Univerzita Karlova v Praze

Matematicko-fyzikální fakulta

DIPLOMOVÁ PRÁCE

Evgeniya Belyaeva

Anotační grafy a Bayesovské sítě

Katedra pravděpodobnosti a matematické statistiky

Vedoucí diplomové práce:	RNDr. Milan Studený, DrSc.
Studijní program:	Matematika
Studijní obor:	Pravděpodobnost, matematická statistika a ekonometrie

Praha 2015

Prohlašuji, že jsem tuto diplomovou práci vypracoval(a) samostatně a výhradně s použitím citovaných pramenů, literatury a dalších odborných zdrojů.

Beru na vědomí, že se na moji práci vztahují práva a povinnosti vyplývající ze zákona č. 121/2000 Sb., autorského zákona v platném znění, zejména skutečnost, že Univerzita Karlova v Praze má právo na uzavření licenční smlouvy o užití této práce jako školního díla podle §60 odst. 1 autorského zákona.

V .... dne .... Podpis autora

Název práce: Anotační grafy a Bayesovské sítě

Autor: Evgeniya Belyaeva

Katedra: Katedra pravděpodobnosti a matematické statistiky

Vedoucí diplomové práce: RNDr. Milan Studený, DrSc., Ústav teorie informace a automatizace

Abstrakt: Existují různé modely, které popisují struktury podmíněné nezávislosti indukované mnohorozměrnými rozděleními. V této práci jsou popsány a porov nány modely neorientovaných grafů, acyklických orientovaných grafů, řetězcových grafů a anotačních grafů. Zvláštní pozornost je věnovaná anotačním grafům. Je ukázáno, že anotační grafy reprezentují třídy ekvivalencí relací, které se dají reprezentovat pomocí acyklických orientovaných grafů. Je dán algoritmus pro rekonstrukci anotačního grafu z řetězcového grafu a taky algoritmus pro zpětnou transformaci. Některé vlastnosti charakteristického imsetu, jenž není grafickou representací, jsou diskutovány. Je prozkoumán vztah mezi charakteristickým imsetem a anotačním grafem: je dán algoritmus, který zajišťuje rekonstrukci anotačního grafu z charakteristického imsetu.

Klíčová slova: anotační grafy řetězcové grafy charakteristické imsety Bayesovské sítě

Title: Annotated graphs and Bayesian networks

Author: Evgeniya Belyaeva

Department: Department of Probability and Mathematical Statistics

Supervisor: RNDr. Milan Studený, DrSc., Institute of Information Theory and Automation

Abstract: There are different models, which describe conditional independence induced by multivariate distributions. Models such as Undirected Graphs, Directed Acyclic Graphs, Essential Graphs and Annotated Graphs are introduced and compared in this thesis. The focus is put on annotated graphs. It is shown that annotated graphs represent equivalence classes of DAG-representable relations. An algorithm for reconstruction of an annotated graph from an essential graph as well as the algorithm for the inverse procedure are given. Some properties of a characteristic imset, which is a non-graphical representation, are discussed. A relationship between annotated graphs and characteristic imsets is investigated, an algorithm, which reconstructs an annotated graph from a characteristic imset is given.

Keywords: annotated graphs essential graphs characteristic imsets Bayesian networks

I would like to thank my supervisor, RNDr. Milan Studený, DrSc., for his helpful guidance and advice. I am grateful to my parents, Aelita Belyaeva and Konstantin Belyaev for their continuous support throughout my studies. I am also thankful to Pavel Čoupek for his overall support.

Contents

List of Symbols

$\mathbb{R}$	the set of real numbers
$\mathbb{N}$	the set of natural numbers
$(\Omega, \mathscr{A}, \mathbf{P})$	probability space
$\mathscr{A}$	σ-algebra
$\mathbf{P}$	a probability measure on the measurable space $(\Omega, \mathscr{A})$
ω	an element in the set Ω
X	random variable
$\mathbf{X}$	random vector or a set of random variables
$\mathscr{B}$	Borel σ-algebra
$(S, \mathscr{S})$	measurable space
$F(x)$	distribution function
Q	Lebesgue-Stieltjes measure
$(X_1, ..., X_n)$	a random vector
$a \in \mathbf{A}$	an element a is included in the set $\mathbf{A}$
$a \notin \mathbf{A}$	an element a is not included in the set $\mathbf{A}$
$(-\infty, \mathbf{x})$	a multi-dimensional interval between minus infinity and a vector $\mathbf{x} \in \mathbb{R}^n$
$\mathbf{A}, \mathbf{B}, ...$	sets
$\mathbf{A} \setminus \mathbf{B}$	set difference
$x \to \infty$	variable x converges to infinity
$\subset$	nonstrict inclusion
$\subsetneq$	strict inclusion
$\prod_{i \in \mathbf{C}}$	a product of functions or sets
$B \times C$	Cartesian product
$\int_a^b f(x)dx$	Lebesgue integral of the function $f(x)$ over the interval (a, b)
$\int_{x \in A} f(x)dx$	Lebesgue integral of the function $f(x)$ over the set $A \in \mathscr{B}$
$P << Q$	measure P is absolutely continuous with respect to measure Q
$\bigotimes_{i \in \mathbf{V}} P_i$	a product measure of measures from the set $\{P_i, i \in \mathbf{V}\}$
$\mathbf{A} \cup \mathbf{B} = \mathbf{AB}$	a union of sets
$G = (\mathbf{V}, E)$	a graph
(u, v)	an ordered pair of elements from some set
$\{u, v\}$	an unordered pair of elements from some set
$u \neq v$	element u is not equal to element v

$:=$	is defined as
$(v_i)_i^n$	a finite sequence of elements $\{v_1, ..., v_n\}$
$D \simeq \tilde{D}$	D is Markov equivalent with $\tilde{D}$
$\varnothing$	an empty set
$\forall x \in \mathbf{X}$	for every element in the set $\mathbf{X}$
$\exists c \in \mathbf{C}$	there exists an element c in the set $\mathbf{C}$
$W \Longrightarrow T$	W induced T
$W \Longleftrightarrow T$	W is equivalent with T
$u \longrightarrow v$	undirected edge between u and v
$u \longrightarrow v$	directed edge between u and v
$ne_G(v)$	a set of neighbours of v
$ch_G(v)$	a set of children of v
$pa_G(v)$	a set of parents of v
$C(\mathbf{V})$	conditional independence structure
$\bigcup_{i \in \mathbf{I}} \mathbf{A}_i$	supremum of sets from the set $\{A_i, i \in \mathbf{I}\}$
$[D]$	a class of equivalent DAGs
$A = (U_A, N_A)$	an annotated graph
$\mathbf{c}_D$	characteristic imset based on graph D

Introduction

This thesis deals with a topic, which lies on the intersection of two mathematical fields: Probability Theory and Graph Theory. We examine multivariate distributions (both in the discrete and continuous case) and try to describe their conditional independence structures. Some variables in the multivariate distribution are independent, some of them are conditionally independent given a set of other variables and this gives structural information on every multivariate distribution. Specifically, the conditional independence structure of a distribution is the set of triplets $(\mathbf{X}, \mathbf{Y}, \mathbf{Z})$ such that each component represents a disjoint subset of variables from the multivariate distribution and the subset $\mathbf{X}$ is conditionally independent of the subset $\mathbf{Y}$ given the subset $\mathbf{Z}$. Recovering such conditional independence structure from a data set is the first step in describing the whole multivariate distribution.

It is inefficient to keep the whole list of the triplets from a conditional independence structure; therefore, more efficient ways of its representation were introduced by various authors. In this thesis we describe some graphical methods such as Undirected Graphs, Directed Acyclic Graphs (DAGs), Essential graphs and Annotated graphs. Also, we present a non-graphical representation called characteristic imset, which is a special zero-one vector. So far the most common representation of conditional independence structures is Directed Acyclic Graphs, also called Bayesian networks. DAG representation of a conditional independence structure is in general not unique: two different DAGs may represent the same conditional independence structure. The set of DAGs, who represent the same conditional independence structure is called the class of equivalence of DAGs.

The main goal of this thesis is to analyse the relations between the annotated graph and the other sorts of conditional independence structure representations. Firstly, we show that each class of equivalence of DAGs is represented by one annotated graph. It means that two directed acyclic graphs from the same equivalence class are represented by the same annotated graph, while two DAGs which belong to different equivalence classes are represented by different annotated graphs. This was discussed by Paz, in his papers (Paz [2003b], [Paz, 2003a]). Our contribution is a new direct graphical proof of the fact that non-equivalent DAGs induce different annotated graphs. This proof was skipped in Paz [2003a] with tacit assumption that a reader is able to derive that fact from results of Paz [2003b] and further results on conditional independence structures induced by DAGs. Our proof, however, does not rely on the concept of conditional independence structure induced by a DAG.

Secondly, we found out what is the connection between the annotated graph and the corresponding essential graph. Essential graph is known and used as a uniquely determined graphical representative of a class of equivalence. We gave two algorithms:

(i) the first one transforms the essential graph to the annotated graph, which represents the same conditional independence structure

(ii) the second one does the opposite transformation using the rules formulated by Meek in his paper Meek [1995].

The fact that there exists the two-way transformation between the annotated graph and the essential graph can also be used to show in another way, that the annotated graph is a unique representative of the equivalence classes, since the essential graph is.

Lastly, we find a similar connection between annotated graphs and characteristic imsets (introduced by Hemmecke and Linder, M. Studený [2010]), which is a unique representative of the equivalence class of DAGs. Characteristic imset is the only non-graphical representation of conditional independence structures discussed in this thesis. Its main advantage is that it allows one to implement efficient algorithms for learning DAG conditional independence structure from data, which is based on integer linear programming M. Studený [2014]. We introduce an algorithm which transforms a characteristic imset into the annotated graph, which corresponds to the same conditional independence structure as the original characteristic imset. We provide a proof of the last statement.

In Chapter 1, we introduce the concepts of multivariate distributions and conditional independence. In Chapter 2 we remind the reader of basic graph-theoretical notions. Chapter 3 deals with different graphical representations of conditional independence structures starting with undirected graphs and finishing with annotated graphs. Two algorithms of transformations between essential graphs and annotated graphs are introduced in Chapter 4. The hypothesis of equivalence introduced by Paz [2003b] is proved in Chapter 5. Finally, in Chapter 6 we look into the concept of characteristic imset and introduce an algorithm, which transforms an imset into an annotated graph, which corresponds to the same conditional independence structure.

1. Preliminaries

1.1 Probability concepts

All objects of interest presented in this thesis have a basis in a probability space $(\Omega, \mathscr{A}, \mathbf{P})$, where we can find multiple (even infinitely many) random variables. We use basic concepts of Probability Theory from the book by Loève [1955], where all important definitions and properties of the probability space can be found. Let us mention the ones, which are the most important for this thesis. A random variable is a function from the above-mentioned probability space to some target measurable space. If the target space of this function is the space of real numbers with Borel sigma-algebra, we call such random variable a *real random variable*. For example a real random variable X, which has the following description:

$$\begin{aligned} X : (\Omega, \mathscr{A}, \mathbf{P}) &\longrightarrow (\mathbb{R}, \mathscr{B}) \\ \omega &\longmapsto X(\omega). \end{aligned}$$

We will mostly use random variables with the target space $\mathbb{R}^n$, $n \in \mathbb{N}$ and call them *random vectors*:

$$X : (\Omega, \mathscr{A}, \mathbf{P}) \longrightarrow (\mathbb{R}^n, \mathscr{B}^n).$$

It is possible to consider more general random variables (*random elements*), with a general target measurable space $(S, \mathscr{S})$, which we can transform into a *real random variable* or *vector* using some measurable function $f : (S, \mathscr{S}) \longrightarrow (\mathbb{R}, \mathscr{B})$:

$$\begin{aligned} X : (\Omega, \mathscr{A}, \mathbf{P}) \longrightarrow \ & (S, \mathscr{S}) \\ f : \ & (S, \mathscr{S}) \longrightarrow (\mathbb{R}, \mathscr{B}) \end{aligned}$$

Definition 1 (Distribution function of a random vector)**.** Distribution function *of a random vector X is a monotonic non-decreasing function F defined as follows:*

$$F(\mathbf{x}) := \mathbf{P}(\{\omega : \mathbf{X}(\omega) \leq \mathbf{x}\}) = \mathbf{P}(\mathbf{X} \leq \mathbf{x}) \ \textit{for each}\ \mathbf{x} \in \mathbb{R}^n,$$

where $\mathbf{X} \leq \mathbf{x}$ is true if and inly if, for each $1 \leq i \leq n$: $X_i \leq x_i$. The Lebesgue-Stieltjes measure P on $(\mathbb{R}^n, \mathscr{B}^n)$ corresponding to the distribution function F is called the distribution *of the random vector* $\mathbf{X}$.

In the following text we use some basic definitions from the book by Anděl [1985]. We start with a Chapter II, where we can find the definition of a *marginal distribution.*

Definition 2 (Marginal distribution function)**.** *Let us have the random vector $(X_1, ..., X_n)$ which has the distribution function $F(x_1, ..., x_n)$. For each non-empty subset $K = \{k_1, ..., k_r\}$ of the set $\{1, ..., n\}$, the distribution function F_K of the random vector $(X_{k_1}, ..., X_{k_r})$ is called a* marginal distribution function*. The* marginal distribution function *can be obtained in the following way:*

$$F_K(x_{k_1}, ..., x_{k_r}) = \lim_{\mathbf{z} \to \infty} F(x_1, ..., x_n),$$

where $\mathbf{z}$ is a sub-vector of the vector $\mathbf{x} = (x_1, ..., x_n)$, such that $\mathbf{z} = (x_i)_{i \in \{1,...,n\} \setminus K}$. Symbol $\mathbf{z} \to \infty$ means, that for every $i \in \{1, ...n\} \setminus K$, $x_i \longrightarrow \infty$.

We will consider a finite set $\mathbf{V}$ of real random variables on a probability space $(\Omega, \mathscr{A}, \mathbf{P})$. Each non-empty subset $\mathbf{A} \subset \mathbf{V}$ of the set of random variables has its own marginal distribution function.

In this thesis we consider either that the random variables from the set $\mathbf{V}$ have a discrete joint distribution (in which case the distribution function has a finite number of different values) or that the variables have a continuous joint distribution (which are characterised by a density function, whose definition is stated below).

Definition 3 (Density function)**.** *If, for a random vector* $\mathbf{X} = (X_1, ..., X_n)$ *with the distribution function* $F(\mathbf{x})$*, there exists a function* $f(\mathbf{x}) = f(x_1, ..., x_n)$*, such that*

$$F(x_1, ..., x_n) = \int_{-\infty}^{x_1} ... \int_{-\infty}^{x_n} f(u_1, ..., u_n)\, du_1 ... du_n, \quad \text{for } x_1, ..., x_n \in \mathbb{R},$$

then we call f *the* joint density function *of the random vector* $\mathbf{X}$*. From the viewpoint of the Measure Theory, the density function is the Radon-Nikodym derivative of the distribution* P *of* $\mathbf{X}$ *with respect to Lebesgue measure on* $(\mathbb{R}^n, \mathscr{B}^n)$*.*

Remark. In Definition 3 we consider the Lebesgue measure on $(\mathbb{R}^n, \mathscr{B}^n)$. In order to cover also the cases, when a random vector $\mathbf{X}$ has a discrete distribution we should instead consider the counting measure on the product of (finite) sample spaces of discrete variables $X_1, ..., X_n$, which are subsets of $\mathbb{R}$. Therefore we say, that a random vector $\mathbf{X}$ has a general density function if there exists a product σ-finite measure μ such that the distribution P satisfies

$$P(A_1, ..., A_n) = \int_{u_1 \in A_1} ... \int_{u_n \in A_n} f(u_1, ..., u_n)\;\; d\mu(u_1 ... u_n)$$

for $A_1, ..., A_n \in \mathscr{B}$. Such a distribution was named *marginally continuous* in Studený [2014], section 1.3.

Some of the random variables defined on the probability space are stochastically independent, some of them might be stochastically dependent. Let us state the definition of independence, which one can be found in the book Anděl [1985], section II.2.

Definition 4 (Independence in terms of a distribution function)**.** *Let a random vector* $\mathbf{X} = (X_1, ..., X_n)$ *has a joint distribution function* $F(x_1, ..., x_n)$*. Let* $F_i(x_i)$ *denote the marginal distribution function of a variable* X_i*,* $i \in \{1, ..., n\}$*. Then* $X_1, ..., X_n$ *are independent if*

$$F(x_1, ..., x_n) = \prod_{i=1}^{n} F_i(x_i) \quad \text{for } x_1, ..., x_n \in \mathbb{R}.$$

The following definition can be found in Anděl [1985], section III.5 about conditional density.

Definition 5 (Conditional density)**.** *Let us have a random vector* $\mathbf{X} = (X_1, ..., X_n)$*. Consider* $\mathbf{Y} = (X_1, ..., X_r)$ *and* $\mathbf{Z} = (X_{r+1}, ..., X_n)$*, where* $1 \le r < n$*. Suppose, that* $\mathbf{X}$ *has the density* $p(\mathbf{x})$ *with respect to a product measure*

$\mu = \lambda \times \nu$, *where* λ *is a* σ*-finite measure on* $(\mathbb{R}^r, \mathscr{B}^r)$ *and* ν *is a* σ*-finite measure on* $(\mathbb{R}^{n-r}, \mathscr{B}^{n-r})$. Conditional density *of a random vector* $\mathbf{Y}$ *given fixed* $\mathbf{Z} = \mathbf{z}$ *is such non-negative measurable function* $r(\mathbf{y} \mid \mathbf{z})$ *that for every* $B \in \mathscr{B}^r$ *and* $C \in \mathscr{B}^{n-r}$ *the following relation is true:*

$$\mathbf{P}(B \times C) = \int_C \left[\int_B r(\mathbf{y} \mid \mathbf{z})\, d\lambda(y) \right] g(z)\, d\nu(z),$$

where $g(z)$ *is the marginal density of vector* $\mathbf{Z}$.

Remark. In the generic cases (either discrete or continuous distributions) it is assumed, that both λ and ν can be factorized into a product of one-dimensional σ-finite measures: $\lambda = \prod_{i=1}^{r} \mu_i$ and $\nu = \prod_{i=r+1}^{n} \mu_i$.

We can also describe the *conditional probability density function* in the following way:

$$f_{\mathbf{X} \mid \mathbf{Y}}(\mathbf{x} \mid \mathbf{y}) \quad = \quad \frac{f_{\mathbf{X},\mathbf{Y}}(\mathbf{x}, \mathbf{y})}{f_{\mathbf{Y}}(\mathbf{y})} \text{ if } f_{\mathbf{Y}}(\mathbf{y}) > 0, \text{ and,}$$

if $f_{\mathbf{Y}}(\mathbf{y}) = 0$, the function $f_{\mathbf{X}|\mathbf{Y}}(\mathbf{x}|\mathbf{y})$ is defined by 0 for all considered $\mathbf{x}$.

As concerns the concept of conditional independence we take over the definition for the case of marginally continuous distributions (see Studený [2014], section 1.3). In order to formulate definition of conditional independence we first need to state a definition of a marginally continuous measure and a system of dominating measures. For union of sets $\mathbf{A} \cup \mathbf{B}$ we use a shortcut $\mathbf{AB}$.

Definition 6 (marginally continuous measure)**.** *Probability measure* P *is* marginally continuous *if it is absolutely continuous with respect to a product of its one-dimensional marginals:* $P << \bigotimes_{i \in \mathbf{V}} P_i$.

Definition 7 (system of dominating measures)**.** *Every system of* $\sigma-$*finite measures* μ^i *on* $(\mathbb{R}^n, \mathscr{B}^n)$, *such that* $P << \bigotimes_{i \in \mathbf{V}} \mu^i$ *is called a* system of dominating measures *for measure* P. *Their product* $\mu = \bigotimes_{i \in \mathbf{V}} \mu^i$ *is called* the joint dominating measure for P.

Definition 8 (Conditional independence)**.** *Let* $(\mathbf{A}, \mathbf{B}, \mathbf{C})$ *be a disjoint triplet over* $\mathbf{V}$. *Let* P *be a marginally continuous measure over* $\mathbf{V}$, *specifically on* $(\mathbb{R}^n, \mathscr{B}^n)$ *(because* $|\mathbf{V}| = n$*). Let* $\{\mu^i : i \in \mathbf{V}\}$ *be a system of one-dimensional dominating measures for* P. *Then we say, that* $\mathbf{A}$ *and* $\mathbf{B}$ *are conditionally independent given* $\mathbf{C}$ *with respect to* P, *if*

$$\text{for } \mu - a.s.\ \mathbf{x} \in \mathbb{R}^n \quad f_{\mathbf{ABC}}(\mathbf{x_C}) f_{\mathbf{C}}(\mathbf{x_C}) = f_{\mathbf{AC}}(\mathbf{x_{AC}}) f_{\mathbf{BC}}(\mathbf{x_{BC}}),$$

where $f_{\mathbf{D}}$ *for* $\mathbf{D} \subset \mathbf{V}$ *denotes a version of marginal density* P *for* D *(with respect to the considered system of dominating measures) and* $\mathbf{x_D}$ *for* $\mathbf{D} \subset \mathbf{V}$ *is a sub-vector of the vector* $\mathbf{x}$, *such that* $\mathbf{x_D} = (\mathbf{x}_i)_{i \in \mathbf{D}}$.

Remark. The concept of conditional independence from the Definition 8 does not depend on the choice of the system of dominating measures. The Definition 8 is equivalent to either of the following conditions (i) or (ii).

(i)
$$f_{\mathbf{XY}|\mathbf{Z}}(\mathbf{x}, \mathbf{y} \mid \mathbf{z}) = f_{\mathbf{X}|\mathbf{Z}}(\mathbf{x} \mid \mathbf{z})\, f_{\mathbf{Y}|\mathbf{Z}}(\mathbf{y} \mid \mathbf{z}),$$

(ii)
$$f_{\mathbf{X}|\mathbf{YZ}}(\mathbf{x} \mid \mathbf{y}, \mathbf{z}) = f_{\mathbf{X}|\mathbf{Z}}(\mathbf{x} \mid \mathbf{z}),$$

where $\mathbf{X}, \mathbf{Y}, \mathbf{Z}$ are random vectors, equations are valid almost surely, the symbol $f_{\mathbf{A}}$ for $\mathbf{A} \subset \mathbf{X} \cup \mathbf{Y} \cup \mathbf{Z}$ denotes the marginal density of $\mathbf{A}$ conditional on a vector $\mathbf{Z}$.

Remark. We write $(\mathbf{X} \perp\!\!\!\perp \mathbf{Y} \mid \mathbf{Z})$ if a random vector $\mathbf{X}$ is conditionally independent of the random vector $\mathbf{Y}$ given $\mathbf{Z}$. If the opposite is true we write $(\mathbf{X} \not\!\perp\!\!\!\perp \mathbf{Y} \mid \mathbf{Z})$ and say that $\mathbf{X}$ is conditionally dependent on $\mathbf{Y}$ given $\mathbf{Z}$.

Every joint distribution of a collection of real random variables defines a certain structure of independences and conditional independences for the given random variables. In this thesis we explore representations of these structures and find some relations between different representations.

Definition 9 (Conditional independence structure)**.** Conditional independence structure *defined by the joint probability distribution* $\mathbf{P}$ *of a set of random variables* $\mathbf{V}$ *is the set of all triplets* $(\boldsymbol{X}, \boldsymbol{Y} | \boldsymbol{Z})$ *with a following property,* $\boldsymbol{X}, \boldsymbol{Y}, \boldsymbol{Z}$ *are disjoint subsets of* $\boldsymbol{V}$ *such that* $(\boldsymbol{X} \perp\!\!\!\perp \boldsymbol{Y} | \boldsymbol{Z})$ *with respect to* $\mathbf{P}$*.*
The structure is denoted by $C(\mathbf{V})$*.* [1]

It is possible to reconstruct the set of all of triplets in a CI structure from a subset of the so-called elementary triplets. Their definition and the following Lemma are stated and proved in lecture notes by Studený Studený [2014], section 2.3.

Lemma 1 (Characterization of a CI structure)**.** *The whole CI structure can be reconstructed from the so-called* elementary triplets*, where there is only one variable on the first and on the second place:* $|\mathbf{X}| = |\mathbf{Y}| = 1$*. Specifically, the CI structure is reconstructible as follows* $(\mathbf{X} \perp\!\!\!\perp \mathbf{Y} | \mathbf{Z})$ *iff*

$$\forall X \in \mathbf{X} \ \ \forall Y \in \mathbf{Y} \ \ \forall \mathbf{W} \ \textit{with}\ \mathbf{Z} \subset \mathbf{W} \subset \mathbf{X} \cup \mathbf{Y} \cup \mathbf{Z} \setminus (X \cup Y) \ \textit{one has}\ (X \perp\!\!\!\perp Y | \mathbf{W}).$$

1.2 Graphical concepts

There are several ways to represent a CI structure in the memory of a computer. One option is to save the whole list of triplets which form the CI structure. But this way is very demanding on the memory of a computer and it is hard to verify if some particular triplet is present in the structure. Another option is to use some representation, where triplets of a CI structure are encoded in a specific way. One of those representations is the Directed Acyclic Graph, also called *Bayesian*

[1] In this thesis we use a shortcut CI for the term conditional independence.

network. The other representations are, for example, a simple undirected graph, an annotated graph or an imset.

Let us give some graphical definitions, which we will need throughout the thesis.

Definition 10 (Graph). *A* graph *G over a non-empty set $\mathbf{V}$ is an ordered pair $(\mathbf{V}, E)$ where $\mathbf{V}$ is called the set of* nodes*, while E is the set of* edges*, which are ordered pairs of nodes (v, u), $u, v \in V$ and $u \neq v$.*

Remark. Since we consider in E only pairs (u, v), where $u \neq v$, our graphs will be without loops. Also, we suppose that the number of nodes, the cardinality of the set $\mathbf{V}$, is finite.

Definition 11 (Types of edges). *We distinguish two types of edges:*

undirected edge *An edge (u, v) is* undirected *if $(u, v) \in E$ and $(v, u) \in E$. We write u — v and say, that u, v are* neighbours/ neighbouring nodes.

directed edge *An edge (u, v) is directed if either $[(u, v) \in E \,\&\, (v, u) \notin E]$ or $[(v, u) \in E \,\&\, (u, v) \notin E]$. If $(u, v) \in E$ and $(v, u) \notin E$ we write $u \longrightarrow v$ and say, that u is a* parent *of v, while v is a* child *of u.*

Definition 12 (Neighbours, children, parents). *For each node $v \in \mathbf{V}$ and for each set of nodes $A \subset \mathbf{V}$ we introduce the following concepts:*

neighbours *We denote this set of nodes as $ne_G(v)$ or $ne_G(A)$ and define in the following way: $ne_G(v) := \{u \in V : u — v\}$, whereas $ne_G(A) := \bigcup_{u \in A} ne_G(u) \setminus A$;*

children *This set of nodes is denoted by $ch_G(v)$ or $ch_G(A)$ respectively, where $ch_G(v) := \{u \in V : v \longrightarrow u\}$, $ch_G(A) := \bigcup_{v \in A} ch_G(v) \setminus A$;*

parents *The set of parents is denoted by $pa_G(v)$ or $pa_G(A)$ and defined as follows:*
$pa_G(v) = \{u \in V : u \longrightarrow v\}$, $pa_G(A) := \bigcup_{v \in A} pa_G(v)$

We will refrain from mentioning the graph G in the definition of neighbours, children or parents when it is obvious, which graph G we have in mind. Thus, for example, instead of $pa_G(u)$ we may write $pa(u)$.

Definition 13 (Subgraph). *A graph $\tilde{G} = (\tilde{\boldsymbol{V}}, \tilde{E})$ is a subgraph of a graph $G = (\mathbf{V}, E)$, if $\tilde{\boldsymbol{V}} \subset \boldsymbol{V}$, $\tilde{E} \subset E$ and all of the edges in $\tilde{G}$ are of the same type as in G.*

Definition 14 (Induced subgraph). *An* induced subgraf *$\widehat{G} = (\widehat{\boldsymbol{V}}, \widehat{E})$ of a graph $G = (\boldsymbol{V}, E)$ given by a non-empty set of nodes $\widehat{\boldsymbol{V}} \subset \boldsymbol{V}$ and the set of edges is determined in the following way: $\widehat{E} = E \cap \left(\widehat{\boldsymbol{V}} \times \widehat{\boldsymbol{V}}\right)$.*

Definition 15 (Path, cycle). *We will distinguish several types of paths in a graph $G = (\boldsymbol{V}, E)$:*

an undirected path *is a sequence* $(v_i)_{i=1}^n$, $n \geq 1$ *of distinct nodes from* $\boldsymbol{V}$ *such that, for every* $i \in \{1, ..., n-1\}$*, it holds that* $v_i \longrightarrow v_{i+1}$*. The length of such path is the number of edges in the path, namely* $n-1$*;*

a semidirected path *from node* v_1 *to node* v_n*,* $n \geq 1$*, is a sequence of distict nodes from* $\mathbf{V}$ *such that, for every* $i \in \{1, ..., n-1\}$*, it holds that* $(v_i, v_{i+1}) \in E$ *and there exists* $j \in \{1, ..., n-1\}$ *such that* $(v_{j+1}, v_j) \notin E$*, that is* $v_j \longrightarrow v_{j+1}$ *and for* $i \neq j$*,* $1 \leq i \leq n-1$ *either* $v_i \longrightarrow v_{i+1}$ *or* $v_i \longrightarrow v_{i+1}$*;*

a directed path *is a sequence* $(v_i)_{i=1}^n$*,* $n \geq 1$*, of distinct nodes from* $\boldsymbol{V}$ *such that, for every* $i \in \{1, ..., n-1\}$*, it holds that* $v_i \longrightarrow v_{i+1}$*;*

an undirected cycle *is an* undirected path *except that* $n \geq 4$ *and there are two equal nodes in the sequence, namely* $v_1 = v_n$*;*

a semidirected cycle *is a* semidirected path *except that* $n \geq 4$ *and there are two equal nodes in the sequence, namely* $v_1 = v_n$*;*

a directed cycle *is a* directed path *except that* $n \geq 4$ *and there are two equal nodes in the sequence, namely* $v_1 = v_n$*.*

Remark. We denote a path $(v_i)_{i=1}^n$ by π and the length of the path π by $l(\pi)$. In this case $l(\pi) = n-1$.

Definition 16 (graph skeleton)**.** *A* skeleton *of a graph* $G = (\mathbf{V}, E)$ *is an undirected graph* $U_G = (\mathbf{V}, E')$*, such that if* $(v, w) \in E$ *then both* $(v, w) \in E'$ *and* $(w, v) \in E'$*. In other words we forget all directions of edges.*

Definition 17 (ancestor, descendant)**.** *Let* $G = (\mathbf{V}, E)$ *be a graph, then*

an ancestor *node* $a \in \mathbf{V}$ *is an* ancestor *of a node* $d \in \mathbf{V}$ *in* G *if there exists a directed path from* a *to* d *in* G*;*

a descendant *node* $d \in \mathbf{V}$ *is a* descendant *of a node* $a \in \mathbf{V}$ *in* D *if* a *is an ancestor of* d *in* G*.*

Remark. Definition 17 takes into account paths which consist of a single node (for $n = 1$). As a consequence, each node is its own ancestor and descendant.

Remark. On the other hand, when G is a Directed Acyclic Graph, a node v can not be both an ancestor and a descendant of a different node w (that would mean that there is at least one directed cycle in D).

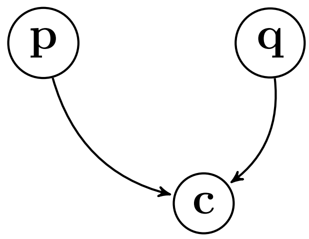
The following definition was mentioned in S. A. Andersson [1977], Appendix A.

Definition 18 (Chain graph)**.** *A graph* $G = (\mathbf{V}, E)$ *is a* chain graph *if it contains no semidirected cycles.*

Definition 19 will be very useful throughout this thesis, the concept of Directed Acyclic Graph is mainly based on the following induced subgraph called an immorality.

Definition 19. *(Immorality in a graph) A triplet of distinct nodes (p, q, c) in a graph G is called an* immorality *in G if there is no edge between the nodes p and q and the node c is a common child of the nodes p and q. In other words, $p \longrightarrow c \longleftarrow q$ is an induced subgraph of D. In this case, the nodes p and q are called the* unmarried parents *or the* unmarried couple*.*

Figure 1.1: Illustration of the immorality (p, q, c) with unmarried parents p and q

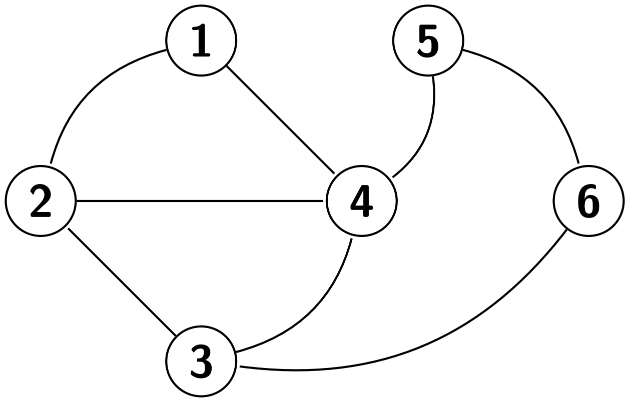


2. Graphical models of conditional independence structures

2.1 Undirected graphs

Definition 20 (undirected graph - UG). *An* undirected graph $U = (\mathbf{V}, E)$ *is a graph where every edge from* E *is undirected.*

Figure 2.1: Example of a UG G



Definition 21 (cut set in a UG). *Let us have an undirected graph* $U = (\mathbf{V}, E)$ *and some disjoint subsets of its set of nodes:* $\mathbf{X}, \mathbf{Y}$ *and* $\mathbf{Z}$. *The set* $\mathbf{Z}$ *is called a* cut set *between* $\mathbf{X}$ *and* $\mathbf{Y}$ *if every undirected path between any node from* $\mathbf{X}$ *and any node from* $\mathbf{Y}$ *has non-empty intersection with* $\mathbf{Z}$. *A* minimal cut set $\mathbf{Z}$ *between* $\mathbf{X}$ *and* $\mathbf{Y}$ *is a cut set, such that for every* $\mathbf{W} \subsetneq \mathbf{Z}$, $\mathbf{W}$ *is not a cut set between* $\mathbf{X}$ *and* $\mathbf{Y}$.

Undirected graphs are used to represent CI structures $C(\mathbf{V})$ implied by a probability distribution. Let us assume that we already have some conditional independence structure $C(\mathbf{V})$. An undirected graph U which represents $C(\mathbf{V})$ is built over a set of random variables $\mathbf{V}$, which means that every node of the graph represents a random variable from $\mathbf{V}$. The question is how to check if a certain triplet from $C(\mathbf{V})$ is represented in U. Due to the Lemma 1 it is enough to consider only elementary triplets from $C(\mathbf{V})$. The following Definition 22 can be extended to the whole CI structure, which consists of general (not only elementary) triplets.

Definition 22 (UG criterion). *Let us have two different nodes* $X, Y \in \mathbf{V}$ *and a disjoint set of nodes* $\mathbf{Z} \subset \mathbf{V}$. *A triplet* $\langle X, Y | \mathbf{Z} \rangle$ *from CI structure* $C(\mathbf{V})$ is represented *in an undirected graph* $U = (\mathbf{V}, E)$ *if* $\mathbf{Z}$ *is a cut set (not necessarily minimal) between* X *and* Y.

For example in Figure 2.1 a set $\mathbf{Z} = \{3, 5\}$ is a *minimal cut set* between two sets of nodes, $\mathbf{X} = \{2\}$ and $\mathbf{Y} = \{6\}$. That means, that the triplet $\langle \mathbf{X}, \mathbf{Y} | \mathbf{Z} \rangle$ is represented in graph G using the UG criterion from the Definition 22.

The following facts come from Studený [2014], section IV. A graph can represent a CI structure in two ways: fully and partially. In the first case every triplet from a CI structure is represented in the graph, in the second case some information about it is lost: some triplets, which can be found in the structure are not represented in the graph. In both cases of representation, triplets which do not belong to the CI structure are not represented in the graph.

Definition 23 (UG representation of a CI structure). *An undirected graph $U = (\mathbf{V}, E)$ represents conditional independence structure $C(\mathbf{V})$ given by a probability distribution*

> ***fully*** [1] *if the set of triplets* represented *in U as in Definition 22 is exactly the same as the set of triplets in $C(\mathbf{V})$,*
>
> ***partially*** [2] *if the set of triplets* represented *in U as in Definition 22 is a subset of the set of triplets in $C(\mathbf{V})$.*

Not every CI structure can be represented fully by a UG. Therefore, we can only approximate such CI structure by undirected graphs, which represent them partially.

On the other hand, for every undirected graph $U = (\mathbf{V}, E)$, there always exists a probability distribution over the set of random variables $\mathbf{V}$, which defines the CI structure induced by the graph U. See D. Geiger [1993].

2.2 Directed acyclic graphs

There are other ways to represent CI structures. We can use Directed Acyclic Graphs instead of undirected graphs. These graphical models of CI structures are named *Bayesian networks* and are widely used in such fields as artificial intelligence. The set of CI structures representable by directed acyclic graphs is different from the one representable by undirected graphs (although their intersection is not empty). On the other hand, there still exist some CI structures which can be represented neither by undirected graphs nor by directed acyclic graphs.

Definition 24 (Directed graph). *A* directed graph $D = (\mathbf{V}, E)$ *is a graph, where each edge $(u, v) \in E$ is directed.*

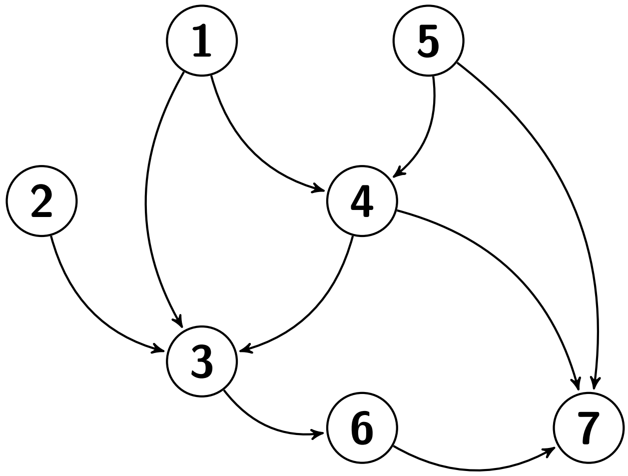
Conditional independence structures are represented by a more restricted class of directed graphs: directed acyclic graphs. It is appropriate to note, that the name is misleading in a certain way: precisely it should have been acyclic directed graphs. Then the meaning could be quite intuitive, because we will consider directed graphs and forbid directed cycles in them. Nevertheless, the term 'directed acyclic graph' has become widely accepted in the community of graphical models.

[1] The terminology may vary here - most common name for the probability distribution, which can be represented fully by a UG is *perfectly Markovian distribution*.

[2] A distribution, which can be represented by a UG partially is also called *Markovian distribution*.

Definition 25 (Directed Acyclic Graph - DAG)**.** *A Directed Acyclic Graph* $D = (\mathbf{V}, E)$ *is a directed graph where there are no directed cycles.*

Figure 2.2: Example of a Directed Acyclic Graph D

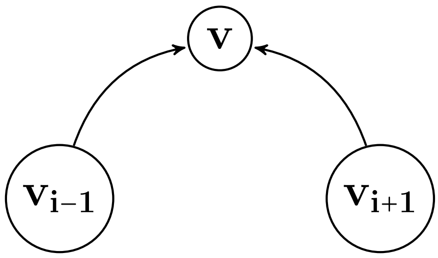


As mentioned before, DAGs are also used to represent CI structures. Similarly to undirected graphs, each node in a DAG represent a random variable. It is possible to decide in multiple ways whether a certain triplet from a given CI structure is represented in a DAG. We will give two criteria. The first criterion is based directly on the original DAG, whereas the second one needs to modify the DAG into an undirected graph and then uses the criterion described in Definition 22 from section 2.1.

We will need the following definition in order to formulate the first criterion.

Definition 26 (collider)**.** *Let us have a directed acyclic graph* $D = (\mathbf{V}, E)$. *Then we say that a node* v *is a* collider on a path $(v_i)_{i=1}^n$ *if there exists an index* $i \in \{2, ..., n-1\}$ *such that* $v_i = v$ *and* $v_{i-1} \longrightarrow v_i, v_{i+1} \longrightarrow v_i$.

Figure 2.3: Illustration of the collider on the path



Definition 27 (DAG criterion 1)**.** *A triplet* $\langle X, Y | \mathbf{Z} \rangle$ *from CI structure* $C(\mathbf{V})$ *is represented in a DAG* $D = (\mathbf{V}, E)$ *if on every path* $(v_i)_{i=1}^n$ *between* X *and* Y*, there is a node* v*, which is one of two sorts:*

- v *is a* collider *on the path* $(v_i)_{i=1}^n$ *and there is no descendant of the node* v *which belongs to* $\mathbf{Z}$*,*
- v *is not a* collider *on the path* $(v_i)_{i=1}^n$ *and* v *belongs to* $\mathbf{Z}$*.*

Remark. Criterion described in Definition 27 is sometimes called *d-separation criterion* and it was introduced by Pearl. The *d-separation criterion* can be found for example in the book Studený [2014], section 5.

There is another option, which includes modifying DAG in a certain way and erasing direction of arrows. To modify a DAG in the right way, we will have to detect certain structures (induced subgraphs) called immoralities and introduced in the Definition 19.

Definition 28 (DAG criterion 2)**.** *A triplet* $\langle X, Y | \mathbf{Z} \rangle$ *from a CI structure* $C(\mathbf{V})$ *is represented in a directed acyclic graph* D *if it is represented in the undirected graph* $U = (\mathbf{V}', E')$*, where* U *is obtained by the following construction:*

(i) let $\mathbf{V}'$ *be the set of ancestors of the nodes* $\{X \cup Y \cup \mathbf{Z}\}$*; consider the subgraph of* D *induced by the set* $\mathbf{V}'$ *denoted as* D'*,*

(ii) moralize immoralities: connect nodes in D'*, which are unmarried parents with an undirected edge,*

(iii) erase all directions (consider the skeleton of a graph D' *after the step (ii)). The resulting graph is the undirected graph* U*.*

Remark. The criterion described in the Definition 28 is also called Lauritzen criterion, in the article Paz [2003b] it is described as the algorithm L1.

Remark. Two criteria described in Definition 27 and in Definition 28 are equivalent. This has been proven in the lecture notes Studený [2014] in the Lemma 5.7 (chapter Bayesian networks). The first criterion from Definition 27 is more suitable if we want to confirm that a certain triplet *is not* represented in a DAG, the second criterion from the Definition 28 is useful for showing that a triplet *is* represented.

2.3 Equivalence of directed acyclic graphs

Definition 29 (Markov equivalent DAGs). *Two DAGs D and $\tilde{D}$ are* Markov equivalent *if the sets of triplets represented in both graphs are identical. If D and $\tilde{D}$ are Markov equivalent, we write $D \simeq \tilde{D}$.*

There are several ways to decide whether two DAGs are Markov equivalent. The Theorem 3, which is based on former results in the theory of Bayesian networks, gives us two options; it was formulated and proven in Studený [2014], section 5.5. In order to state the observation we need to formulate the definition of a reversible arrow.

Definition 30 (Reversible Arrow in a DAG). *Let us have a directed acyclic graph $D = (\mathbf{V}, E)$ and an arrow $a \longrightarrow b$ which belongs to D. An arrow $a \longrightarrow b$ is* reversible *if the directed graph $\tilde{D} = (\mathbf{V}, \tilde{E})$, where $\tilde{E} = (E \setminus (a,b)) \cup (b,a)$, has exactly the same immoralities as D and doesn't contain directed cycles.*

There exist several characterizations of the concept described in Definition 30. They are useful in different situations. The following characterization is formulated in Studený [2014], Lemma 5.22.

Theorem 2. (Characterization of the reversibility) *An arrow $a \longrightarrow b$ is reversible in a DAG D if and only if $pa_D(a) = pa_D(b) \setminus a$.*

The following claim from Studený [2014] Statement 5.23, gathers different characterisations of Markov equivalence for DAGs.

Theorem 3. (Characterization of DAG equivalence) *Let us have two DAGs D and $\tilde{D}$. Then the following statements are equivalent:*

- *D and $\tilde{D}$ are Markov equivalent ($D \simeq \tilde{D}$),*
- *There exists a sequence of DAGs: $D_1, ..., D_n, n \geq 1$ with a following property. The first graph D_1 is the original DAG D, the last one D_n is $\tilde{D}$. For every $i \in \{1, 2, ..., n-1\}$ it holds that D_{i+1} is obtained from D_i by changing the direction of one of the reversible arrows,*
- *D and $\tilde{D}$ have the same skeleton and immoralities.*

2.4 Essential graphs

The set of all DAGs can be divided into equivalence classes using the relation of *Markov equivalence* defined in the Definition 29. Let us denote the class containing a DAG D by $[D]$. We can decide whether two DAGs are *Markov equivalent* using one of the criteria from the Theorem 3 - Characterization of DAG equivalence. Moreover, there is a way to represent each equivalence class by a special graph called the *essential graph of the equivalence class. The essential graph* is a hybrid graph, which means that it may have both directed and undirected edges. Its definition taken from the paper of Andersson, Madigan and Perlman S. A. Andersson [1977] is stated below.

Definition 31 (Essential graph)**.** *The* essential graph D^* *associated with DAG* D *is the graph*

$$D^* := \bigcup (D' \mid D' \simeq D)\,,$$

that is, D^* *is the smallest hybrid graph larger than every* D' *in* $[D]$ *(symbol* $\bigcup$ *stands for supremum of graphs with respect to the ordering by larger).*
Here, if $H = (\mathbf{V}, E_1)$ *and* $G = (\mathbf{V} E_2)$ *are hybrid graphs* H is larger *than* G *if* $E_2 \subset E_1$*.*

For every DAG D let us also denote a class of equivalence $[D]$ by $[D^*]$, where D^* is an essential graph corresponding to D. This notation will simplify future arguments.

Each *essential graph* represents a Markov equivalence class of DAGs and for each class only one such representative exists. It follows from Definition 31 that an edge e in the *essential graph* D^* is directed if and only if in each DAG D in the class $[D^*]$ this particular edge is equally directed. Such arrow is called *essential.* On the other side, if there exist two DAGs D and D' in the equivalence class $[D^*]$, such that in D an edge e between two nodes a and b is directed form a to b and it is directed from b to a in D' then the edge e is undirected in the *essential* graph D^*.

In the paper S. A. Andersson [1977] a characterization of graphs that are essential graphs is given. The characterization gives a necessary and sufficient condition for a graph $G = (\mathbf{V}, E)$ to be the essential graph D^* for some class $[D^*]$ of equivalent DAGs (in this case we are talking about a hybrid graph). The following Theorem is a corollary of the Theorem 4.1 in S. A. Andersson [1977].

Theorem 4 (Properties of the essential graph D^*)**.** *If a graph* $G = (\mathbf{V}, E)$ *equals to* D^* *for some DAG* D *then* G *satisfies the following conditions.*

(i) G *is a chain graph;*

(ii) No configuration $a \longrightarrow b —— c$ *(called a* flag*) occurs as an induced subgraph of* G*;*

2.5 Annotated graphs

Here we introduce the concept of an *annotated graph*, which is the main topic of this thesis. The annotated graphs can also be used as an efficient representation of CI structures. The concept of an annotated graph does not fit into the general definition of the graph stated Section 1.1, but its basis it comes from the concept of the undirected graph. Each edge in an annotated graph is undirected, but some of the edges have an annotation upon them. The annotation is a set of nodes, which also belong to the node set of the graph. Therefore, an annotated graph is basically an undirected graph, where some of the edges are paired with certain sets of nodes from the graph. Let us give a formal definition of an annotated graph, which uses the term of an *element.*

Definition 32 (Element)**.** *Let us have an undirected graph* $G = (\mathbf{V}, E)$*. An element* e *in this graph is a triplet* $(p, q, \mathbf{S})$*, where* $p, q \in \mathbf{V}$*,* $p —— q$ *in* G *and* $\mathbf{S} \subset \mathbf{V}$ *is a set of nodes distinct from* p *and* q*.*

Definition 33 (Annotated graph)**.** *An* annotated graph *is a pair* $A = (U_A, N_A)$ *where* $U_A = (\mathbf{V}, E)$ *is an undirected graph and* N_A *is a set of elements of* U_A*.*

Figure 2.4: Example of an annotated graph $A = (U_A, N_A)$

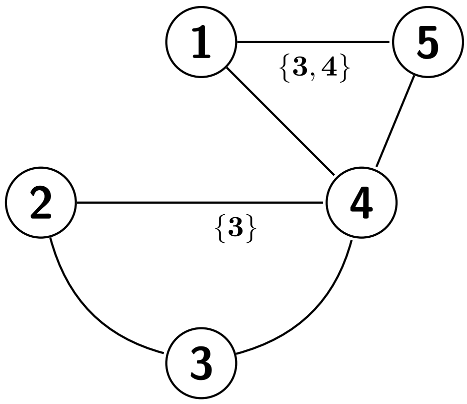
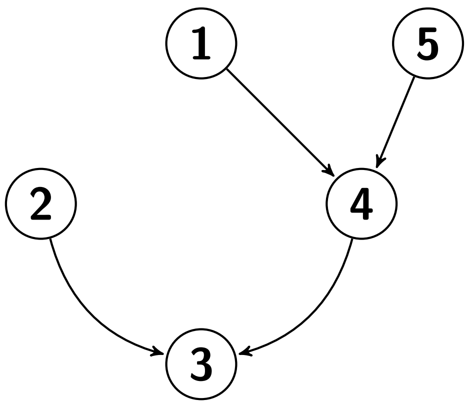


Figure 2.5: A DAG D which corresponds to the annotated graph $A = (U_A, N_A)$



Annotated graphs as well as DAGs and UGs can represent conditional independence structures. Nodes from an undirected graph U_A represent random variables. In the paper Paz [2003b] annotated graphs were used to introduce an alternative version of the algorithm described in Definition 28. It was stated there, that with a preconditioning procedure of transforming DAG into an annotated graph, the task of deciding whether a given triplet is represented in a DAG is faster, on the average, when compared with the other algorithms. The algorithm provides us with a way to get an annotated graph from a DAG which would represent the same CI structure.

Algorithm 1. (DAG to Annotated graph transformation) [3]
Input: a DAG $D = (\mathbf{V}, E)$*;*
Output: an annotated graph $A = (U_A, N_A)$*;*
The steps are as follows:

[3]In Paz [2003b] there is a slightly complicated version of this algorithm, which cuts off parts of annotations. It is called the algorithm L2 - preconditioning

(i) *Construct a UG out of DAG D, to be denoted by U_A, by connecting all unmarried parents by an edge and removing all directions;*

(ii) *Construct the set N_A as a union of elements created as follows: if v, w is an unmarried couple in D then construct an element $(v, w, \mathbf{S})$, where $\mathbf{S}$ consists of all common children of the nodes v and w and of all descendants of those children.*

The following algorithm provides us with a way to decide if a certain independence triplet $\langle X, Y | \mathbf{Z}\rangle$ is represented in annotated graph $A = (U_A, N_A)$.

Algorithm 2. (criterion - annotated graph)
Input: *an annotated graph $A = (U_A, N_A)$ and a triplet of disjoint subsets of nodes $\mathbf{X}, \mathbf{Y}, \mathbf{Z}$.*
Output: *the decision whether $\langle \mathbf{X}, \mathbf{Y} | \mathbf{Z}\rangle$ is represented in the annotated graph A.*

(i)
0. *Denote $\mathbf{T} = \mathbf{X} \bigcup \mathbf{Y} \bigcup \mathbf{Z}$.*
1. *Find unmarked elements from N_A and put them in a stack. If no such elements exist, go to 3.*
2. *While the stack is not empty do*
   begin
   (a) *Remove an element $e = (p, q, \mathbf{S})$ from the stack.*
   (b) *If $\mathbf{S} \bigcap \mathbf{T} = \varnothing$ then (process the element)*
      begin
      - *Remove the nodes in $\mathbf{S}$ from U and remove incident edges. If some of the nodes or edges have been removed in a previous iteration, then remove only the remaining parts.*
      - *Disconnect in U the pair of nodes $\{p, q\}$.*
      - *Remove the element $(p, q, \mathbf{S})$ from N_A.*

      end.
   (c) *Else mark $(p, q, \mathbf{S})$* end.

   Go to 1.
3. *Halt.*

(ii) *A triplet $\langle \mathbf{X}, \mathbf{Y} | \mathbf{Z}\rangle$ is represented in an annotated graph A if it is represented in the sense of Definition 22 in the UG U generated by the (i).*

Theorem 5. *Given a DAG D, if A is the annotated graph obtained by Algorithm 1 from D, then the set of triplets represented in the DAG D is exactly the same as the set represented in the annotated graph A using the criterion described in Algorithm 2.*

Theorem 5 was proven in Paz [2003b].

3. Hypothesis of equivalence

In the following section we will prove, that an annotated graph represents a Markov equivalence class of directed acyclic graphs. This means that annotated graphs corresponding to two Markov equivalent DAGs are identical and, for each annotated graph, the set of corresponding DAGs is the equivalence class. This idea has been introduced in paper Paz [2003b], where he offered an indirect proof (using conditional independence structures) for one of the implications mentioned above. In this thesis we give a direct (graphical) proof of both implications.

Definition 34 (skeleton for an annotated graph)**.** *The* skeleton *of an annotated graph $A = (U_A, N_A)$ is the undirected graph U_A. In other words, we forget the annotations.*

Theorem 6. *Let us have two DAGs D and $\tilde{D}$ which are Markov equivalent. Then the annotated graphs A and $\tilde{A}$ which correspond to each DAG (in the sense of Algorithm 1) are identical, that means, that they have the same nodes, edges and annotations.*

Proof. We will apply Theorem 3 to simplify the task of verifying the statement of Theorem 6. Instead of proving the result for two general DAGs D and $\tilde{D}$ we can consider, without loss of generality, two DAGs which differ only in the direction of one reversible arrow.

Let us look into the process of the construction of the annotated graph A corresponding to some DAG D. The nodes of the annotated graph originate from the given DAG D, whereas the edges in A are of two sorts:

- the edges present in D,
- the edges between parents in immoralities in the graph D.

The edges of the first sort are identical for the graphs A and $\tilde{A}$, because they come from DAGs $D, \tilde{D}$ which only differ in direction of one arrow. The edges of the second sort are not present in either D or $\tilde{D}$. But since D and $\tilde{D}$ have the same immoralities, the pairs of nodes to be connected and annotated are the same in both cases of A and $\tilde{A}$. Therefore, the skeletons of both annotated graphs are identical.

The only component in which they could differ is the annotation for the edges connecting unmarried parents. Let us fix an annotated edge between nodes p and q. This edge (p, q) is present in both annotated graphs A and $\tilde{A}$, but we do not know yet, if the annotations for this edge are identical in A and in $\tilde{A}$. Let us denote the set of common children (of nodes p and q) and their descendants as $\mathbf{S}$ for the first graph A and similarly $\tilde{\mathbf{S}}$ for the second graph $\tilde{A}$.

Let node c be a common child of nodes p and q in D. Then the triplet (p, q, c) is an immorality in D. Since D and $\tilde{D}$ are equivalent, they have the same immoralities (due to the Theorem 4). Therefore, (p, q, c) is an immorality also in the graph $\tilde{D}$, which means, that c is a common child of p and q in $\tilde{D}$.

Let us assume, that node a belongs to $\mathbf{S}$, but it is not a common child of p and q in D. Then, in the DAG D, there must exist a common child of nodes p

and q whose descendant is the node a. We can denote it as c and know that it is a common child of p and q in the graph $\tilde{D}$ as well.

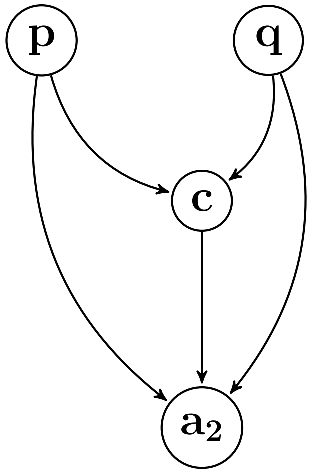
Since a is a descendant of c, we also know that there exist some directed paths in the DAG D from c to a. We can choose the shortest one from these paths (a path with the smallest number of edges):

$$\pi : (c = a_1) \longrightarrow a_2 \longrightarrow \ldots \longrightarrow (a_n = a).$$

If $\tilde{D}$ differs from D in the direction of an edge between a_i and a_{i+1} for some $i \in \{1, \ldots, n-1\}$ and there is no edge between a_{i-1} and a_{i+1} then we get a contradiction as described below.

If $i = 1$, then the node a is child of a common child c of nodes p and q. If $\tilde{D}$ differs from D in the direction of the edge between c and a, then to avoid immoralities (p, a, c) and (q, a, c) in $\tilde{D}$, which are not present in D, there must be two edges between p and a and between q and a in both DAGs. There are no directed cycles in D, therefore $p \longrightarrow a$ and $q \longrightarrow a$. But that means, that a is a common child of p and q in D, which is a contradiction with our assumption.

Figure 3.1: Illustration of the case, when a_2 is a child of $c = a_1$



In the case when $i \geq 2$ we proceed as follows. Since no other edge in $\tilde{D}$ can have a different direction than the one in D, there is an immorality $a_{i-1} \longrightarrow a_i \longleftarrow a_{i+1}$ in $\tilde{D}$, which is not present in D. It is a contradiction, because D and $\tilde{D}$ are equivalent, therefore there must be an edge between a_{i-1} and a_{i+1}.

There are two possible directions of an edge in the graph D between a_i and a_{i+1}, for $i \geq 1$ we have:

- An arrow $a_{i-1} \longleftarrow a_{i+1}$. However, this arrow can not be present in D, because D has no directed cycles. See Figure 3.2.

- An arrow $a_{i-1} \longrightarrow a_{i+1}$. If this arrow is present in D, then there exists a path between c and a which has less number of edges than the one that we picked at the first place. But this fact is a contradiction with the assumption that the path π has the smallest possible length in D. We can shorten it by using an arrow $a_{i-1} \longrightarrow a_{i+1}$ instead of two arrows $a_{i-1} \longrightarrow a_i \longrightarrow a_{i+1}$. See Figure 3.3

Figure 3.2: Illustration of the first option: directions of edges in D.

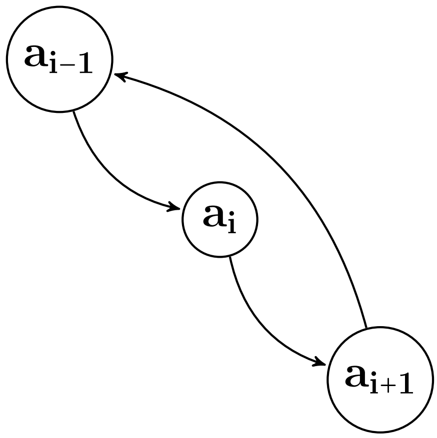
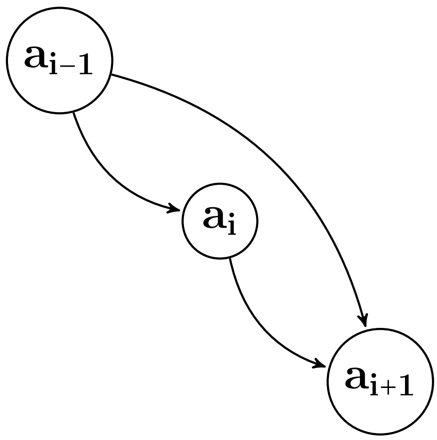


Figure 3.3: Illustration of the second option: directions od edges in D.



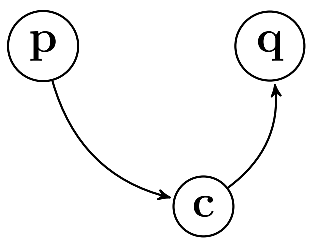
Therefore, the directions of all arrows in the path π in $\tilde{D}$ are the same as in D. This means that in $\tilde{D}$ we can also find a directed path from c to a and the node c is a common child in the graph $\tilde{D}$ as well as in D. Therefore, $a \in \tilde{\mathbf{S}}$ and we have that $\mathbf{S} \subset \tilde{\mathbf{S}}$. We can prove by a symmetrical argument that $\tilde{\mathbf{S}} \subset \mathbf{S}$. Therefore, the graphs A and $\tilde{A}$ have the same annotations: $N_A = N_{\tilde{A}}$. □

Theorem 7. *Let us have two different DAGs D and $\tilde{D}$, by the procedure 1 we get the corresponding annotated graphs A and $\tilde{A}$. If these two graphs are identical, then the original DAGs D and $\tilde{D}$ are Markov equivalent.*

Proof. Since the annotated graphs are identical, let us denote the unique annotated graph which represents both DAGs as A. Due to Theorem 3 it is enough to show that the graphs D and $\tilde{D}$ have the same skeleton and immoralities. Due to the construction 1 there exists a bijection between non-annotated edges in the graph A and the edges present in the graph D. The same is true for the other graph $\tilde{D}$, therefore, both DAGs have the same edges.

Let us fix an immorality in the graph D to prove that it is also an immorality in $\tilde{D}$. Denote the immorality by $p \longrightarrow c \longleftarrow q$, where p and q are parents and c is their common child. First notice that there is no edge between p and q in graph $\tilde{D}$. Let us assume though that the triple (p, q, c) is not an immorality in graph $\tilde{D}$. Without loss of generality we can assume, that an arrow $c \longrightarrow q$ is present in $\tilde{D}$ (see Figure 3.4).

Figure 3.4: An induced subgraph of a graph $\tilde{D}$.



Since $c \in \mathbf{S}$, where $\mathbf{S}$ is the annotation of the edge (p, q) in A, in $\tilde{D}$ there must exist a common child d of nodes p and q such that c is a descendant of d. Therefore, there also exist a directed path from d to c in $\tilde{D}$:

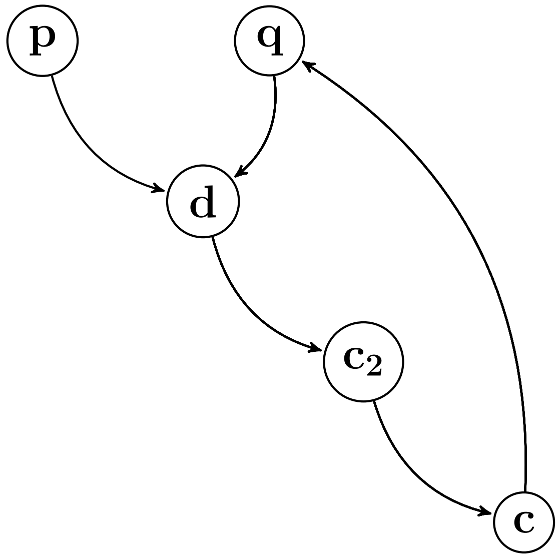
$$\pi : d = c_1 \longrightarrow \ldots \longrightarrow c_n = c.$$

We can use this path to see, that in this situation, there is actually a directed circle in $\tilde{D}$ starting and finishing in the node q:
$q \longrightarrow d = c_1 \longrightarrow \ldots \longrightarrow c_n = c \longrightarrow q$. Which leads to a contradiction with the assumption that $\tilde{D}$ is acyclic.

As a consequence, an arrow $c \longleftarrow q$ must be present in the DAG $\tilde{D}$. Using a symmetrical argument $p \longrightarrow c$ is also in $\tilde{D}$. Hence we can find an *immorality* $p \longrightarrow c \longleftarrow q$ from D also in $\tilde{D}$. Therefore, each immorality present in D can be also found in $\tilde{D}$. Using the same reasoning, we can show that every immorality from $\tilde{D}$ is also in D. The edges and immoralities are the same in both graphs D and $\tilde{D}$; therefore, they are Markov equivalent.

□

Figure 3.5: An illustration of a cycle.



Remark. Theorem 7 has been proven in the article Paz [2003b] in a different way. Paz detected all triples from the CI structure which were represented in an annotated graph A and concluded that since both DAGs D and $\tilde{D}$ represent the same set of triplets, they must be Markov equivalent. The Theorem was also mentioned in the paper Paz [2003a].

Theorems 6 and 7 give us an interesting result. For each CI structure represented fully by a DAG there exists an annotated graph A, which represents this structure and it is unique.

4. Essential graphs and annotated graphs

Another task is to find an algorithm which would provide us with an annotated graph corresponding to a given essential graph introduced in Definition 31. The question is whether we can find an annotated graph A, which would represent the same CI structure as the given essential graph D^*. In other words, we are trying to find an annotated graph A which corresponds to any DAG D from the equivalence class $[D^*]$.

In order to construct the annotated graph which corresponds to a given essential graph we need to recognise the skeleton of the annotated graph, annotated edges and annotations. An annotated graph is built over a DAG D, therefore we need to examine some DAG D as an instance from the class $[D^*]$.

Algorithm 3. ***(from essential graph to annotated graph)***
Input: an essential graph $D^* = (\mathbf{V}, E)$*, which represents an equivalence class* $[D^*]$*;*
Output: an annotated graph $A = (U_A, N_A)$*.*

(i) The undirected graph $U_A = (\mathbf{V}, E')$ *is built over the same set of nodes* $\mathbf{V}$ *as* D^**;*

(ii) If $(u, v) \in E$ *then* $(u, v) \in E'$ *and* $(v, u) \in E'$ *for each pair of nodes* $u, v \in \mathbf{V}$ *(the undirected graph* U_A *will contain all directed and undirected edges from* E*);*

(iii) For each immorality $(p, q, c) \in D^*$ *with unmarried parents* p, q *we add moral edges* (p, q) *and* (q, p) *to* E'*;*

(iv) For each moral edge $p —— q$ *we create its annotation* $\mathbf{S}$*.* $\mathbf{S}$ *contains all common children of nodes* p *and* q *and descendants of those children in essential graph* D^**.*

Theorem 8. *(Correctness of Algorithm 4) The annotated graph* $A = (U_A, N_A)$ *produced as the output of the Algorithm 3 corresponds to the same CI structure* $C(\mathbf{V})$ *as the input essential graph* D^**.*

Proof. Essential graph D^* represents a Markov-equivalence class $[D^*]$. Let us take one DAG D from $[D^*]$ and prove, that annotated graph A could have been constructed from D using Algorithm 1. Then we will also know, that A represents $[D^*]$, due to Theorem 5.

(1) Steps (i), (ii), (iii) give us, that by Theorem 3 and the construction of the essential graph (Definition 31) the essential graph D^* has the same skeleton as any DAG $D \in [D^*]$. Also, it has exactly the same immoralities (the immoralities from a DAG D are preserved, whereas new ones can not occur due to the fact that we can only erase arrows and replace them by undirected edges).

(2) In step (iv) we create the annotation for edges between immoral parents. The question is, whether an annotation created from essential graph D^* equals to an annotation recovered from DAG D. Let us have an immorality (p, q, c), denote the set of all common children of p and q and their descendants in essential graph D^* as $\mathbf{S}^*$. We claim that the set $\mathbf{S}^*$ is equal to the set $\mathbf{S}$, which denotes the range for the unmarried couple (p, q) constructed from the DAG $D \in [D^*]$.
First, let us state, that $\mathbf{S}^* \subset \mathbf{S}$. That is true, because if we can find a directed path from one of the nodes p or q in D^* to some node d through their common child c, then we can find the same directed path in D (D contains all of the directed edges present in D^*).
What about the other inclusion $\mathbf{S} \subset \mathbf{S}^*$? First, let us state, that all common children of p and q are both in $\mathbf{S}$ and $\mathbf{S}^*$. Let us fix one $d \in \mathbf{S}$, who is not a common child of p and q. The fact, that $d \in \mathbf{S}$ implies that there is a directed path from one of the nodes from the domain (p, q) through their common child c to d. Let us take a common child c of the nodes p and q, such that the directed path between c and d is the shortest (contains minimum of edges), denoted by

$$\pi_{D^*} : (c = d_1) \longrightarrow d_2 \longrightarrow d_3 \longrightarrow \ldots \longrightarrow (d_n = d).$$

Let us assume, that $d \notin \mathbf{S}^*$. That means, that there is no directed path from any common child of nodes p and q to node d. Which means, that a path π_D does not exist in essential graph D^*. The path is thus 'broken' somewhere, which means, that one of the directed arrows is not essential and looses its direction in the essential graph D^*. This arrow can not be the one between p and c, because these belong to an immorality and stay in the graph D^*. Let us, therefore, take the smallest $i \in \{1, \ldots, n-1\}$, such that the edge between d_i and d_{i+1} is undirected in the essential graph D^*.

Let us consider the case, when $i = 1$ separately: when $i = 1$, then there is a DAG D' in D^*, such that $d_2 \longrightarrow d_1$. Therefore, there should be an arrow from p to d_2 as well as from q to d_2 (see Figure 4.1), which makes d_2 a common child of p and q, which is a contradiction with the assumption.

Figure 4.1: Illustration of the case, when $i = 1$

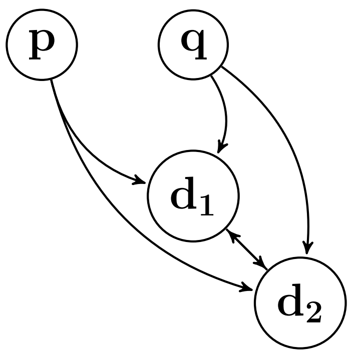


Figure 4.2: Illustration of the situation in the essential graph

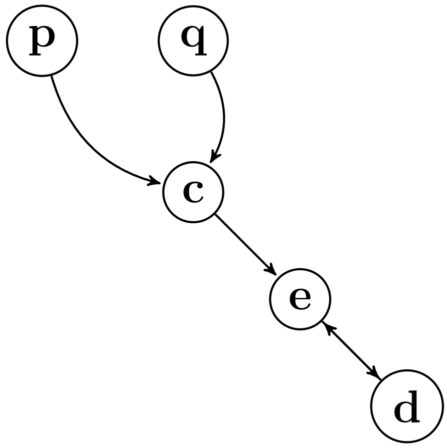
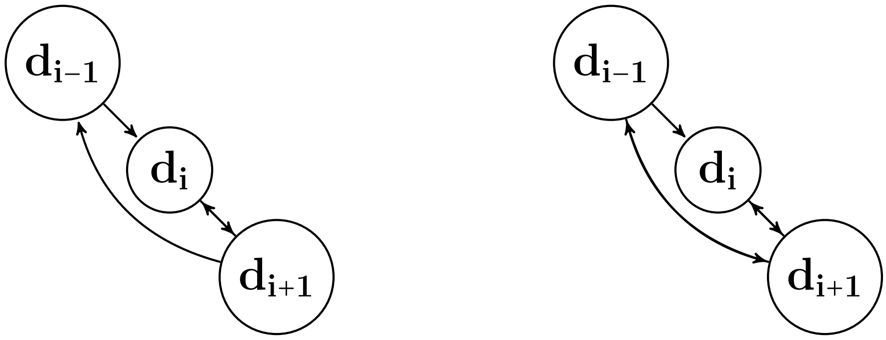


Figure 4.3: Semidericted cycles



Let us examine the case, $i \geq 2$:

Since there are no *flags* in the essential graph (Theorem (4), point (ii)), nodes d_{i-1} and d_{i+1} must be connected by an edge. Since the essential graph D^* is a chain graph (Theorem (4), point (i)) neither of the configurations from Figure 4.3 can be a subgraph of D^*: because these are semidirected cycles. Therefore node d_{i-1} is connected with node d_{i+1} by an arrow: $d_{i-1} \longrightarrow d_{i+1}$. This means, that there exists a shorter directed path between c and d in essential graph D^*:
$\pi'_{D^*} : (c = d_1) \longrightarrow ... \longrightarrow d_{i-1} \longrightarrow d_{i+1} \longrightarrow ... \longrightarrow (d_n = d)$. In particular the same directed path exists in DAG D:
$\pi'_D : c = d_1 \rightarrow ... \rightarrow d_{i-1} \rightarrow d_{i+1} \rightarrow ... \rightarrow d_n = d$. Which is a contradiction to the assumption, that π_{D*} is the shortest directed path from a common child of nodes p and q, c to d.

In both cases we reach a contradiction, which means, that the shortest directed path should be also in the essential graph D^*, which represents a DAG D. Therefore, $\mathbf{S}^* \subset \mathbf{S}$

The skeleton of A is the same, as of A' built from $D \in [D^*]$ using Algorithm 1. The annotated edges as well as the annotation for those edges are equal in A and in A', therefore A is equal to A'. □

The following concepts, which include algorithm formulated by C. Meek will be sed in the following text to formulate an inverse algorithm: from annotated graph to corresponding essential graph. Let us formulate the result of C. Meek, published in the paper Meek [1995], section 2.1.2. In his work, Meek understands the essential graph as a partially directed graph whose edges (called adjacencies in the paper) are the same as any complete casual explanation for $C(\mathbf{V})$. This is in concordance with our observations after Definition 31. A hybrid graph D^{**} used in the following Theorem is called a *pattern*. In Meek [1995] the set of orientation rules for patterns is formulated and Theorem 9 is proven.

Theorem 9. *Let us have a DAG D. Its pattern $D^{*}\ast$, has the same skeleton and the same immoralities as D. It is possible to reconstruct the whole essential graph D^*, such that $[D^*] = [D]$ following a set of rules from Meek [1995].*

Algorithm 4. *(from annotated graph to essential graph)*
Input: an annotated graph $A = (U_A, N_A)$, where $U_A = (\mathbf{V_A}, E_A)$ is a UG;
Output: an essential graph $D^* = (\mathbf{V}, E)$;
The steps are as follows.

(i) The set of nodes is identical in graphs U_A and D^*, in other words $\mathbf{V} = \mathbf{V_A}$;

(ii) The set of edges differs in moral edges*, $E = E_A \setminus \{annotated\ edges\}$;*

(iii) For each element $e = (p, q, \mathbf{S}) \in N_A$ let us find all $c \in \mathbf{S}$ such that $(p, c) \in E$ and $(q, c) \in E$, then direct those edges in the following way: $p \longrightarrow c$, $q \longrightarrow c$.

(iv) Expand the directed edges using the set of rules formulated by Meek.

Theorem 10. *Using the Algorith 4 from an annotated graph A we get an essential graph D^*, which corresponds to the same CI structure $C(\mathbf{V})$.*

Proof. First two steps (i) and (ii) of the Algorithm 4 are quite intuitive: both graphs, A and D^* have to be defined over the same set of nodes, $\mathbf{V}$. Edge set of an essential graph D^* differs only in annotated edges, therefore by removing the annotated edges we obtain the correct set of edges for D^*. The third step (iii) recovers all of the immoralities in the essential graph D^*.

(i) If (p, q, c) is an immorality in the essential graph D^*, then it is an immorality in every DAG D, which belongs to $[D^*]$. Therefore, $p — c$, $q — c$ in A and in N_A, there is an element $(p, q, \mathbf{C}) \in N_A$ with $c \in \mathbf{S}$.

(ii) If $(p, q, \mathbf{S}) \in N_A$, then p, q is an unmarried couple in some DAG D, which belongs to the class $[D^*]$. It also means, that every node from the set $\mathbf{S}$ is either a common child of the nodes p and q in the DAG D or a descendant of some common child from $\mathbf{S}$. Let us take a node $c \in \mathbf{S}$, that satisfies $p — c$ and $q — c$ in A. If c is not a common child of the nodes p and q in D, then c must be a parent of at least one of the nodes p or q. Let us assume without loss of generality, that c is a parent of q in D. But it is also a descendant of a common child of p and q. Therefore there exists a node d, such that $q \longrightarrow d$, there is a directed path π_D from the node d to the node c and also an arrow $c \longrightarrow q$ in D. All of these arrows form a directed cycle in D, which is a contradiction since D is a DAG. Therefore the node c is a common child of the nodes p and q.

The step (iv) is proven in Meek [1995]. □

5. Characteristic imsets

Another essentially different way to represent conditional independence structures induced by Bayesian networks is based on the concept of a *characteristic imset*, which was first introduced in the paperM. Studený [2010]. The *characteristic imset* which represents a CI $C(\mathbf{V})$ is a zero-one vector whose components are indexed by subsets of $\mathbf{V}$ which have at least two elements. Its definition is built on the Directed Acyclic Graph, which represents some CI structure $C(\mathbf{V})$. The following definition is taken over from M. Studený [2014].

Definition 35 (Characteristic imset). *Let $D = (\mathbf{V}, E)$ be an acyclic directed graph. The* characteristic imset *over the set $\mathbf{V}$ for D can be introduced as a zero-one vector $\mathbf{c}_D$ with components $\mathbf{c}_D(\mathbf{S})$, where $\mathbf{S} \subset \mathbf{V}$, $|\,\mathbf{S}\,| \geq 2$, and*

$$\mathbf{c}_D(\mathbf{S}) = 1 \iff \exists i \in \mathbf{S} \text{ such that } \mathbf{S} \setminus \{i\} \subset pa_D(i).$$

Let us consider a characteristic imset $\mathbf{c}_D$. If its component $\mathbf{c}_D(\mathbf{S})$ for $\mathbf{S} \subset \mathbf{V}$ is 1, then there exists a node $i \in \mathbf{S}$, such that every other node in $\mathbf{S}$ is its parent. We call such node *sink node* within $\mathbf{S}$.

Let us mention the most important properties of characteristic imset, which are mentioned in M. Studený [2014], section 2.4.

Theorem 11. *The characteristic imset is uniquely identified by its components which are indexed by sets with two or three elements. Let us denote this essential set by $\mathbf{T} = \{\mathbf{c}_D(\mathbf{S}), |\,\mathbf{S}\,| \leq 3\}$. All of the values for components which are indexed by larger sets can be reconstructed from the set $\mathbf{T}$.*

Theorem 12. *Two DAGs D and $\tilde{D}$ are Markov equivalent if and only if their corresponding characteristic sets are the same: $\mathbf{c}_D = \mathbf{c}_{\tilde{D}}$.*

Let us give an example, illustrating, how a characteristic imset can be obtained from a DAG. In Figure 5.1 we can see a Directed Acyclic Graph, whereas Table 5.1 shows us the values of the corresponding characteristic imset.

Figure 5.1: Example of a Directed Acyclic Graph D used to illustrate the concept of a characteristic imset.

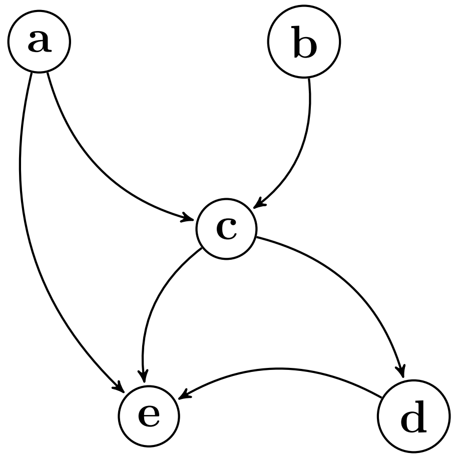


Table 5.1: The characteristic imset $\mathbf{c}_D$, which corresponds to the DAG D from Figure 5.1.

ab	0	**ac**	1	**ad**	0	**ae**	1	**bc**	1	**bd**	0
be	0	**cd**	1	**ce**	1	**de**	1	**abc**	1	**abd**	0
abe	0	**acd**	0	**ace**	1	**ade**	1	**bcd**	0	**bce**	0
bde	0	**cde**	1	**abcd**	0	**abce**	0	**abde**	0	**acde**	1
bcde	0	**abcde**	0								

For example, the component indexed by **ace** is 1 because the set $\{a, c, e\}$ contains a *sink node* e. Both nodes in $\mathbf{S} \setminus e$ are parents of e. On the other hand, the component indexed by the set $\{a, b, e\}$ equals 0, because neither of the nodes a, b or e is a *sink node*: a is a parent of e, but b is not a parent of e. The component indexed by the set **acde** has a value 1, because a, c and d are parents of e. Its value may be also deduced from the values of the components which are indexed by couples or triplets of nodes.

Since the primary goal of this thesis is to study properties of annotated graphs and its connections with other representations of CI structures, in the following text we provide an algorithm, which takes the characteristic imset $\mathbf{c}_D$ and, without reconstructing a corresponding DAG D, gives us the annotated graph A. Whereas A corresponds to the same CI structure $C(\mathbf{V})$ as $\mathbf{c}_D$. The Algorithm 5 is a new original contribution of this thesis.

A reconstruction algorithm for the essential graph from the characteristic imset is provided in M. Studený [2009].

Algorithm 5. *(From a characteristic imset to an annotated graph)*
Input: the characteristic imset $\mathbf{c}_D$*, shortly* $\mathbf{c}$*;*
Output: the annotated graph $A = (U_A, N_A)$ *, where* $U_A = (\mathbf{V}, E)$*.*

(i) begin with the undirected graph $U_A = (\mathbf{V}, E)$*, where* E *is the empty set of edges and* $\mathbf{V}$ *is the set over which* $\mathbf{c}$ *is defined;*

(ii) identify those edges from E *that are not annotated: for each* $p, q \in \mathbf{V}$*,* $p \neq q$*, if* $\mathbf{c}(\{p,q\}) = 1$ *then* $p — q$*, that is* (p,q) *and* (q,p) *are included in* E*;*

(iii) identify the set of annotated edges and the annotations for them: for each non-edge couple of nodes $p, q \in \mathbf{V}$*,* $p \neq q$ *such, that there exists* $c \in \mathbf{V} \setminus \{p,q\}$ *with* $\mathbf{c}(\{p,q,c\}) = 1$*,* $p — q$ *will be an annotated edge;*
The annotation of this edge is constructed by applying the following steps:

(1) *construct the set* $\mathbf{C}$*:*
$\mathbf{C} = \{c \in \mathbf{V} : \mathbf{c}(\{p,q,c\}) = 1\}$*;* $\mathbf{S} = \mathbf{C} \cup \{p,q\}$*;* $\mathbf{M} = \{p,q\}$*;*

(2) *while the set* $\mathbf{C} \setminus \mathbf{M}$ *is not empty, take* $c \in \mathbf{C} \setminus \mathbf{M}$*,*
find all nodes d *in* $\mathbf{V} \setminus \mathbf{S}$ *such that* $\mathbf{c}(\{c,d\}) = 1$ *and either* $\mathbf{c}(\{p,c,d\}) = 0$ *or* $\mathbf{c}(\{q,c,d\}) = 0$*.*
Then include d *in* $\mathbf{S}$*:* $\mathbf{S} = d \cup \mathbf{S}$*.*
Mark c *as a predecessor of* d*:* $pre(d) = c$ *.*
After finding all such d *for the considered* $c \in \mathbf{C} \setminus \mathbf{M}$ *include* c *in* M*:* $\mathbf{M} = c \cup \mathbf{M}$*.*

(3) *the set* $\mathbf{S} \setminus \mathbf{M}$ *is not empty, take a node* $d \in \mathbf{S} \setminus \mathbf{M}$*,*
for each node $e \in \mathbf{V} \setminus \mathbf{S}$*, such that* $\mathbf{c}(\{d,e\}) = 1$ *and* $\mathbf{c}(\{pre(d), d, e\}) = 0$*,*
put $\mathbf{S} = e \cup \mathbf{S}$*,* $pre(e) = d$*. After finding all such* e *for considered* $d \in \mathbb{S} \setminus \mathbf{M}$ *put* $\mathbf{M} = d \cup \mathbf{M}$*.*

(4) *the nodes* p *and* q *from the set* $\mathbf{S}$*:* $\mathbf{S} = \mathbf{S} \setminus \{p,q\}$*,*
add the element $(p,q,\mathbf{S})$ *to* N_A*.*

Remark. Let us take a closer look at what each set in the Algorithm 5 represents. In the step (iii) we fix one annotated edge $p — q$ and reconstruct its annotation from the characteristic imset $\mathbf{c}_D$ and a skeleton of the annotated graph A built in the steps (i) and (ii). The set $\mathbf{S}$ represents the annotation for the edge $p — q$, therefore $\mathbf{S}$ contains all common children and their descendants in every DAG D, which corresponds to the imset $\mathbf{c}_D$. The set $\mathbf{C}$ is a subset of $\mathbf{S}$ and it represents all of the common children of p and q in the DAG D. The set $\mathbf{M}$ represents the parents of the immorality (p, q) and all of the nodes from $\mathbf{S}$, whose children in the DAG D are already in the set $\mathbf{S}$.
The symbol $pre(d)$ stands for the predecessor of the node d on the implied path between the unmarried parents p, q through their common child c to the node d. The value of $pre(d)$ does not change for fixed annotated edge (p,q) throughout step (iii).

Example. Let us see how the algorithm works in case of the characteristic imset $\mathbf{c}_D$ from Table 5.1. First two steps, (i) and (ii), will give us the undirected graph in Figure 5.2:

Let us start with the part (iii). For the triplet (a,b,c) it is true that $\mathbf{c}(\{a,b,c\}) = 1$, $\mathbf{c}(\{a,b\}) = 0$. Therefore, $\{p,q\} = \{a,b\}$ is an annotated edge in the graph A. Now we will focus on the annotation of the edge $\{a,b\}$. The set $\mathbf{C}$ is $\mathbf{C} = \{c\}$, because c is the only node, for which $a — d$, $b — d$ and $\mathbf{c}(\{a,b,d\}) = 1$. After step (1) we have $\mathbf{S} = \mathbf{C} \cup \{a,b\} = \{c,a,b\}$ and $\mathbf{M} = \{a,b\}$.

In the step (2) for $\{p,q\} = \{a,b\}$ we examine neighbours of the node c (except from those connected by an annotated edge) from the set $\mathbf{V} \setminus \mathbf{S}$. These are nodes e and d since $\mathbf{c}(\{c,e\}) = \mathbf{c}(\{c,d\}) = 1$ and $\{e,d\} \notin \mathbf{S}$. Since $\mathbf{c}(\{b,c,e\}) = 0$ and $\mathbf{c}(\{a,c,d\}) = 0$ we include both e and d into $\mathbf{S}$. Therefore, $\mathbf{S} = \{e,d\} \cup \mathbf{S}$ and $pre(e) = pre(d) = c$. We need to modify the set $\mathbf{M}$: $\mathbf{M} = c \cup \mathbf{M}$. Since the set $\mathbf{C} \setminus \mathbf{M}$ is empty, part (2) is finished.

In the step (iii) part (3) we take a node $e \in \mathbf{S} \setminus \mathbf{M}$. Since there are no nodes in the set $\mathbf{V} \setminus \mathbf{S}$, we expand the set $\mathbf{M}$ to $e \cup \mathbf{M}$ and process the node d from $\mathbf{S} \setminus \mathbf{M}$. Situation is similar for d, the set $\mathbf{V} \setminus \mathbf{S}$ is empty, therefore $\mathbf{M} = d \cup \mathbf{M}$. Since $\mathbf{S} \setminus \mathbf{M}$ is empty, the step (iii) part (3) is finished and we can move to the next one.

In the step (4) we just remove the nodes a, b from the annotation $\mathbf{S}$ and add an element $(a,b,\{c,d,e\})$ to N_A. The partial result is in the Figure 5.3.

An analogous procedure (iii) can be performed for another non-edge couple of nodes $\{p,q\} = \{a,d\}$. Since $\mathbf{c}(\{a,e,d\}) = 1$, $a — d$ is an annotated edge in A. The only 'common child' is the node e, therefore $\mathbf{C} = \mathbf{S} = \{a,d,e\}$. $\mathbf{M} = \{a,d\}$. The only node which from $\mathbf{V} \setminus \mathbf{S}$, which is a neighbour of e in the undirected graph from step (ii) is the node c. Since $\mathbf{c}(\{a,e,c\}) = 1$ and $\mathbf{c}(\{d,e,c\}) = 1$, c does not belong to the range $\mathbf{S}$. Therefore, after removing nodes a and d from the range in the step (4), we get $\mathbf{S} = \{e\}$. Since there are no more non - edge couples p, q, where $\mathbf{c}(p,q,c) = 1$ for some $c \in \mathbf{V} \setminus \{p,q\}$, the Algorithm 5 is finished and $N_A = \{(a,b,\{c,d,e\}),(a,d,\{e\})\}$. The resulting annotated graph is shown in Figure 5.4.

Now let us state the main result on Algorithm 5. But first we need to formulate one useful lemma, which is an observation about Algorithm 5.

Lemma 13. *Let us have a characteristic imset c_D built over some DAG D. Then in the part (iii) of the Algorithm 5 the following is true. If $d \in \mathbf{M}$, then d is either one of the unmarried parents p, q or Algorithm 5 has already included all of the children of the node d in the DAG D to the set $\mathbf{S}$.*

Proof. First, let us assume, that d is not p or q.
For each node $d \in \mathbf{V}$, let us consider a number $n(d) \in \mathbb{N}$, which stands for the length of the directed path $\pi^c(d)$ from one of the parents p or q through their common child c to d in D. The directed path $\pi^(d) = (v_i)_{i=1}^n$ is implied by the Algorithm 5 in the following way. $v_n = d$, $v_i = pre(v_{i+1})$ for $i \in 2, ...n-1$ and $v_1 = p$. The path π^d is unique, every node d has at most one predecessor $pre(d)$. Let us proceed with the induction method and prove that for $d \in \mathbf{M}$, $ch_D(d) \subset \mathbf{S}$:

(i) If $n(d) = 1$, then for every node $e \in \mathbf{V} \setminus \mathbf{S}$, such that $e \in ne_D(d)$, $\mathbf{c}_D(\{p,d,e\}) = 1$ and $\mathbf{c}_D(\{q,d,e\}) = 1$. It implies, that d is a sink node for both sets $\{p,d,e\}$ and $\{q,d,e\}$ ($e \notin \mathbf{C}$, therefore it can not be a sink node of the sets), therefore e can not be a child of d.

(ii) If $n(d) > 1$, then for every node $e \in \mathbf{V} \setminus \mathbf{S}$, such that $e \in ne_D(d)$, $\mathbf{c}_D(\{pre(d), d, e\}) = 1$. $n(pre(d)) < n(d)$, $pre(d) \in \mathbf{M}$therefore due to induction hypothesis $ch_D(pre(d)) \subset \mathbf{S}$. If d is a sink node in the set $\{pre(d), d, e\}$, then $e \notin ch_D(d)$. If e is a sink node in the set $\{pre(d), d, e\}$, then $einch_D(d)$ and $e \in ch_D(pre(d)) \subset \mathbf{S}$, which means, that e is already in $\mathbf{S}$.

Therefore, we can conclude, that if $d \in \mathbf{M}$, then $ch_D(d) \subset \mathbf{S}$. □

Theorem 14. *Let D be a DAG over $\mathbf{V}$. Algorithm 5 described in this section transforms the characteristic imset $\mathbf{c_D}$ into an annotated graph $A = (U_A, N_A)$ ($U_A = (\mathbf{V}, E)$). Then both $\mathbf{c_D}$ and A represent the same conditional independence structure over $\mathbf{V}$, namely the one corresponding to D.*

Proof. The characteristic imset $\mathbf{c_D}$ represents the same conditional independence structure as the DAG D from which the characteristic imset was built.
Let us consider an annotated graph $A' = (U'_A, N'_A)$, which is the output of Algorithm 1 used on the input DAG D. We will prove, that $A = A'$.

Let us consider couples of nodes p, q, which we find in the step (iii) of the Algorithm 5. Firstly we prove, that those couples are immoral parents of immoralities from D and that every immoral couple from D is recovered by the Algorithm 5.

(i) If (p, q, c) is an immorality in D with unmarried parents p and q then $\mathbf{c}_D(\{p, q\}) = 0$, because there is no edge in D between p and q. Moreover, $\mathbf{c}_D(\{p, q, c\}) = 1$, because the node c satisfies the following condition: $\{p, q, c\} \setminus \{c\} \subset pa_D(c)$;

(ii) If $\mathbf{c}_D(\{p, q\}) = 0$ and $\mathbf{c}_D(\{p, q, c\}) = 1$, then there is no edge between p and q and c has to be a sink node in the set $\{p, q, c\}$, because it can not be any of the other nodes: p is not be a sink node, because there is no edge between p and q (similarly for q). Therefore $\{p, q\} \subset pa_D(c)$ and (p, q, c) is an immorality with unmarried parents p and q.

Therefore the set of annotated edges in A is equal to the set of annotated edges in A'. This also means, that $U_A = U'_{A'}$, due to the fact, that both U'_A and U_A are the skeletons of the DAG D together with the annotated edges.

Secondly we prove, that in the step (iii) part (1) of the algorithm, the set $\mathbf{C}$ of common children of p and q in D is identified:

(i) If c is a common child of unmarried parents p and q then $\mathbf{c}_D(\{p, q, c\}) = 1$;

(ii) If $\mathbf{c}_D(\{p, q, c\}) = 1$ and p and q are disconnected then $p \longrightarrow c$ and $q \longrightarrow c$, so that the condition of a characteristic imset is satisfied $(\{p, q, c\} \setminus \{c\} \subset pa_D(c))$.

Thirdly we prove that the set of children of common children of p and q in D is identified correctly in the step (iii), (2). In this part we add to the set $\mathbf{S}$ all nodes which are children of nodes from $\mathbf{C}$. We process every node from $\mathbf{C}$ one by one and mark those nodes which are already processed by adding them to the set $\mathbf{M}$ (so that we do not process them twice). For each $c \in \mathbf{M}$ we consider a potential child in the set $\mathbf{V} \setminus \mathbf{S}$, because we are not interested in the nodes, which already belong to the annotation.

(i) If d is a child of $c \in \mathbf{C}$ in D and both $\mathbf{c}_D(p,c,d) = 1$ and $\mathbf{c}_D(p,c,d) = 1$, we show, $p \longrightarrow d \longleftarrow q$ in D. Indeed, if $\mathbf{c}_D(p,c,d) = 1$, since p and q have children in $\{p,c,d\}$ the sink node in $\{p,c,d\}$ must be d. Similar reasoning works for q. Then d is a common child of p and q, which is a contradiction with the assumption, that d was chosen outside $\mathbf{S}$, because $\mathbf{C} \subset \mathbf{S}$. Therefore either $\mathbf{c}_D(\{p,c,d\}) = 0$ or $\mathbf{c}_D(\{p,c,d\}) = 0$.

(ii) If $c — d$ and either $\mathbf{c}_D(\{p,c,d\}) = 0$ or $\mathbf{c}_D(\{p,c,d\}) = 0$ then d is a child of c, because if $d \longrightarrow c$, then $\mathbf{c}_D(\{p,c,d\}) = 1$ and $\mathbf{c}_D(\{q,c,d\}) = 1$ (c would be a sink node in both sets $\{p,c,d\}$ and $\{q,c,d\}$).

Lastly, we identify the set of descendants of children of common children of p, q. In Algorithm 5 in the step (iii) (3) for each $d \in \mathbf{S} \setminus \mathbf{M}$ we consider a set of nodes from $\mathbf{V} \setminus \mathbf{S}$ which are neighbours of d (not by an annotated edge). Let us take one such node e. Since $e \in \mathbf{V} \setminus \mathbf{S}$ there is no directed arrow from $pre(d)$ to e (d was added to $\mathbf{S}$ as a child of $pre(d)$, therefore $pre(d) \in \mathbf{M}$ and Lemma 13 gives us, that $ch_D(pre(d)) \subset \mathbf{S}$). Therefore there are two options:

(i) There is no edge between $pre(d)$ and e. Then $\mathbf{c}_D(\{pre(d),d,e\}) = 0$ if and only if d is not a sink node and $d \longrightarrow e$ (therefore e belongs to the range);

(ii) There is an edge $e \longrightarrow pre(d)$ then again $\mathbf{c}_D(\{pre(d),d,e\}) = 0$ if and only if d is not a sink node and $d \longrightarrow e$ (therefore e belongs to the range);

As a conclusion, let us see, that for each annotated edge $\{p,q\}$ in A, the set of annotation is the set of common children of nodes p and q in D and descendants of those children. Therefore, the annotation of the edge $\{p,q\}$ is equal in A and in A'. Since the set of the annotated edges is also equal in two annotated graphs A and A', $N_A = N'_{A'}$. Therefore $A = A'$ and we have proven, that the output annotated graph of the Algorithm 5 represent the same CI structure as the input imset.

$\square$

Figure 5.2: Annotated graph in process: after steps (i) and (ii)

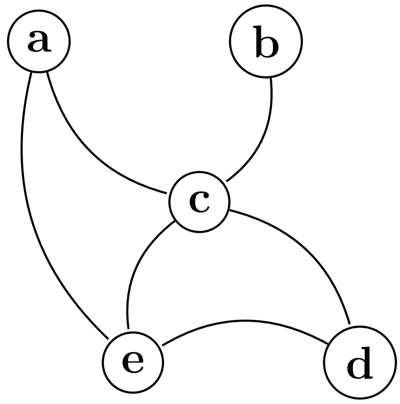


Figure 5.3: Annotated graph in process, (iii), first annotated edge

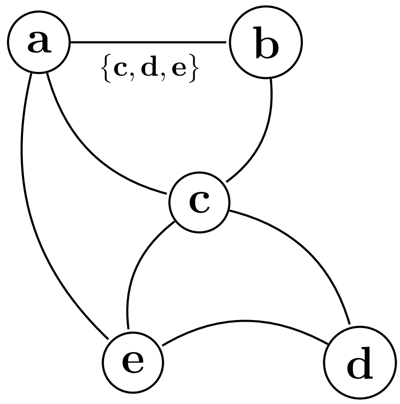
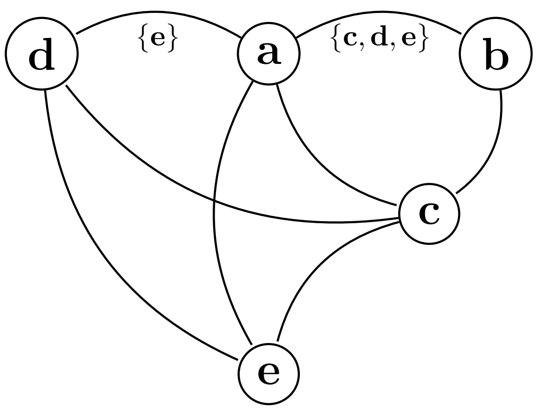


Figure 5.4: The final annotated graph A



Bibliography

J. Anděl. *Matematická statistika.* SNTL - Nakladatelství technické literatury, Praha, 1985.

J. Pearl D. Geiger. Logical and algorithmic properties of conditional independence and graphical models. *The Annals of Statisctics*, 21(4):2001–2021, 1993.

M. Loève. *Probability Theory Foundations - Random Sequences.* D. van Nostrand, Toronto, 1955.

D. Haws M. Studený. Learning bayesian network structure: Towards the essential graph by integer linear programming tools. *International Journal of Approcimate reasoning*, 55:11034–1071, 2014.

J. Vomel M. Studený. A reconstruction algorithm for the essential graph. *International Journal of Approximate Reasoning*, 50:385–413, 2009.

S. Lindner M. Studený, R. Hemmecke. Characteristic imset: a simple algebraic representative of a bayesian network structure. *Proceedings of the th European-Workshop of Probabilistic Graphical Models (PMG)*, pages 257 – 264, 2010.

C. Meek. Causal inference and causal explanation with background knowledge. *Proceedings of the Eleventh Annual Conference on Uncertainty in Artificial Intelligence*, Morgan Kaufmann:403–410, 1995.

A. Paz. Annotated graphs model for representing dag - representable relations. algorithmic approach and extentions. *Proceedings of WUPES*, pages 205 – 220, 2003a.

A. Paz. An alternative version of lauritzen et al's algorithm for checking representation of independencies. *Soft Computing*, 7:344–349, 2003b.

M. D. Perlman S. A. Andersson, D. Madigan. A characterization of markov equivalence classes for acyclic digraphs. *The Annals of Statisctics*, 25(2):505–541, 1977.

M. Studený. *Struktury podmíněné nezávislosti.* Matfyzpress, 2014.

List of Figures

List of Tables