Interactive Learning of Bayesian Networks

Andrés Masegosa, Serafín Moral

Department of Computer Science and Artificial Intelligence - University of Granada
andrew@decsai.ugr.es

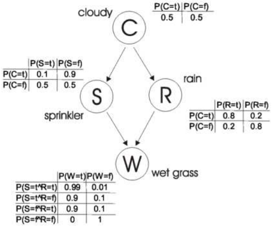
Utrecht, July 2012

Outline

1. Motivation
2. The Bayesian Framework
3. Interactive integration of expert knowledge for model selection
4. Applications:
   - Learning the parent set of a variable in BN.
   - Learning Markov Boundaries.
   - Learning complete Bayesian Networks.
5. Conclusions

Outline

Bayesian Networks



Bayesian Networks

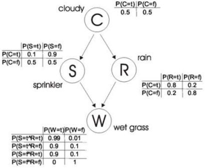
- Excellent models to **graphically represent the dependency structure (Markov Condition and d-separation)** of the underlying distribution in multivariate domain problems: **very relevant source of knowledge**.
- **Inference tasks**: compute marginal, evidence propagation, abductive-inductive reasoning, etc.

Learning Bayesian Networks from Data

Attributes

Objects

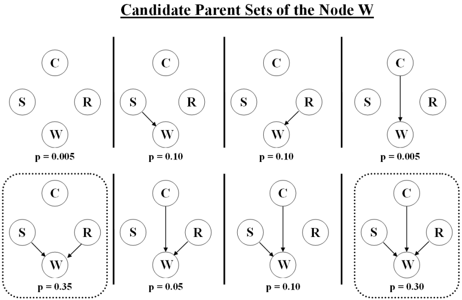
Tid	C	S	R	W
1	Yes	Single	125K	No
2	No	Married	100K	No
3	No	Single	70K	No
4	Yes	Married	120K	No
5	No	Divorced	95K	Yes
6	No	Married	60K	No
7	Yes	Divorced	220K	No
8	No	Single	85K	Yes
9	No	Married	75K	No
10	No	Single	90K	Yes



Learning Algorithms

- **Learning Bayesian networks from data** is quite challenging: the DAG space is super-exponential.
- **There usually are several models with explain the data similarly well**.
  - **Bayesian framework:** high posterior probability given the data.

Learning Bayesian Networks from Data



Uncertainty in Model Selection

- This situation is **specially common in problem domains with high number of variables and low sample sizes**.

Integration of Domain/Expert Knowledge

Integration of Domain/Expert Knowledge

- Find the best statistical model by the **combination of data and expert/domain knowledge**.

Integration of Domain/Expert Knowledge

Integration of Domain/Expert Knowledge

- Find the best statistical model by the **combination of data and expert/domain knowledge**.

- Emerging approach in **gene expression data mining**:
  - There is a growing number of biological knowledge data bases.
  - The model space is usually huge and the number of samples is low (costly to collect).
  - Many approaches have shifted from begin pure data-oriented to try to include domain knowledge.

Integration of Expert Knowledge

Previous Works

- There have been many attempts to introduce expert knowledge when learning BNs from data.
- **Via Prior Distribution:** Use of specific **prior distributions over the possible graph structures** to integrate expert knowledge:
  - Expert assigns **higher prior probabilities to most likely edges**.

Integration of Expert Knowledge

Previous Works

- There have been many attempts to introduce expert knowledge when learning BNs from data.
- **Via Prior Distribution:** Use of specific **prior distributions over the possible graph structures** to integrate expert knowledge:
  - Expert assigns **higher prior probabilities to most likely edges**.
- **Via structural Restrictions:** Expert **codify his/her knowledge** as structural restrictions.
  - Expert defines the existence/absence of arcs and/or edges and causal ordering restrictions.
  - Retrieved model should satisfy these restrictions.

Integration of Domain/Expert Knowledge

- **Limitations of "Prior" Expert Knowledge:**
  - We are forced to **include expert/domain knowledge for each of the elements of the models** (e.g. for every possible edge of a BN).

Integration of Domain/Expert Knowledge

- **Limitations of "Prior" Expert Knowledge:**
  - We are forced to **include expert/domain knowledge for each of the elements of the models** (e.g. for every possible edge of a BN).
  - Expert could **be biased to provided the “clearest” knowledge,** which could be the easiest to be find in the data.

Integration of Domain/Expert Knowledge

- **Limitations of "Prior" Expert Knowledge:**
  - We are forced to **include expert/domain knowledge for each of the elements of the models** (e.g. for every possible edge of a BN).
  - Expert could **be biased to provided the “clearest” knowledge,** which could be the easiest to be find in the data.
  - The **system does not help to the user** to introduce information about the BN structure.

Integration of Domain/Expert Knowledge

Interactive Integration of Expert Knowledge

- Data is firstly analyzed.
- The system only inquires to the expert about **most uncertain elements** considering the information present in the data.

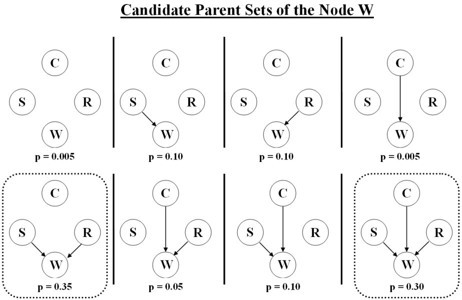
Integration of Domain/Expert Knowledge

Interactive Integration of Expert Knowledge

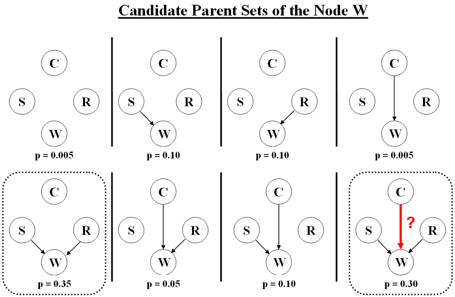
- Data is firstly analyzed.
- The system only inquires to the expert about **most uncertain elements** considering the information present in the data.

- **Benefits**:
  - Expert is only asked **a smaller number of times**.
  - We **explicitly show to the expert** which are the elements about which data do not provide enough evidence to make a reliable model selection.

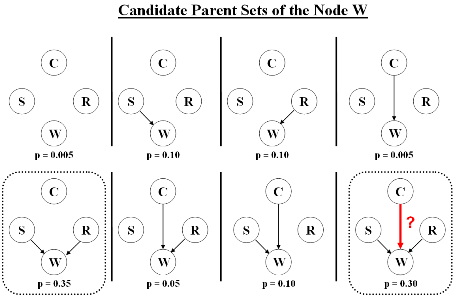
Integration of Domain/Expert Knowledge



Integration of Domain/Expert Knowledge



Integration of Domain/Expert Knowledge



Active Interaction with the Expert

- **Strategy:** Ask to the expert by the presence of the **edges that most reduce the model uncertainty**.
- **Method:** Framework to allow an efficient and effective interaction with the expert.
  - **Expert is only asked for this controversial structural features**.

Outline

1 Motivation

2 The Bayesian Framework

Notation

- Let us denote by $\mathbf{X} = (X_1, ..., X_n)$ a vector of random variables and by D a fully observed set of instances $\mathbf{x} = (x_1, ..., x_n) \in Val(\mathbf{X})$.

Notation

- Let us denote by $\mathbf{X} = (X_1, ..., X_n)$ a vector of random variables and by D a fully observed set of instances $\mathbf{x} = (x_1, ..., x_n) \in Val(\mathbf{X})$.
- Let be **M** a model and $\mathcal{M}$ the set of all possible models. **M** may define:
  - **Joint probability distribution:** $P(\mathbf{X}|\mathbf{M})$ in the case of a Bayesian network.
  - **Conditional probability distribution** for target variable: $P(T|\mathbf{X}, \mathbf{M})$ in the case of a Markov Blanket.

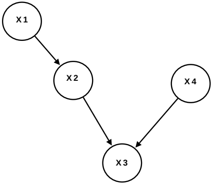
Examples of Models

- Each model **M** is structured:
  - It is defined by a vector of elements $\mathbf{M} = (m_1, ..., m_K)$ where K is the number of possible components of **M**.

Examples of Models

- Each model **M** is structured:
  - It is defined by a vector of elements $\mathbf{M} = (m_1, ..., m_K)$ where K is the number of possible components of **M**.
- **Examples**:

Bayesian Network



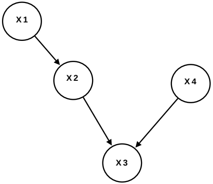
M = (1, 0, 0, 1, 0, -1)

Examples of Models

- Each model **M** is structured:
  - It is defined by a vector of elements $\mathbf{M} = (m_1, ..., m_K)$ where K is the number of possible components of **M**.

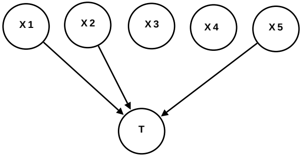
- **Examples**:

Bayesian Network



M = (1, 0, 0, 1, 0, -1)

Markov Blankets



M = (1, 1, 0, 0, 1)

The Model Selection Problem: Bayesian Framework

- Define a **prior probability over the space of alternative models** $P(\mathcal{M})$.
  - It is not the classic uniform prior (the multplicity problem).

The Model Selection Problem: Bayesian Framework

- Define a **prior probability over the space of alternative models** $P(\mathcal{M})$.
  - It is not the classic uniform prior (the multplicity problem).
- For each model, it is computed its **Bayesian score**:

$$score(\mathbf{M}|D) = P(D|\mathbf{M})P(\mathbf{M})$$

where $P(D|\mathbf{M}) = \int_\theta P(D|\theta, \mathbf{M})P(\theta|M)$ is the marginal likelihood of the model.

Benefits of the full Bayesian approach

- We assume that we are **able to approximate the posterior** by means of any **Monte Carlo method**:

$$P(\mathbf{M}|D) = \frac{P(D|\mathbf{M})P(\mathbf{M})}{\sum_{\mathbf{M}' \in Val(\mathcal{M})} P(D|\mathbf{M}')P(\mathbf{M}')}$$

Benefits of the full Bayesian approach

- We assume that we are **able to approximate the posterior** by means of any **Monte Carlo method**:

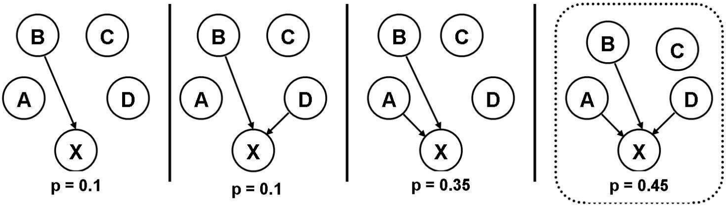
$$P(\mathbf{M}|D) = \frac{P(D|\mathbf{M})P(\mathbf{M})}{\sum_{\mathbf{M}' \in Val(\mathcal{M})} P(D|\mathbf{M}')P(\mathbf{M}')}$$

- We can compute the **posterior probability of any of the elements of a model**:

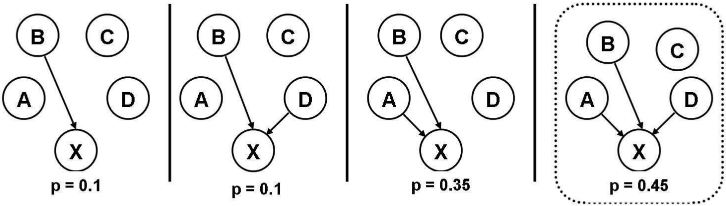
$$P(m_i|D) = \sum_{\mathbf{M}} P(\mathbf{M}|D) I_{\mathbf{M}}(m_i)$$

where $I_{\mathbf{M}}(m_i)$ is the indicator function: 1 if m_i is present in **M** and 0 otherwise.

Examples

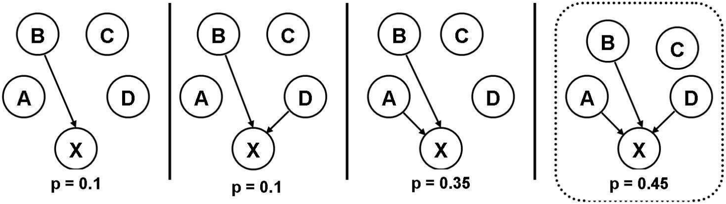


Examples



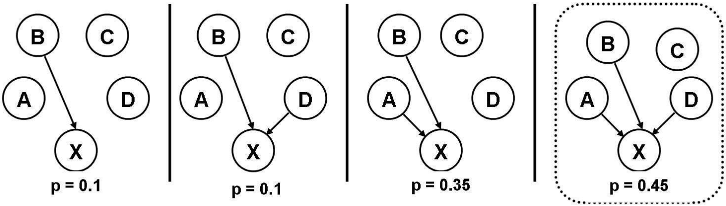
- **High Probable Edges:** $P(B \rightarrow X|D) = 1.0$ and $P(A \rightarrow X|D) = 0.8$

Examples



- **High Probable Edges:** $P(B \rightarrow X|D) = 1.0$ and $P(A \rightarrow X|D) = 0.8$
- **Low Probable Edges:** $P(C \rightarrow X|D) = 0.0$

Examples



- **High Probable Edges:** $P(B \to X|D) = 1.0$ and $P(A \to X|D) = 0.8$
- **Low Probable Edges:** $P(C \to X|D) = 0.0$
- **Uncertain Edges:** $P(D \to X|D) = 0.55$

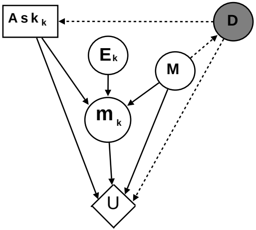
Outline

1. Motivation
2. The Bayesian Framework

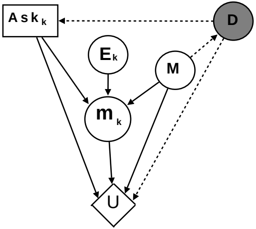
Interaction with of Expert/Domain Knowledge

- **Interactive Integration of of Domain/Expert Knowledge**:
  - **Expert/Domain Knowledge** is given for particular elements m_k of the the models $\mathbf{M}$:
    - If a variable is present or not in the final variable selection.
    - If there is an edge between any two variables in a BN.
  - Expert/Domain Knowledge may be **not fully reliable**.

Our Approach for Expert Interaction

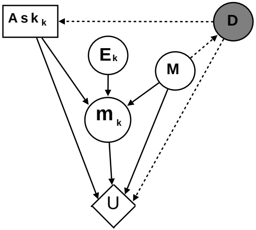


Our Approach for Expert Interaction



- E represents **expert reliability**: $\Omega(E) = \{Right, Wrong\}$.
    - If expert is wrong we assume a random answer.

Our Approach for Expert Interaction



- E represents **expert reliability**: $\Omega(E) = \{Right, Wrong\}$.
  - If expert is wrong we assume a random answer.
- Our goal is to infer model structure:

$$U(Ask_k, m_k, M, D) = logP(M|m_k, Ask_k, D)$$

Our Approach for Expert Interaction

Our Approach for Expert Interaction

- The **expected utility of asking and not-asking** is:

$$V(Ask_k) = \sum_{m_k} \sum_{M} P(M|D)P(m_k|M, Ask_k, D) log P(M|m_k, Ask_k, D)$$

Our Approach for Expert Interaction

- The **expected utility of asking and not-asking** is:

$$V(Ask_k) = \sum_{m_k} \sum_{M} P(M|D)P(m_k|M, Ask_k, D) logP(M|m_k, Ask_k, D)$$

$$V(\overline{Ask}_k) = \sum_{M} P(M|D) logP(M|D)$$

Our Approach for Expert Interaction

- The **expected utility of asking and not-asking** is:

$$V(Ask_k) = \sum_{m_k} \sum_{M} P(M|D)P(m_k|M, Ask_k, D) log P(M|m_k, Ask_k, D)$$

$$V(\overline{Ask}_k) = \sum_{M} P(M|D) log P(M|D)$$

- The difference between both actions, $V(Ask_k) - V(\overline{Ask}_k)$, is the **information gain function**:

$$IG(M : m_k|D) = H(M|D) - \sum_{m_k} P(m_k|D)H(M|m_k, D)$$

Our Approach for Expert Interaction

- The **expected utility of asking and not-asking** is:

$$V(Ask_k) = \sum_{m_k} \sum_{M} P(M|D)P(m_k|M, Ask_k, D) logP(M|m_k, Ask_k, D)$$

$$V(\overline{Ask}_k) = \sum_{M} P(M|D) logP(M|D)$$

- The difference between both actions, $V(Ask_k) - V(\overline{Ask}_k)$, is the **information gain function**:

$$IG(M : m_k|D) = H(M|D) - \sum_{m_k} P(m_k|D)H(M|m_k, D)$$

- It can be shown that the element with the highest information gain is **the one with highest entropy**:

$$m_k^\star = \max_k IG(M : m_k|D) = \max_k H(m_k|D) - H(E_k)$$

Our Approach for Expert Interaction

1. **Approximate** $P(\mathcal{M}|D)$ by means of a MC technique.

Our Approach for Expert Interaction

1. **Approximate** $P(\mathcal{M}|D)$ by means of a MC technique.
2. **Ask the expert** about the element m_k with **the highest entropy**.
    - $\mathbf{a} = \mathbf{a} \cup a(m_k)$.

Our Approach for Expert Interaction

1. **Approximate** $P(\mathcal{M}|D)$ by means of a MC technique.
2. **Ask the expert** about the element m_k with **the highest entropy**.
   - $\mathbf{a} = \mathbf{a} \cup a(m_k)$.
   - Update $P(\mathcal{M}|D, \mathbf{a})$, which is equivalent to:

$$P(\mathcal{M}|D, \mathbf{a}) \propto P(D|\mathcal{M})P(\mathcal{M}|\mathbf{a})$$

Our Approach for Expert Interaction

1. **Approximate** $P(\mathcal{M}|D)$ by means of a MC technique.

2. **Ask the expert** about the element m_k with **the highest entropy**.
   - $\mathbf{a} = \mathbf{a} \cup a(m_k)$.
   - Update $P(\mathcal{M}|D, \mathbf{a})$, which is equivalent to:

$$P(\mathcal{M}|D, \mathbf{a}) \propto P(D|\mathcal{M})P(\mathcal{M}|\mathbf{a})$$

3. **Stop Condition:** $H(m_k|D, \mathbf{a}) < \lambda$.

Our Approach for Expert Interaction

1. **Approximate** $P(\mathcal{M}|D)$ by means of a MC technique.
2. **Ask the expert** about the element m_k with **the highest entropy**.
   - $\mathbf{a} = \mathbf{a} \cup a(m_k)$.
   - Update $P(\mathcal{M}|D, \mathbf{a})$, which is equivalent to:

$$P(\mathcal{M}|D, \mathbf{a}) \propto P(D|\mathcal{M})P(\mathcal{M}|\mathbf{a})$$

3. **Stop Condition:** $H(m_k|D, \mathbf{a}) < \lambda$.
4. **Otherwise:**
   - **Option 1:** Go to Step 2 and ask to the expert again.

Our Approach for Expert Interaction

1. **Approximate** $P(\mathcal{M}|D)$ by means of a MC technique.

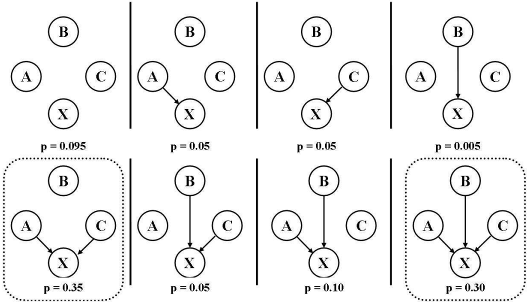
2. **Ask the expert** about the element m_k with **the highest entropy**.
   - $\mathbf{a} = \mathbf{a} \cup a(m_k)$.
   - Update $P(\mathcal{M}|D,\mathbf{a})$, which is equivalent to:

$$P(\mathcal{M}|D,\mathbf{a}) \propto P(D|\mathcal{M})P(\mathcal{M}|\mathbf{a})$$

3. **Stop Condition:** $H(m_k|D,\mathbf{a}) < \lambda$.

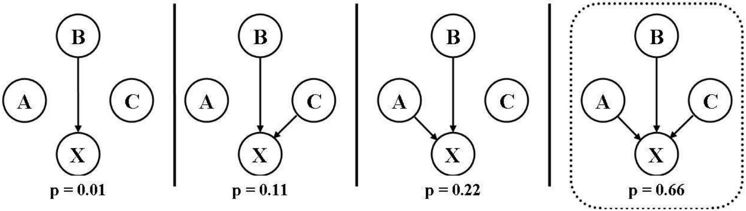
4. **Otherwise:**
   - **Option 1:** Go to Step 2 and ask to the expert again.
   - **Option 2:** Go to Step 1 and sample now using $P(\mathcal{M}|\mathbf{a})$.

Example I



- **Probability of the edges**:
  - $P(A \rightarrow X|D) = 0.8$
  - $P(B \rightarrow X|D) = 0.455$
  - $P(C \rightarrow X|D) = 0.75$

Example II



- **Expert say that $B \rightarrow X$ is present in the model**:
  - $P(A \rightarrow X|D) = 0.88$
  - $P(B \rightarrow X|D) = 1.0$
  - $P(C \rightarrow X|D) = 0.77$

Developments

- This methodology has been applied to the following **model selection problems**:

Developments

- This methodology has been applied to the following **model selection problems**:
  - Induce a **Bayesian network conditioned to a previously given causal order** of the variables.

Developments

- This methodology has been applied to the following **model selection problems**:
  - Induce a **Bayesian network conditioned to a previously given causal order** of the variables.
  - Induce the **Markov Blanket** of a target variable (Feature Selection).

Developments

- This methodology has been applied to the following **model selection problems**:
  - Induce a **Bayesian network conditioned to a previously given causal order** of the variables.
  - Induce the **Markov Blanket** of a target variable (Feature Selection).
  - Induce **Bayesian Networks** without any restriction.

Conclusions

- We have developed a general method for model selection which allow the inclusion of expert knowledge.
- The method is robust even when expert knowledge is wrong.
- The number of interactions is minimized.
- It has been successfully applied to different model selection problems.

Future Works

- Develop a new score to measure the impact of the interaction in model selection.
- Extend this methodology to other probabilistic graphical models.
- Evaluate the impact of the prior over the parameters.
  - Preference among models may change with the parameter prior.
  - Detect the problem and let the user choose.
- Employ alternative ways to introduce expert knowledge:
  - E.g. In BN, we ask about edges: direct causal probabilistic relationships.
  - Many domain knowledge is about correlations and conditional independencies.