HPDGClassifier

An R package for supervised classification using hybrid probabilistic decision graphs

Antonio Fernández[1] José del Sagrado[2]
Rafael Rumí[3] Antonio Salmerón[3]

[1]Banco de Crédito Social Cooperativo (BCC)
[2]Department of Informatics (University of Almería, Spain)
[3]Department of Mathematics (University of Almería, Spain)

PGMs4SDA 1st Meeting in Consortium
Granada, 4-5 February 2016

Contents

Motivation

Using MoTBF-PDGs for supervised classification

- Take advantage of context specific independencies.
- Inference is carried out directly over the PDG structure in a time linear in the size.
- No restrictions on the model structure.
- No normality assumption.

Motivation

Using MoTBF-PDGs for supervised classification

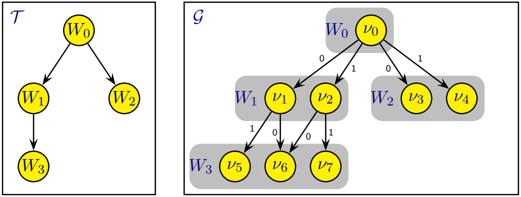
- Take advantage of context specific independencies.
- Inference is carried out directly over the PDG structure in a time linear in the size.
- No restrictions on the model structure.
- No normality assumption.

Why an R package?

- Make it available to the data science community.
- Take advantage of the R framework.
- Facilitate experimentation and comparison.

Contents

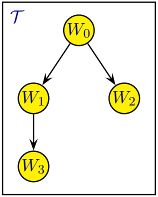
The PDG model

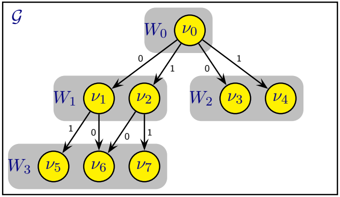


PDG structure

- Variables are organized in a tree structure $\mathcal{T}$.
- Each variable is represented by a set of nodes in $\mathcal{G}$.
- Every node ν_j belongs to one unique variable W_i.
- Each node has an outgoing arc for every state of its variable.

The PDG model





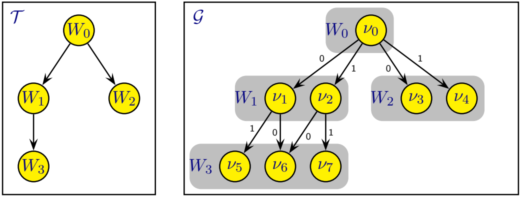
Operation reach

Let $\mathbf{w} = \{W_0 = 0, W_1 = 1, W_2 = 1, W_3 = 1\}$. Then:

$\texttt{reach}(W_0, \mathbf{w}) = \nu_0$; $\texttt{reach}(W_1, \mathbf{w}) = \nu_1$

$\texttt{reach}(W_2, \mathbf{w}) = \nu_3$; $\texttt{reach}(W_3, \mathbf{w}) = \nu_5$

The PDG model



Operation `reach`

Let $\mathbf{w} = \{W_0 = 0, W_1 = 1, W_2 = 1, W_3 = 1\}$. Then:

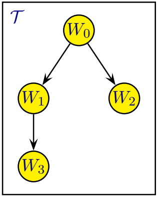
$\texttt{reach}(W_0, \mathbf{w}) = \nu_0$; $\texttt{reach}(W_1, \mathbf{w}) = \nu_1$

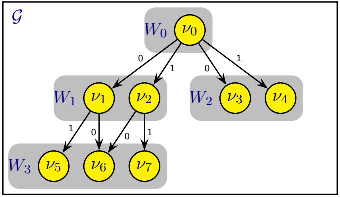
$\texttt{reach}(W_2, \mathbf{w}) = \nu_3$; $\texttt{reach}(W_3, \mathbf{w}) = \nu_5$

Factorisation

$$f(\mathbf{w}) := \prod_{W_i \in \mathbf{W}} f^{\texttt{reach}(W_i, \mathbf{w})}(\mathbf{w}[W_i])$$

The PDG model

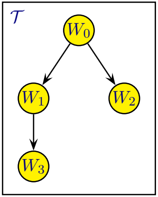


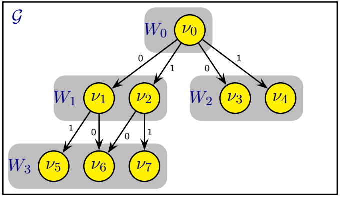


- Every ν_j contains a local distribution f^{ν_j}:

$$
\begin{array}{ll}
f^{\nu_0} = P(W_0) & f^{\nu_4} = P(W_2|W_0 = 1) \\
f^{\nu_1} = P(W_1|W_0 = 0) & f^{\nu_5} = P(W_3|W_0 = 0, W_1 = 1) \\
f^{\nu_2} = P(W_1|W_0 = 1) & f^{\nu_6} = P(W_3|W_1 = 0, \{W_0 = 0 \vee W_0 = 1\}) \\
f^{\nu_3} = P(W_2|W_0 = 0) & f^{\nu_7} = P(W_3|W_0 = 1, W_1 = 1)
\end{array}
$$

The PDG model





- Every ν_j contains a local distribution f^{ν_j}:

$$
\begin{array}{ll}
f^{\nu_0} = P(W_0) & f^{\nu_4} = P(W_2|W_0 = 1) \\
f^{\nu_1} = P(W_1|W_0 = 0) & f^{\nu_5} = P(W_3|W_0 = 0, W_1 = 1) \\
f^{\nu_2} = P(W_1|W_0 = 1) & \boxed{f^{\nu_6} = P(W_3|W_1 = 0, \{W_0 = 0 \vee W_0 = 1\})} \\
f^{\nu_3} = P(W_2|W_0 = 0) & f^{\nu_7} = P(W_3|W_0 = 1, W_1 = 1)
\end{array}
$$

Context-specific independencies:

$$I(W_3, W_0 \mid W_1 = 0) \ , \ \neg I(W_3, W_0 \mid W_1 = 1)$$

Contents

MoTBF model

Univariate MoTBF potential

$$g_k(x) = \sum_{i=0}^{k} a_i \, \psi_i\,(x)\,, \quad a_i \in \mathbb{R}.$$

MoP example

$\Psi = \{1, x, x^2, x^3\} : g(x) = 0.29 - 0.58x + 1.17x^2 + 0.44x^3$

MTE example

$\Psi = \{1, e^{-x}, e^{x}, e^{-2x}, e^{2x}\} : g(x) = 0.14 + 0.29e^{-x} + 0.59e^{x} - 0.22e^{-2x} + 0.08e^{2x}$

MoTBF model

Univariate MoTBF potential

$$g_k(x) = \sum_{i=0}^{k} a_i \, \psi_i\,(x)\,, \quad a_i \in \mathbb{R}.$$

MoP example

$\Psi = \{1, x, x^2, x^3\} : g(x) = 0.29 - 0.58x + 1.17x^2 + 0.44x^3$

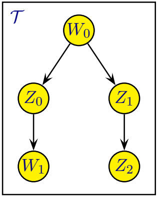
MTE example

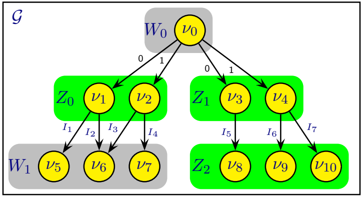
$\Psi = \{1, e^{-x}, e^{x}, e^{-2x}, e^{2x}\} : g(x) = 0.14 + 0.29e^{-x} + 0.59e^{x} - 0.22e^{-2x} + 0.08e^{2x}$

Conditional MoTBF potential

$$g_k^{(j)}(x \mid \mathbf{z} \in \Omega_{\mathbf{Z}}^{j}) = \sum_{i=0}^{k} a_i^{(j)} \, \psi_i^{(j)}(x).$$

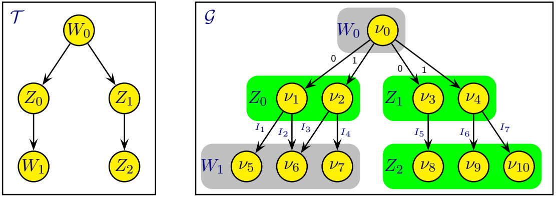
MoTBF-PDG model





- No restrictions on the structure.
- A node ν representing $Z \in \mathbf{Z}$:
  - can have one or more outgoing edges for each Z_i child of Z in $\mathcal{T}$.
  - each edge represents an interval of Ω_Z.
- Normality assumption is not required.

MoTBF-PDG model



I_1	I_2	I_3	I_4	I_5	I_6	I_7
$[0, 0.5)$	$[0.5, 1]$	$[0.5, 1]$	$[0, 0.5)$	$[0, 1]$	$[0, 0.3)$	$[0.3, 1]$

$f^{\nu_0} = P(W_0)$	$f^{\nu_5} = P(W_1 \vert W_0 = 0, Z_0 \in [0, 0.5))$
$f^{\nu_1} = \rho(Z_0 \vert W_0 = 0)$	$f^{\nu_6} = P(W_1 \vert Z_0 \in [0.5, 1])$
$f^{\nu_2} = \rho(Z_0 \vert W_0 = 1)$	$f^{\nu_7} = P(W_1 \vert W_0 = 1, Z_0 \in [0, 0.5))$
$f^{\nu_3} = \rho(Z_1 \vert W_0 = 0)$	$f^{\nu_8} = \rho(Z_2 \vert W_0 = 0)$
$f^{\nu_4} = \rho(Z_1 \vert W_0 = 1)$	$f^{\nu_9} = \rho(Z_2 \vert W_0 = 1, Z_1 \in [0, 0.3))$
	$f^{\nu_{10}} = \rho(Z_2 \vert W_0 = 1, Z_1 \in [0.3, 1])$

MoTBF-PDG classifiers

PDG classifier structure

- Contains a single tree over the variables $\mathbf{C} = \{C\} \cup \mathbf{X}$
- Variable C must be the root of the tree.

MoTBF-PDG classifiers

PDG classifier structure

- Contains a single tree over the variables $\mathbf{C} = \{C\} \cup \mathbf{X}$
- Variable C must be the root of the tree.

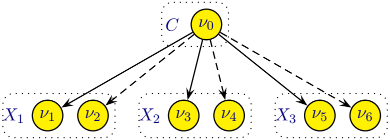
Classification

Given an evidence $\mathbf{x} = \{x_1, \ldots, x_n\}$ the goal is to find c^* such that:

$$\begin{aligned} c^* &= \arg\max_{c \in R(C)} f(c \mid \mathbf{x}) = \arg\max_{c \in R(C)} \frac{f(c, \mathbf{x})}{\sum_{c \in R(C)} f(c, \mathbf{x})} \\ &\propto \arg\max_{c \in R(C)} f(c, \mathbf{x}) \\ &= \arg\max_{c \in R(C)} f(c) \times f(x_1 \mid c) \times \ldots \times f(x_n \mid c, x_1, \ldots, x_{n-1}) \end{aligned}$$

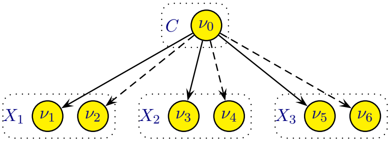
Evaluate the conditional functions in the nodes reached by $\mathbf{x}$ for each $c \in R(C)$.

Learning MoTBF-PDGs

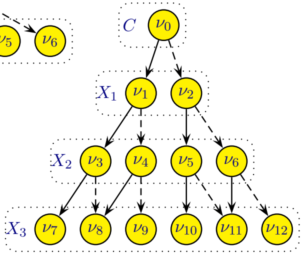


naïve Bayes

Learning MoTBF-PDGs

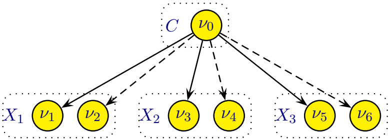


naïve Bayes

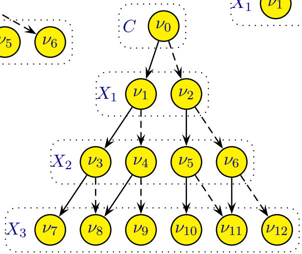


ranked

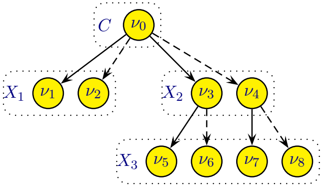
Learning MoTBF-PDGs



naïve Bayes



ranked

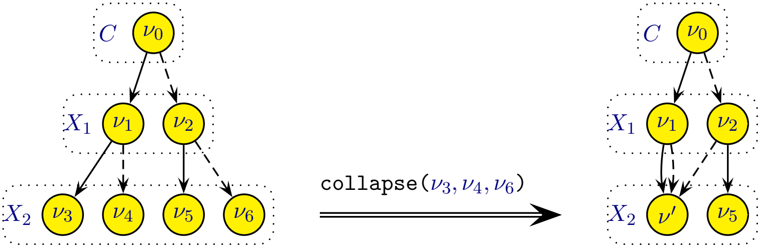


rankedCR

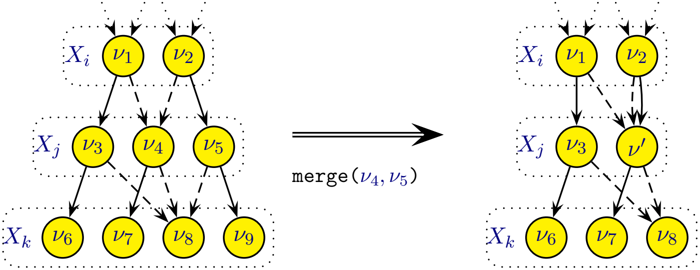
Other methods: CR, Chou-Liu

Collapsing nodes

- PDG is expanded during learning: reached data goes down.
- Collapse nodes to avoid learning from tiny samples.
- Applied when a new variable is included.



Merging nodes



- Merge operation is checked bottom-up once the model is learnt.
- Reduce overfitting and the model size.

Contents

The HPDGClassifier `R` package

Implementation

- Object oriented design.
- Uses packages:
  - `polynom`
  - `quadprog`
  - `foreach`
  - `infotheo`
  - `bnlearn`
  - `methods`
  - `codetools`

The HPDGClassifier R package

Implementation

- Object oriented design.
- Uses packages:
  - `polynom`
  - `quadprog`
  - `foreach`
  - `infotheo`
  - `bnlearn`
  - `methods`
  - `codetools`

Features

- Standard classification setting
- Parallelisation