Conditional independence and Causality

INTELLIGENT SYSTEMS FOR PATTERN RECOGNITION (ISPR)

DAVIDE BACCIU – DIPARTIMENTO DI INFORMATICA - UNIVERSITA' DI PISA

DAVIDE.BACCIU@DI.UNIPI.IT
On the Nature of Relationships in Bayesian and Markov Networks

Directed edges representing asymmetric cause-effect relationships

Can we reason on the structure of the graph to infer direct/indirect relationships between RVs?
Bayesian Network

- Directed Acyclic Graph (DAG) $\mathcal{G} = (\mathcal{V}, \mathcal{E})$
- Nodes $v \in \mathcal{V}$ represent random variables
  - Shaded $\Rightarrow$ observed
  - Empty $\Rightarrow$ un-observed
- Edges $e \in \mathcal{E}$ describe the conditional independence relationships

**Conditional Probability Tables (CPT)** local to each node describe the probability distribution given its parents

$$P(Y_1, \ldots, Y_N) = \prod_{i=1}^{N} P(Y_i | pa(Y_i))$$
Local Markov Property

Definition (Local Markov property)
Each node / random variable is conditionally independent of all its non-descendants given a joint state of its parents

\[ Y_v \perp Y_{V \setminus \text{ch}(v)} | Y_{pa(v)} \text{ for all } v \in V \]

*Party* and *Study* are marginally independent
- *Party* \( \perp \) *Study*

However, local Markov property does not support
- *Party* \( \perp \) *Study* | *Headache*
- *Tabs* \( \perp \) *Party*

But *Party* and *Tabs* are independent given *Headache*
Markov Blanket

- The Markov Blanket $Mb(A)$ of a node $A$ is the minimal set of vertices that shield the node from the rest of Bayesian Network.
- The behavior of a node can be completely determined and predicted from the knowledge of its Markov blanket:
  \[ P(A|M_b(A), Z) = P(A|M_b(A)) \quad \forall Z \notin M_b(A) \]
- The Markov blanket of $A$ contains:
  - Its parents $pa(A)$
  - Its children $ch(A)$
  - Its children’s parents $pa(ch(A))$
Joint Probability Factorization

An application of **Chain rule** and **Local Markov Property**

1. Pick a topological ordering of nodes
2. Apply chain rule following the order
3. Use the conditional independence assumptions

\[
P(P_A, S, H, T, C) = \\
P(P_A) \cdot P(S|P_A) \cdot P(H|S, P_A) \cdot P(T|H, S, P_A) \cdot P(C|T, H, S, P_A) \\
= P(P_A) \cdot P(S) \cdot P(H|S, P_A) \cdot P(T|H) \cdot P(C|H)
\]
(Ancestral) Sampling of a BN

A BN describes a generative process for observations

1. Pick a topological ordering of nodes
2. Generate data by sampling from the local conditional probabilities following this order

Generate i-th sample for each variable PA, S, H, T, C

1. $p_{ai} \sim P(PA)$
2. $s_i \sim P(S)$
3. $h_i \sim P(H|S = s_i, PA = p_{ai})$
4. $t_i \sim P(T|H = h_i)$
5. $c_i \sim P(C|H = h_i)$
There exist 3 fundamental substructures that determine the conditional independence relationships in a Bayesian network:

- **Tail to tail** (Common Cause)
- **Head to tail** (Causal Effect)
- **Head to head** (Common Effect)
Tail to Tail Connections

○ Corresponds to
\[ P(Y_1, Y_3 | Y_2)P(Y_2) = P(Y_1 | Y_2)P(Y_3 | Y_2)P(Y_2) \]

○ If \( Y_2 \) is unobserved then \( Y_1 \) and \( Y_3 \) are marginally dependent

\[ Y_1 \not\perp Y_3 \]

○ If \( Y_2 \) is observed then \( Y_1 \) and \( Y_3 \) are conditionally independent

\[ Y_1 \perp Y_3 | Y_2 \]

When \( Y_2 \) in observed is said to block the path from \( Y_1 \) to \( Y_3 \)
Head to Tail Connections

○ Corresponds to
\[
P(Y_1, Y_2, Y_3) = P(Y_1)P(Y_2|Y_1)P(Y_3|Y_2)
= P(Y_1|Y_2)P(Y_3|Y_2)P(Y_2)
\]

○ If $Y_2$ is unobserved then $Y_1$ and $Y_3$ are marginally dependent

\[Y_1 \perp Y_3\]

○ If $Y_2$ is observed then $Y_1$ and $Y_3$ are conditionally independent

\[Y_1 \perp Y_3|Y_2\]
Head to Head Connections

○ Corresponds to
\[ P(Y_1, Y_2, Y_3) = P(Y_1)P(Y_3)P(Y_2|Y_1, Y_3) \]

○ If \( Y_2 \) is observed then \( Y_1 \) and \( Y_3 \) are conditionally dependent
\[ Y_1 \not\perp Y_3 | Y_2 \]

○ If \( Y_2 \) is unobserved then \( Y_1 \) and \( Y_3 \) are marginally independent
\[ Y_1 \perp Y_3 \]

If any \( Y_2 \) descendants is observed it unlocks the path
Derived Conditional Independence Relationships

A Bayesian Network represents the local relationships encoded by the 3 basic structures plus the derived relationships.

Consider:

Local Markov Relationships:
\[ Y_1 \perp Y_3 | Y_2 \]
\[ Y_4 \perp Y_1, Y_2 | Y_3 \]

Derived Relationship:
\[ Y_1 \perp Y_4 | Y_2 \]
d-Separation

Definition (d-separation)

Let $r = Y_1 \leftrightarrow \cdots \leftrightarrow Y_2$ be an undirected path between $Y_1$ and $Y_2$, then $r$ is d-separated by $Z$ if there exist at least one node $Y_c \in Z$ for which path $r$ is blocked.

In other words, d-separation holds if at least one of the following holds

- $r$ contains an head-to-tail structure $Y_i \rightarrow Y_c \rightarrow Y_j$ (or $Y_i \leftarrow Y_c \leftarrow Y_j$) and $Y_c \in Z$
- $r$ contains a tail-to-tail structure $Y_i \leftarrow Y_c \rightarrow Y_j$ and $Y_c \in Z$
- $r$ contains an head-to-head structure $Y_i \rightarrow Y_c \leftarrow Y_j$ and neither $Y_c$ nor its descendants are in $Z$
# Markov Blanket and d-Separation

## Definition (Nodes d-separation)

Two nodes $Y_i$ and $Y_j$ in a BN $\mathcal{G}$ are said to be **d-separated** by $Z \subseteq \mathcal{V}$ (denoted by $\text{Dsep}_\mathcal{G}(Y_i, Y_j \mid Z)$) if and only if all undirected paths between $Y_i$ and $Y_j$ are d-separated by $Z$.

## Definition (Markov Blanket)

The Markov blanket $\text{Mb}(Y)$ is the minimal set of nodes which d-separates a node $Y$ from all other nodes (i.e. it makes $Y$ conditionally independent of all other nodes in the BN).

$$\text{Mb}(Y) = \{\text{pa}(Y), \text{ch}(Y), \text{pa}(\text{ch}(Y))\}$$
Are Directed Models Enough?

- Bayesian Networks are used to model asymmetric dependencies (e.g. causal)
- What if we want to model symmetric dependencies
  - Bidirectional effects, e.g. spatial dependencies
  - Need undirected approaches

Directed models cannot represent some (bidirectional) dependencies in the distributions

What if we want to represent $Y_1 \perp Y_3 | Y_2, Y_4$?
What if we also want $Y_2 \perp Y_4 | Y_1, Y_3$?

Cannot be done in BN! Need undirected model
Markov Random Fields

What is the undirected equivalent of $d$-separation in directed models?

Again it is based on node separation, although it is way simpler!

- Node subsets $A, B \subset \mathcal{V}$ are **conditionally independent** given $C \subset \mathcal{V}\setminus\{A, B\}$ if all paths between nodes in $A$ and $B$ pass through at least one of the nodes in $C$.

- The **Markov Blanket** of a node includes all and only its neighbors.
Joint Probability Factorization

What is the undirected equivalent of conditional probability factorization in directed models?

- We seek a product of functions defined over a set of nodes associated with some local property of the graph.
- Markov blanket tells that nodes that are not neighbors are conditionally independent given the remainder of the nodes:
  \[ P(X_v, X_i | X_{\mathcal{V}\setminus\{v, i\}}) = P(X_v | X_{\mathcal{V}\setminus\{v, i\}}) P(X_i | X_{\mathcal{V}\setminus\{v, i\}}) \]
- Factorization should be chosen in such a way that nodes \( X_v \) and \( X_i \) are not in the same factor.

What is a well-known graph structure that includes only nodes that are pairwise connected?
Cliques

Definition (Clique)
A subset of nodes $C$ in graph $G$ such that $G$ contains an edge between all pair of nodes in $C$

Definition (Maximal Clique)
A clique $C$ that cannot include any further node from the graph without ceasing to be a clique
Maximal Clique Factorization

Define $X = X_1, \ldots, X_N$ as the RVs associated to the $N$ nodes in the undirected graph $\mathcal{G}$

$$P(X) = \frac{1}{Z} \prod_C \psi(X_C)$$

- $X_C \rightarrow$ RV associated with nodes in the maximal clique $C$
- $\psi(X_C) \rightarrow$ potential function over the maximal cliques $C$
- $Z \rightarrow$ partition function ensuring normalization

$$Z = \sum_X \prod_C \psi(X_C)$$

Partition function is the computational bottleneck of undirected modes: e.g. $O(K^N)$ for $N$ discrete RV with $K$ distinct values
Potential Functions

- Potential functions $\psi(X_C)$ are not probabilities!
- Express which configurations of the local variables are preferred
- If we restrict to strictly positive potential functions, the Hammersley-Clifford theorem provides guarantees on the distribution that can be represented by the clique factorization

### Definition (Boltzmann distribution)

A convenient and widely used strictly positive representation of the potential functions is

$$
\psi(X_C) = \exp\{-E(X_C)\}
$$

where $E(X_C)$ is called energy function
From Directed To Undirected

Straightforward in some cases

Requires a little bit of thinking for v-structures

Moralization a.k.a. marrying of the parents
Learning Causation (from data)
Learning with Bayesian Networks

<table>
<thead>
<tr>
<th>Data</th>
<th>Complete</th>
<th>Incomplete</th>
</tr>
</thead>
<tbody>
<tr>
<td>Structure</td>
<td>Parameters Learning</td>
<td>Structure Learning</td>
</tr>
<tr>
<td>Fixed Structure</td>
<td>$P(Y</td>
<td>X)$</td>
</tr>
<tr>
<td>Fixed Variables</td>
<td>$P(X, Y)$</td>
<td></td>
</tr>
<tr>
<td>Naive Bayes</td>
<td>Calculate Frequencies (ML)</td>
<td></td>
</tr>
<tr>
<td>Latent variables</td>
<td>EM Algorithm (ML)</td>
<td></td>
</tr>
<tr>
<td>MCMC, VBEM (Bayesian)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

In summary:
- **Complete Data**:
  - Fixed Structure: Naive Bayes, Calculate Frequencies (ML)
  - Fixed Variables: Discover dependencies from the data, Structure Search
- **Incomplete Data**:
  - Latent variables: EM Algorithm (ML), MCMC, VBEM (Bayesian)
  - Discover dependencies from the data, Structural EM
The Structure Learning Problem

- Observations are given for a set of fixed random variables
- Network structure is not specified
  - Determine which arcs exist in the network (causal relationships)
  - Compute Bayesian network parameters (conditional probability tables)
- Determining causal relationships between variables entails
  - Deciding on arc presence
  - Directing edges
Structure Finding Approaches

- Search and Score
  - Model selection approach
  - Search in the space of the graphs

- Constraint Based
  - Use tests of conditional independence
  - Constrain the network

- Hybrid
  - Model selection of constrained structures
Search & Score

- Search the space $\text{Graph}(\mathbf{Y})$ of graphs $G_k$ that can be built on the random variables $\mathbf{Y} = Y_1, \ldots, Y_N$
- Score each structure by $S(G_k)$
- Return the highest scoring graph $G^*$
- Two fundamental aspects
  - Scoring function
  - Search strategy
Scoring Function

- Fundamental properties
  - **Consistency** - Same score for graphs in the same equivalence class
  - **Decomposability** - Can be locally computed

- Approaches
  - **Information theoretic** - Based on data likelihood plus some model-complexity penalization terms (AIC, BIC, MDL, ...)
  - **Bayesian** – Score the structures using a graph posterior (likelihood + proper prior choice)
Search Strategy

○ Finding maximal scoring structures is NP complete (Chickering, 2002)

○ Constrain search strategy
  ● Starting from a candidate structure modify iteratively by local operations (edge/node addition or deletion)
  ● Each operation has a cost
  ● Cost optimization problem: greedy hill-climbing, simulated annealing, ...

○ Constrain search space
  ● Known node order – Can reduce the search space to the parents of each node (Markov Blanket)
  ● Search in the space of structure equivalence classes (GES algorithm)
  ● Search in the space of node orderings (Friedman and Koller, 2003)
Constraint-based Models

- Tests of conditional independence $I(X_i, X_j|Z)$ determine edge presence (network skeleton)
- Based on measures of association between two variables/nodes $X_i$ and $X_j$, given their neighbor nodes $Z$
  - Conditional mutual information
  - Statistical hypothesis testing on association measures with a known distribution, e.g. $\chi^2$
- Use deterministic rules based on local Markovian dependencies to determine edge orientation (DAG)
Testing Strategy

○ Choice of the testing order is fundamental for avoiding a super-exponential complexity

○ Level-wise testing
  ● Tests $I(X_i, X_j|Z)$ are performed in order of increasing size of the conditioning set $Z$ (starting from empty $Z$)
  ● PC algorithm (Spirtes, 1995)

○ Node-wise testing
  ● Tests are performed on a single edge at the time, exhausting independence checks on all conditioning variables
  ● TPDA Algorithm

○ Nodes that enter $Z$ are chosen in the neighborhood of $X_i$ and $X_j$
PC Algorithm

Initialize a fully connected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$

for each edge $(Y_i, Y_j) \in \mathcal{V}$
  - if $I(Y_i, Y_j)$ then prune $(Y_i, Y_j)$

$K \leftarrow 1$

for each test of order $K = |Z|$
  - for each edge $(Y_i, Y_j) \in \mathcal{V}$
    - $Z \leftarrow$ set of conditioning sets of $K$-th order for $Y_i, Y_j$
    - if $I(Y_i, Y_j|z)$ for any $z \in Z$ then prune $(Y_i, Y_j)$
  - $K \leftarrow K + 1$

return $\mathcal{G}$
Hybrid Models

- Multi-stage algorithms combining previous approaches
- Independence tests to find a sub-optimal skeleton (good starting point)
- Search and score starting from the skeleton
  - Skeleton refinement
  - Edge orientation
- Max-Min Hill Climbing (MMHC) model
  - Optimized constraint-based approach to reconstruct the skeleton (Max-Min Parents and Children)
  - Use the candidate parents in the skeleton to run a search and score approach
Learning a COVID-19 causal model

Example of integration of clinical knowledge with causation information inferred from data
Take Home Messages

○ Directed graphical models
  ● Represent **asymmetric (causal) relationships** between RV and conditional probabilities in compact way
  ● Difficult to assess conditional independence (v-structures)
  ● Ok for **prior knowledge and interpretation**

○ Undirected graphical models
  ● Represent **bi-directional relationships** (e.g. constraints)
  ● Factorization in terms of generic **potential functions** (not probabilities)
  ● Easy to assess conditional independence, but **difficult to interpret**
  ● Serious **computational issues** due to normalization factor

○ Structure learning to **infer multivariate causation relationships** from data
Next Two Lectures

Hidden Markov Model (HMM)
- A dynamic graphical model for sequences
- Unfolding learning models on structures
- Exact inference on a chain with observed and unobserved variables
- The Expectation-Maximization algorithm for HMMs