|X_1) + ... + H(X_n|X_{n-1}, ..., X_1) \Longrightarrow$

$\therefore H(X_1, ..., X_n) = \sum_{i=1}^{n} H(X_i|X_{i-1}, ..., X_1)$

Additional Justification for the K2′ algorithm The K2′ algorithm inhibits nodal connections among evidence variables in order to reduce the search space. Figure B.2a and Figure B.2b in that there exists a relationship between the evidence variable X_2 and X_3 in Figure B.2a. If it is known prior to training that X_2 and X_3 will always be observed, then it is stated that Figure B.2a and Figure B.2b are equivelant structures.

Proof. A structure is described as a joint density over all of the n variables by (5.1), restated here as

$$P(X_1, ..., X_n) = \prod_{i=1}^{n} P(X_i|\pi_i). \tag{B.12}$$

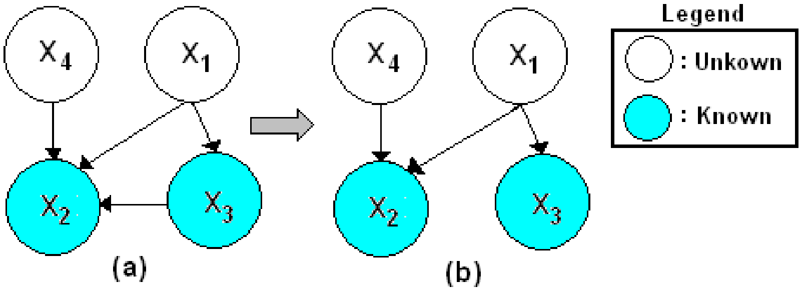

Figure B.2: An example of a BN in which the evidence variables are connected (a) and the equivalent structure inhibiting the connection of evidence variables

Equation (B.12) is also referred to as *recursive factorization* according to the DAG. The recursive factorization [5] for the graphical model in Figure B.2a is

$$\begin{aligned} P(X_1, X_2, X_3, X_4) &= P(X_4)P(X_1)P(X_2|X_4, X_1, X_3)P(X_3|X_1) \\ &= P(X_4)P(X_1)P(X_2|X_4)P(X_2|X_1)P(X_2|X_3)P(X_3|X_1). \end{aligned} \tag{B.13}$$

The *fundamental rule* for probability calculus [12] is

$$P(a|b)p(b) = P(a, b), \tag{B.14}$$

which becomes

$$P(a|b)p(b) = P(b|a)P(a). \tag{B.15}$$

Equation (B.15) is applied to the following terms in (B.13) to yield

$$P(X_4)P(X_2|X_4) = P(X_2)P(X_4|X_2), \tag{B.16}$$

$$P(X_1)P(X_3|X_1) = P(X_3)P(X_1|X_3), \tag{B.17}$$

$$P(X_2|X_1) = \frac{P(X_1|X_2)P(X_2)}{P(X_1)}. \tag{B.18}$$

Finally, if X_2 and X_3 are both known, then it follows that

$$P(X_2|X_3) = 1. \tag{B.19}$$

Applying (B.16-B.19) to (B.13), the joint density over all the variables in Figure B.2a becomes

$$P(X_1, X_2, X_3, X_4) = \frac{P(X_2)P(X_4|X_2)P(X_3)P(X_1|X_3)P(X_1|X_2)P(X_2)}{P(X_1)}. \tag{B.20}$$

Similarly, the recursive factorization for the graphical model in Figure B.2b is

$$\begin{aligned} P(X_1, X_2, X_3, X_4) &= P(X_4)P(X_1)P(X_2|X_4, X_1)P(X_3|X_1) \\ &= P(X_4)P(X_1)P(X_2|X_4)P(X_2|X_1)P(X_3|X_1). \end{aligned} \tag{B.21}$$

Utilizing the relationships (B.16-B.18), the joint density over all the variables in Figure B.2b becomes

$$P(X_1, X_2, X_3, X_4) = \frac{P(X_2)P(X_4|X_2)P(X_3)P(X_1|X_3)P(X_1|X_2)P(X_2)}{P(X_1)}, \tag{B.22}$$

which is the same as (B.22). Thus, Figure B.2a is equivalent to Figure B.2b *iff* X_2 and X_3 are observed.

Appendix C

Additional Figures and Tables

Figures C.1-C.2 show the internal stability plots for K2 and F, which is similar to Figure 5.5.

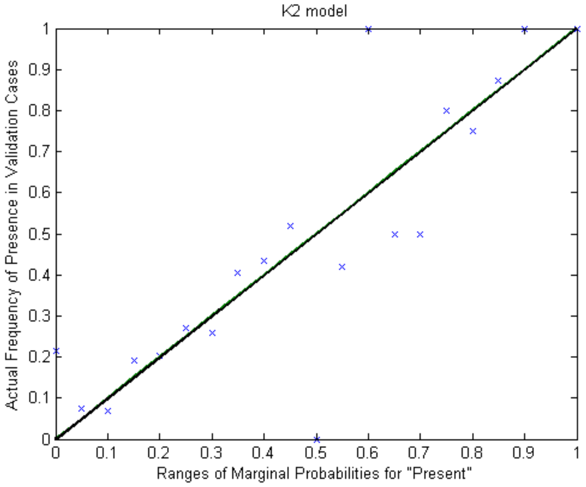

Figure C.1: Internal stability of the K2 model

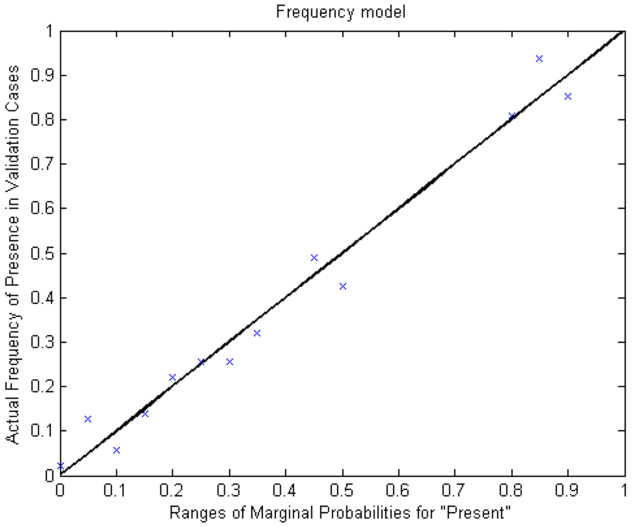

Figure C.2: Internal stability of the F model

This figure compliments Table 5.5. Whereas Table 5.5 displays the CL analysis for K2 and K2′ algorithms for $CL = \geq 50\%, \geq 70\%, \geq 90\%$, Figure C.3 shows more in-depth the behavior for CL ranging from $\geq$ 50% to $\geq$ 95%, with increments of 5%. It is obvious that as the CL increases, the difference between K_C and $K_{C,CL}$ decreases. Figure C.3 supports the claim that the higher the marginal probability, which translates to a higher confidence, then the better the predictive accuracy.

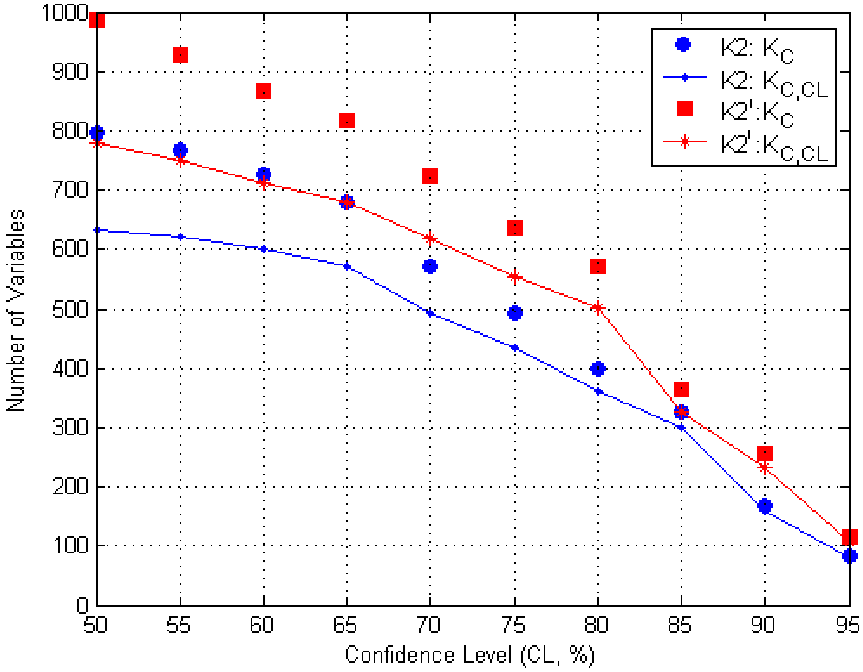

Figure C.3: Comparison of the K2 and K2′ algorithms' confidence level of predictions

The lower bound on the predictive accuracy for individual variables, obtained from the K2′ model, is shown in Table C.1. These results show that a significant number of offender variables has a predictive accuracy higher than 60%, and that 76% of the variables has an IPA greater or equal to 70%. Table C.2 shows the predictive accuracies of the output nodes defined in Table 5.1 and used in the graph in Figure 5.3. The mean predictive accuracy for this sample of 8 offender nodes is 70.2%, with a standard deviation of 15.1%. This sample mean and standard deviation is consistent with the rest of the offender variables, which has a mean predictive accuracy of 79.1% and standard deviation of 13.7%.

Table C.1: The number of offender nodes (21 possible) with the corresponding PA for the K2′ model.

IPA (%)	**Number of offender nodes**
$< 50\%$	1
$\geq 50\%$	20
$\geq 60\%$	19
$\geq 70\%$	16
$\geq 80\%$	11
$\geq 90\%$	5

Table C.2: The IPA (%) for each offender (output) variable defined in Table 5.1 inferred over 47 validation cases.

Variable:	*IPA* (%)
X_1^O:	68.1%
X_2^O:	57.4%
X_3^O:	72.3%
X_4^O:	72.3%
X_5^O:	63.8%
X_6^O:	46.8%
X_7^O:	87.2%
X_8^O:	93.6%

A complement to Table C.1, Table C.3 shows the number of offender nodes with an IPA above a specified lower bound. The number of offender variables in each category is the same for K2$'$ and F, except for one more node in F with an IPA$\geq$ 70%. Even though the two are similar in this regard, it was described throughout Section 5.2 as to the additional benefits of K2$'$ over F.

Table C.3: The number of offender nodes (21 possible) with the corresponding PA for the K2$'$, K2, and F algorithms.

IPA (%)	**K2$'$**	**K2**	**F**
$< 50\%$	1	2	1
$\geq 50\%$	20	19	20
$\geq 60\%$	19	13	19
$\geq 70\%$	16	7	17
$\geq 80\%$	11	0	11
$\geq 90\%$	5	0	5

Bibliography

[1] D. Canter. *Criminal shadows: Inside the mind of a serial killer.* London: Harper Collins, 1994.

[2] G.F. Cooper and E. Herskovits. A Bayesian method for the induction of probabilistic networks from data. *Machine Learning*, 9:309–347, 1992.

[3] T.M. Covers and J.A. Thomas. *Elements of Information Theory.* John Wiley & Sons, Inc., 1991.

[4] R. Cowell. Advanced inference in Bayesian networks. In M.I. Jordan, editor, *Learning in Graphical Models*, pages 27–50. 1998.

[5] R. Cowell. Introduction to inference for Bayesian networks. In M.I. Jordan, editor, *Learning in Graphical Models*, pages 9–26. 1998.

[6] M.H. deGroot. *Optimal Statistical Decisions.* New York: McGraw-Hill, 1970.

[7] S.A. Egger. Psychological profiling: Past, present, and future. *Journal of Contemporary Criminal Justice*, 15(3):242–261, August 1999.

[8] S. Ferrari and A. Vaghi. Sensor modeling and feature-level fusion by Bayesian networks. *Journal of Smart Structures and Systems*, 1(1):1–9, November 2004.

[9] D. Heckerman. A tutorial on learning with Bayesian networks. In M.I. Jordan, editor, *Learning in Graphical Models*, pages 301–354. 1998.

[10] D. Heckerman. A Bayesian approach to learning causal networks. *Technical Report MSR-TR-95-04*, pages 1–23, May 1995.

[11] D. Heckerman, D. Geiger, and D.M Chickering. Learning Bayesian networks: The combination of knowledge and statistical data. *Machine Learning*, 20:197–243, 1995.

[12] F.V. Jensen. *Bayesian Networks and Decision Graphs.* Springer-Verlag, 2001.

[13] Ming Jiang. *Markov Random Fields and Application.* [Online]. Avaliable: http://ct.radiology.uiowa.edu/ jiangm/courses/mm-cv-ip/, 2001.

[14] R.N. Kocsis, R.W. Cooksey, and H.J. Irwin. Psychological profiling of offender characteristics from crime scene behaviors in serial rape offences. *International Journal of Offender Therapy and Comparative Criminology*, 46(2):144–169, 2002.

[15] R.N. Kocsis, H.J. Irwin, and A.F. Hayes. Organized and disorganized behavior syndromes in arsonists: A validation study of a psychological profiling concept. *Psychiatry, Psychology, and Law*, 5:117–130, 1998.

[16] K. Murphy. *A Brief Introduction to Graphical Models and Bayesian Networks.* [Online]. Avaliable: http://www.cs.ubc.ca/ murphyk/Bayes/bayes.html, 1998.

[17] K. Murphy. *How To Use Bayes Net Toolbox.* [Online]. Avaliable: http://www.ai.mit.edu/ murphyk/Software/BNT/bnt.html, 2004.

[18] A.J. Pininzzotto. Forensic psychology: Criminal personality profiling. *Journal of Police Sciences and Administration*, 12(1):32–40, 1984.

[19] A.J. Pinizzotto and Finkel. Criminal personality profiling: An outcome and process study. *Law and Human Behavior*, 14(3):215–233, June 1990.

[20] R.K. Ressler, A. Burgess, and J.E. Douglas. *Sexual Homicide: Patterns and motives.* New York: Lexington Books, 1988.

[21] R.W. Robinson. *Counting unlabeled acyclic digraphs. In C.H.C. Little(Ed.)* Lecture notes mathematics, 622: Combinatorial mathematics V. Springer-Verlag, 1977.

[22] C.G. Salfati. Profiling homicide: A multidimensional approach. *Homicide Studies*, 4:265–293, 2000.

[23] C.G. Salfati. Greek homocide, a behavioral examination of offender crime-scene actions. *Homicides Studies*, 5(4):335–362, November 2001.

[24] C.G. Salfati. Offender interaction with victims in homicide: A multidimensional analysis of crime scene behaviors. *Journal of Interpersonal Violence*, 18(5):490–512, 2003.

[25] C.G. Salfati and D.V. Canter. Differentiating stranger murders: Profiling offender characteristics from behavioral styles. *Behavioral Science and the Law*, 17:391–406, 1999.

[26] C.G. Salfati and F. Dupont. Canadian homicide: An investigation of crime scene actions. *Homicide Studies*, In Press for 2005.

[27] C.G. Salfati and L.T. Kucharski. *The Psychology of Criminal Conduct. In J. Trevino and S. Fuarino (Eds.), The common subject of crime: A multidisciplinary approach.* Anderson Publishing, In Press for 2005.

[28] P. Santtila, H. Häkkänen, D. Canter, and T. Elfgren. Classifying homocide offenders and predicting their characteristics from crime scene behavior. *Scandinavian Journal of Psychology*, 44:107–118, 2003.

[29] T.T. Taiuhenua. Internet traders of child pornogrpahy and other censorship offenders in New Zealand. *the Department of Internal Affairs*, 4:103–131, 2003.

[30] S.S. Wilks. *Mathematical statistics.* New York: John Wiley & Sons, 1994.

[31] M. Woodworth and S. Porter. Historical foundations and current applications of criminal profiling in violent crime investigations. *Expert Evidence*, 7:241–264, 1999.