

Probabilistic Modeling and Reasoning

Traiko Dinev <traiko.dinev@gmail.com>

July 1, 2019

NOTE: This partially follows Probabilistic Modeling and Reasoning, a masters level course at the University of Edinburgh.

NOTE: Note this "summary" is NOT a reproduction of the course materials nor is it copied from the corresponding courses. It was entirely written and typeset from scratch.

License: Creative Commons public license; See README.md of repository

1 Probability Identities

Non-exhaustive list of identities useful for the rest of this cheatsheet:

$$\begin{aligned} P(A, B) &= P(A \mid B) P(B) && \text{product rule} \\ P(A) &= \sum_B P(A \mid B) P(B) = \sum_B P(A, B) && \text{sum rule} \\ x \perp\!\!\!\perp y &\iff P(x, y) = P(x) P(y) \\ x_1 \perp\!\!\!\perp x_2 \perp\!\!\!\perp \dots, x_N &\iff P(x_1, \dots, x_N) = \prod_i P(x_i) \end{aligned}$$

We can store discrete distributions as tables of data. Conditional independence allows us to save space. Consider the conditional independence rules first:

$$\begin{aligned} P(x, y \mid z) &= P(x \mid z) P(y \mid z) \iff x \perp\!\!\!\perp y \mid z \\ P(x \mid y, z) &= P(x \mid z) \end{aligned}$$

Then to store $P(x, y, z) = P(x)P(y)P(z)$ we would need $\dim(x) \times \dim(y) \times \dim(z)$ space. If they are all equal this means $k^{3d} - 1$ entries. The factorization allows us to have $3(k^d - 1)$ entries instead.

2 Directed Graphical Models

From the chain rule, we can factorize any distribution as:

$$P(\mathbf{x}) = \prod_i P(x_i \mid \pi_i), \quad \pi_i = \{x_1, \dots, x_{i-1}\}$$

We can prove that the following holds true by induction:

$$P(\mathbf{x}) = \prod_i P(x_i \mid \pi_i) \iff x_i \perp\!\!\!\perp (\text{pre}_i \setminus \pi_i) \mid \pi_i, \forall i$$

where π_i is some subset of elements. This is to say that the factorization implies independence and a set of independences implies a factorization.

Thus we can visualize a distribution by drawing a DAG (directed acyclic graph) where the parents are the above π_i 's. Thus if:

$$P(\mathbf{x}) = P(x_1) P(x_2) P(x_3 \mid x_1, x_2) P(x_4 \mid x_3) P(x_5 \mid x_2)$$

then the DAG is in Figure 1.

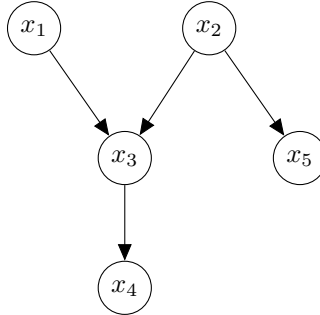


Figure 1: Simple DAG

A graph can be generated from a distribution and a (topological) ordering of the elements. A topological ordering is one where the parents come before the children. Note that different orderings may generate different graphs.

2.1 Examples

Markov Models (of order 1) or chains are a series of serial connections (Figure 2).

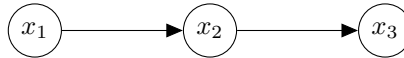


Figure 2: Markov Chain

Hidden Markov Models contain a markov chain that is not observed. Each hidden \mathbf{h} influences an observable. \mathbf{x} 's are often at different timesteps, making the chain represent a time series. Figure 3.

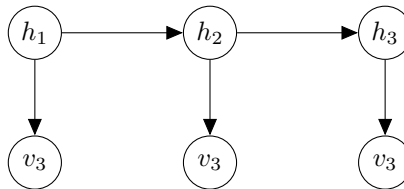


Figure 3: Hidden Markov Model

Probabilistic PCA/ Independent Component Analysis are methods that both use the same graphical model. Here the latents (hiddens) variables are not connected. At the same time they influence all of the observables. Figure 4

2.2 D-Separation

The main reason for using graphical models is to more easily determine independencies between variables. In a DAG a tool for using this is D-separation. We start by examining the three possible trail connections in a DAG.

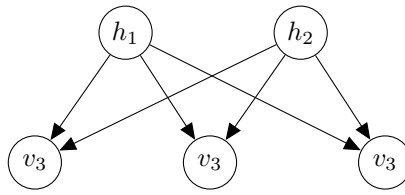


Figure 4: PPCA/ ICA graphical model

Note that these are not all possible connections between three elements, but rather *all possible connections when following a trail*.

Serial Connections are like Markov chains.

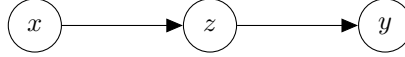


Figure 5: Serial Connecton

Importantly we have:

$$P(x, z, y) = P(x)P(z | x)P(y | z)$$

$$x \perp\!\!\!\perp y \mid z \quad x \not\perp\!\!\!\perp y$$

This means that if we know the variable z , x and y , **and all their parents and children** that are not connected to z are independent of each other.

Diverging Connections

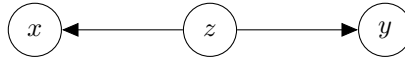


Figure 6: Diverging Connecton

The same property of independence holds true here:

$$P(x, z, y) = P(z)P(x | z)P(y | z)$$

$$x \perp\!\!\!\perp y \mid z \quad x \not\perp\!\!\!\perp y$$

Converging Connections (Colliders)

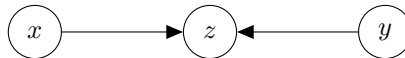


Figure 7: Collider

For colliders if we **do not know** z , x and y are independent:

$$P(x, z, y) = P(z)P(x | z)P(y | z)$$

$$x \perp\!\!\!\perp y \quad x \not\perp\!\!\!\perp y \mid z$$

This is true since $P(\cdot) = \dots$

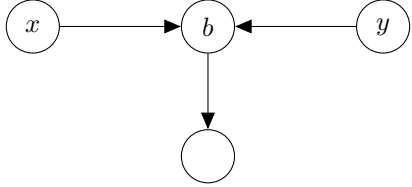
D-Separation Sets X and Y are d-separated by Z iff all trails are blocked by Z . One of the following needs to be true for a trail to be blocked:

1) Either b is in a head-tail or tail-tail configuration



and b is in Z .

2) b is a part of a collider



and neither b or its descendents are in z . **Then** $X \perp\!\!\!\perp Y \mid Z$.

2.3 I-Maps

A graph is an I-map for a set of independencies I iff all independencies asserted by the graph are part of I . A graph can thus have fewer independencies than the set. A fully-connected graph is a trivial I-map for all sets I .

2.4 Directed Local Markov Property

$$\mathbf{x}_i \perp\!\!\!\perp (\text{pre}_i \setminus \text{pa}_i) \mid \text{pa}_i \leftrightarrow \mathbf{x}_i \perp\!\!\!\perp (\text{nondesc}(\mathbf{x}_i) \setminus \text{pa}_i) \mid \text{pa}_i$$

This [todo] figure.

2.5 Global directed Markov Property

All independencies asserted by D-separation.

2.6 Markov Blanket

By definition:

$$x \perp\!\!\!\perp (\text{all} \setminus \mathbf{x} \setminus \text{MB}(\mathbf{x})) \mid \text{MB}(\mathbf{x})$$

And for DAGs we get:

$$\text{MB}(\mathbf{x}) = \text{pa}(\mathbf{x}) \cup \text{children}(\mathbf{x}) \cup \text{co-parents}(\mathbf{x})$$

3 Undirected Graphical Models

Firstly, we note the following. For non-negative functions a and b :

$$\begin{aligned}
 x \perp\!\!\!\perp y \mid z &\leftrightarrow P(x, y, z) = a(x, z) \times b(y, z) \\
 x \perp\!\!\!\perp y &\leftrightarrow P(x, y) = a(x) \times b(y) \\
 \sum_{x, y, z} a(x, z) b(y, z) &= 1 \\
 \text{if } p(x, y, z) &= \frac{1}{Z} \phi_A(x, z) \phi_B(y, z), \quad Z = \sum_{x, y, z} \phi_A(x, z) \phi_B(y, z)
 \end{aligned}$$

3.1 Gibbs Distribution

.. is a distribution that factorizes as:

$$P(\mathbf{x}) = \frac{1}{Z} \prod_c \phi_C(\mathcal{X}_C), \quad \mathcal{X}_C \subseteq \{x_1, \dots, x_d\}$$

3.2 Energy-Based Model

If in the above $\phi_C(\mathcal{X}_C) = \exp(-E_c(\mathcal{X}_c))$. Then:

$$P(\mathbf{x}) = \frac{1}{Z} = \frac{1}{Z} \exp \left[- \sum_c E_c(\mathcal{X}_c) \right] = \frac{1}{Z} \prod_c \underbrace{\exp^{-E_c(\mathcal{X}_c)}}_{\phi_c(\mathcal{X}_c)}$$

3.3 Undirected Graphs

Assuming a distribution (up to a constant) factorizes as:

$$P(\mathbf{x}) \propto \phi_1(x_1, x_2, x_3) \phi_2(x_2, x_3, x_4) \phi_3(x_3, x_5) \phi_4(x_5, x_6)$$

then we visualize it as the following graph:

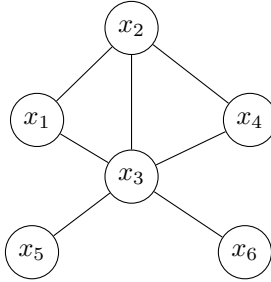


Figure 8: Undirected Graph

We form cliques for all variables in each factor ϕ_i .

3.4 Independencies in Undirected Models

If

$$P(\mathbf{x}) \propto \phi_1(x_1, x_2) \phi_2(x_2, x_3) \phi_3(x_4)$$

Then the corresponding graph is:



This directly implies that

$$\begin{aligned} x_4 &\perp\!\!\!\perp x_1, x_2, x_3 \\ x_1 &\perp\!\!\!\perp x_3 \mid x_2 \end{aligned}$$

In other words, a trail is blocked if there are no paths between the two nodes. Thus D-separation is more easily done here.

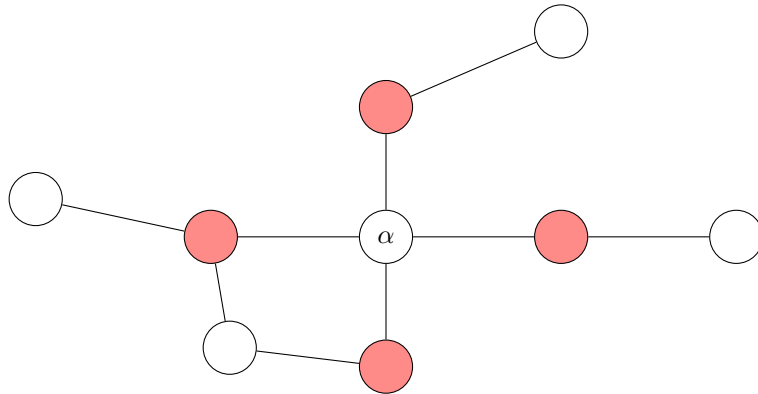
3.5 Graph \rightarrow Distribution

Since we built undirected graphs by connecting cliques, it follows that given a graph we look at the maximum cliques to recover the distribution. **I-maps** are defined as before.

3.6 Local Markov Property

An edge is independent of all other edges given its neighbors. In the below graph the neighbors are colored in red. Formally, if $\text{ne}(\alpha)$ are the neighbors of α , then:

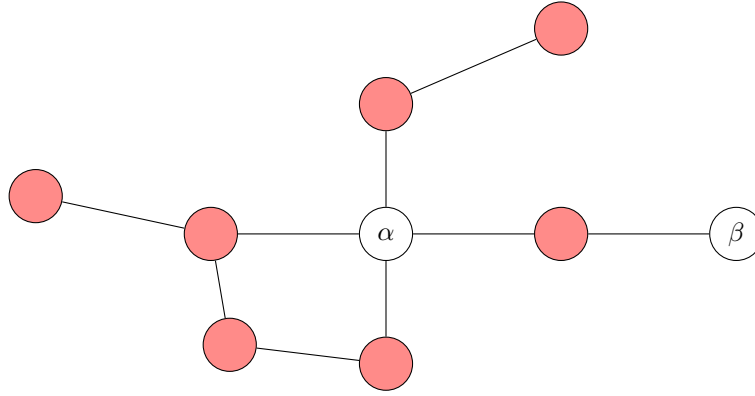
$$\alpha \perp\!\!\!\perp X \setminus (\alpha \cup \text{ne}(\alpha)) \mid \text{ne}(\alpha), \quad \forall \alpha \in X$$



3.7 Pairwise Markov Property

$$\alpha \perp\!\!\!\perp \beta \mid \underbrace{X}_{\text{all nodes}} \setminus \{\alpha, \beta\}$$

for all non-neighboring $\alpha, \beta \in X$.



3.8 Markov Blanket

For undirected graph, the markov blanket of x is the neighbors of x .

N.B. All of the above properties (Markov properties) are equivalent for both directed and undirected graphical models. This means if one of them is true **for a distribution**, then all of them are true.

TODO: Add Bishop plots with repetition boxes.

3.9 Minimal I-map

- If an edge is removed, it ceases to be an imap.
- A graph is an I-map if $P(\cdot)$ factorizes over the graph.

Undirected Models

- For all \mathbf{x}_i find $MB(\mathbf{x}_i)$ and connect.

Directed Models

- For all \mathbf{x}_i find $\pi_i \subseteq \text{pre}_i$ such that $\mathbf{x}_i \perp\!\!\!\perp \{\text{pre}_i \setminus \pi_i\} \mid \pi_i$
- Set $\text{pa}_i = \pi_i$

4 Equivalence and Conversion Between Models

Two directed graphs G_1 and G_2 are I -equivalent if they have the same set of immoralities and the same skeleton. An immorality is a collider without covering edge. **Look for colliders that don't match.** Since serial (head-tail) and diverging (tail-tail) connections imply the same independencies, we only need to look for converging (head-head) connections that don't match.

4.1 Directed \rightarrow Undirected

We have:

$$P(\cdot) = \prod_i P(\mathbf{x}_i \mid \text{pa}_i) = \prod_i \underbrace{\phi_i(\mathbf{x}_i, \text{pa}_i)}_{\text{cliques}}$$

- This is called **moralization** and we obtain a **moral graph**
- This is **NOT** an undirected I-map for the distribution. Most notably we can not represent collider independencies.

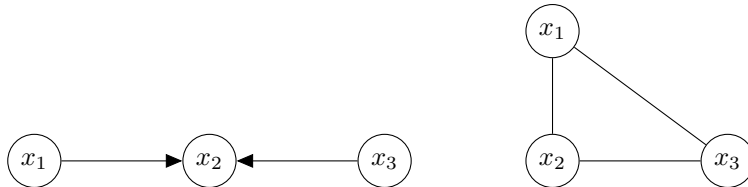
4.2 Undirected \rightarrow Directed

(See example below)

- Choose an ordering.
- Read independencies off of the graph and find π_i for each i .
- Connect $\pi_i \rightarrow \mathbf{x}_i$

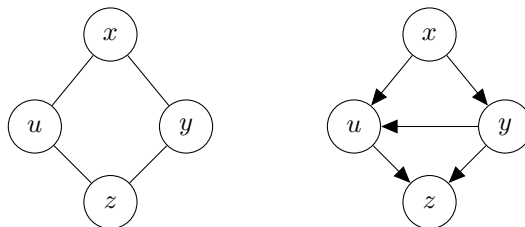
4.3 Non-equivalent Trails

Colliders can not be represented by undirected graphs.



In the moralized graph on the right the independence $\mathbf{x}_1 \perp\!\!\!\perp \mathbf{x}_3$ is lost.

Closed loops can not be represented by directed graphs.



Consider the left graph. Let's review the process of creating the directed graph:

- Choose an ordering: x, y, u, z .
- For each element, consider the parent set and read independencies off the directed graph.

- Start with $y \not\perp x$, which implies the edge $x \rightarrow y$.
- Find the minimal set for which u is independent of the parents π_u . This is x, y . Hence $x, y \rightarrow u$.
- For z this set is u, y . Hence $u \rightarrow z$ and $y \rightarrow z$

We no longer have the independence $u \perp y \mid x, z$.

5 Factor Graphs

$$P(\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3, \mathbf{x}_4) = \frac{1}{Z} \phi_1(\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3) \phi_2(\mathbf{x}_3, \mathbf{x}_4) \phi_3(\mathbf{x}_4) \quad (1)$$

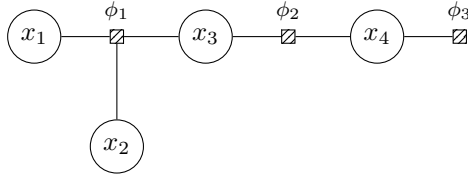


Figure 9: Factor Graph

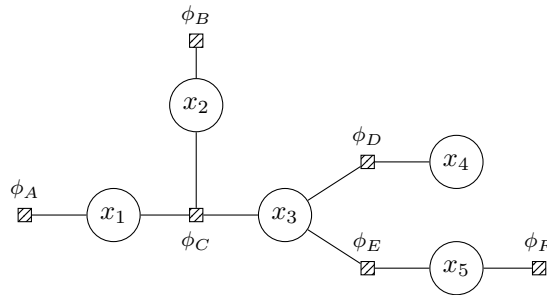
6 Exact Inference in Factor Graphs

Assume discrete variables. The task is to compute $P(\mathbf{x}_k = k)$ for all k .

We can group terms in the leaves of the tree. Consider:

$$P(\cdot) \propto \phi_A(\mathbf{x}_1) \phi_B(\mathbf{x}_2) \phi_C(\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3) \phi_D(\mathbf{x}_3, \mathbf{x}_4) \phi_E(\mathbf{x}_3, \mathbf{x}_5) \phi_F(\mathbf{x}_5) \quad (2)$$

which is:



We iteratively "eliminate" variables by summing or integrating them out. Firstly, we eliminate \mathbf{x}_5 :

$$\begin{aligned}
P(\mathbf{x}_1, \dots, \mathbf{x}_4) &= \sum_{\mathbf{x}_5} P(\mathbf{x}_1, \dots, \mathbf{x}_5) \\
&\propto \sum_{\mathbf{x}_5} \phi_A(\mathbf{x}_1) \phi_B(\mathbf{x}_2) \phi_C(\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3) \phi_D(\mathbf{x}_3, \mathbf{x}_4) \phi_E(\mathbf{x}_3, \mathbf{x}_5) \phi_F(\mathbf{x}_5) \\
&\propto \phi_A(\mathbf{x}_1) \phi_B(\mathbf{x}_2) \phi_C(\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3) \phi_D(\mathbf{x}_3, \mathbf{x}_4) \sum_{\mathbf{x}_5} \phi_E(\mathbf{x}_3, \mathbf{x}_5) \phi_F(\mathbf{x}_5) \\
&\propto \phi_A(\mathbf{x}_1) \phi_B(\mathbf{x}_2) \phi_C(\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3) \phi_D(\mathbf{x}_3, \mathbf{x}_4) \tilde{\phi}_5(\mathbf{x}_3)
\end{aligned}$$

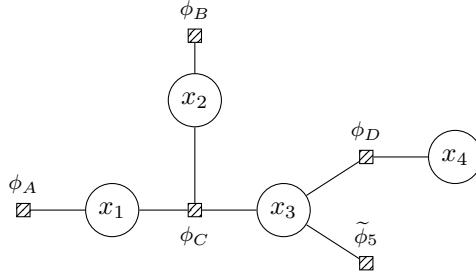
The idea is that we have reduced the factors above. Numerically, we would have the following:

$$\tilde{\phi}_5(\mathbf{x}_3) = \begin{cases} a & \mathbf{x}_3 = 1 \\ \dots & \\ z & \mathbf{x}_3 = N \end{cases}$$

For each value of the factor. This is pre-computing all of the values, which in this case costs $O(N^2)$. For each value of \mathbf{x}_3 we need to sum over \mathbf{x}_5 .

If we keep on doing this, the total cost will be greatly reduced from the $O(N^4)$ that is needed to sum over $\mathbf{x}_2, \mathbf{x}_3, \mathbf{x}_4, \mathbf{x}_5$.

The above operation is represented by the following **reduced** factor graph:



6.1 Message Passing

7 Inference for Markov Chains

7.1 α -recursion

7.2 Smoothing

7.3 α - β Recursion

8 Model-Based Learning

9 Moment Matching

10 Bayesian Inference

11 Factor Analysis

12 Independent Component Analysis (ICA)

13 Intractable Likelihood Functions

14 Score Matching

15 Sampling and Monte Carlo

16 Sampling from Continuous Distributions

16.1 Rejection Sampling

16.2 Ancestral Sampling

16.3 Gibbs Sampling

17 Variational Inference