

Lecture Notes for CS 726 - Spring 2021

Eeshaan Jain

January 10, 2022

These are my lecture notes taken during the Advanced Machine Learning (CS 726) course at IIT Bombay during the Spring 2021 session.

Contents

<i>Probabilistic Modeling</i>	2
<i>Alternatives to explicit joint distributions</i>	2
<i>Bayesian Networks</i>	3
<i>Definition</i>	3
<i>Minimal Construction</i>	4
<i>D-Separation</i>	6

Probabilistic Modeling

Given a set of n random variables $\mathcal{X} = \{X_1, X_2, \dots, X_n\}$ where n is large, we want to build a joint probability distribution P over this set. Explicitly representing the joint distribution is computationally expensive, since just having binary values variables requires the joint distribution to specify $2^n - 1$ numbers, and for more practical variables, the count is too large.

We want to efficiently represent, estimate and answer inference queries on the distribution.

An example of a query can be -
Estimate the fraction of people
with a bachelor's degree.

Alternatives to explicit joint distributions

▷ Can we assume all columns are independent? **NO** - this is obviously a very bad assumption.

▷ Can we use data to detect highly correlated column pairs, and estimate their pairwise frequencies? **MAYBE** - but there might be too many correlated pairs, and the method is ad hoc.

To solve the above two not so good ways, we explore conditional independencies. It may be possible that income $\not\perp$ age but income \perp age|experience.

Note that we write that a set X is
conditionally independent of Y given Z ,
i.e. $X \perp\!\!\!\perp Y | Z$ if

$$\Pr(X|Y, Z) = \Pr(X|Z)$$

Probabilistic graphical models use a graph-based representation as the basis for compactly encoding a complex distribution over a high-dimensional space.

It is convenient to represent the independence assumption using a graph. The so called graphical model has nodes as the variables (continuous or discrete), and the edges represent direct interaction. If we consider directed edges, we talk about Bayesian Networks, and if we consider undirected edges, we talk about Markov Random Fields.

Essentially the graphical model is a combination of the graph and potentials.

Definition 1 (Potentials). Potentials $\psi_c(\mathbf{x}_c)$ are scores for assignment of values to subsets c of directly interacting variables. We factorize the probability as a product of these potentials, i.e

$$\Pr(\mathbf{x} = x_1, \dots, x_n) \propto \prod \psi_s(\mathbf{x}_s) \quad (1)$$

Bayesian Networks

Bayesian Networks, also referred to as *directed graphical models* are a family of probability distributions that has a compact parameterization representable using a directed graph.

It is known that

$$\Pr(x_1, x_2, \dots, x_n) = \Pr(x_1)\Pr(x_2|x_1)\Pr(x_3|x_2, x_1) \cdots \Pr(x_n|x_{n-1}, \dots, x_1) \quad (2)$$

A compact Bayesian Network is a distribution in which each factor in the above equation depends on the *parent* variables represented by $\text{Pa}(x_i)$ for variable x_i . Thus, we have

$$\Pr(x_i|x_{i-1}, x_{i-2}, \dots, x_1) = \Pr(x_i|\text{Pa}(x_i)) \quad (3)$$

and the corresponding potentials at each node in terms of its parents are

$$\psi_i(x_i, \text{Pa}(x_i)) = \Pr(x_i, \text{Pa}(x_i)) \quad (4)$$

Thus,

$$\Pr(x_1, x_2, \dots, x_n) = \prod_{i=1}^n \Pr(x_i|\text{Pa}(x_i)) \quad (5)$$

Consider the situation when each variable can take d values. The naive approach gives us $\mathcal{O}(d^n)$ parameters. If we think of the potentials as probability tables (with the rows corresponding to $\text{Pa}(x_i)$) and columns corresponding to the values of x_i , with entries as $\psi_i(x_i, \text{Pa}(x_i))$, we can notice that if $|\text{Pa}(x_i)| \leq k$, then the number of parameters are $\mathcal{O}(d^{k+1})$, and for n variables, we have $\mathcal{O}(nd^{k+1})$, which provides us the compact representation.

Definition

Now we formally define these -

Definition 3 (Bayesian Network). A Bayesian Network is a directed graph $G = (V, E)$ together with

- ◇ a random variable x_i for each node $i \in V$
- ◇ a potential $\psi_i(x_i, \text{Pa}(x_i))$ for each node $i \in V$

For a variable x_i in our Bayesian Network \mathcal{G} , denote $\text{ND}(x_i)$ as the non-descendants of x_i . The following local conditional independencies hold in \mathcal{G} -

$$x_i \perp\!\!\!\perp \text{ND}(x_i) | \text{Pa}(x_i) \quad (6)$$

Example 2 shows the independencies in a simple Bayesian Network.

Example 2.

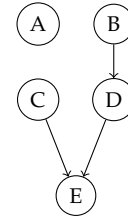


Figure 1: Sample BN

Consider the BN above. We will consider each variable at a time.

- ◇ A has no parent, and has no descendent. Thus,

$$A \perp\!\!\!\perp B, C, D, E$$

- ◇ B has no parent, but has D as a descendent. Thus,

$$B \perp\!\!\!\perp A, C$$

- ◇ C has no parent, but has E as a descendent. Thus,

$$C \perp\!\!\!\perp A, B, D$$

- ◇ D has B as a parent, and has E as the descendent. Thus,

$$D \perp\!\!\!\perp A, C | B$$

- ◇ E has C and D as parents, but has no descendent. Thus,

$$I \perp\!\!\!\perp A, B | C, D$$

Definition 4 (Factorization). Let \mathcal{G} be a Bayesian Network graph over the variables $\{X_i\}_{i=1}^n$. We say that a distribution P over the same space factorizes according to \mathcal{G} if P can be expressed as a product described in Equation 5. Such factorization is also known as the chain rule for Bayesian Networks, and is denoted as $\text{Factorize}(P, \mathcal{G})$.

Definition 5. Let P be a distribution over \mathcal{X} . We define $\mathcal{I}(P)$ to be the set of independent assertions of the form $\mathbf{X} \perp\!\!\!\perp \mathbf{Y} | \mathbf{Z}$ that hold in P .

We can now write " P satisfies the local independencies associated with \mathcal{G} " as $\mathcal{I}_\ell(\mathcal{G}) \subseteq \mathcal{I}(P)$.

Definition 6 (Independency-Map). Let \mathcal{K} be any graph object associated with a set of independencies $\mathcal{I}(\mathcal{K})$. We call \mathcal{K} an I-map for a set of independencies \mathcal{I} if $\mathcal{I}(\mathcal{K}) \subseteq \mathcal{I}$.

Thus for \mathcal{G} to be an I-map for P , any independence that asserts in \mathcal{G} must also assert in P , but P can have additional independencies not reflected in \mathcal{G} .

Remark 7 (Notation Alert). Note that we will use the following interchangeably - P satisfies the local conditional independencies satisfied by \mathcal{G} and \mathcal{G} is an I-map for P , i.e

$$\text{Local-CI}(P, \mathcal{G}) \equiv \mathcal{I}(\mathcal{G}) \subseteq \mathcal{I}(P) \quad (7)$$

Theorem 8. Given a distribution $P(x_1, x_2, \dots, x_n)$ and a directed acyclic graph (DAG) \mathcal{G} ,

$$\text{Local-CI}(P, \mathcal{G}) \iff \text{Factorize}(P, \mathcal{G}) \quad (8)$$

Proof. (\implies) We essentially need to show that if \mathcal{G} is an I-map for P , then P factorizes according to \mathcal{G} . Consider a topologically sorted order x_1, x_2, \dots, x_n in \mathcal{G} . Local-CI(P, \mathcal{G}) tells us that

$$\Pr(x_i | x_1, \dots, x_{i-1}) = \Pr(x_i | \text{Pa}(x_i))$$

We can write

$$P(x_1, x_2, \dots, x_n) = \prod_{i=1}^n P(x_i | x_1, \dots, x_{i-1})$$

Each term in the product can be simplified due to the notion of Local-CI stated above, and we reach Equation 5, proving factorization.

(\impliedby) Proof has been skipped. \square

Minimal Construction

Our goal is to construct a minimal and correct BN \mathcal{G} to represent P . A DAG \mathcal{G} is correct if all Local-CIs that are implied in \mathcal{G} hold in P , and

a DAG \mathcal{G} is minimal if we cannot remove any edge(s) from \mathcal{G} and still get a correct BN for P .

In the setting, we define our oracle \mathcal{O} to whom we can ask any query of the type "Is $X \perp\!\!\!\perp Y|Z$?" pertaining to P and get a boolean answer. We will query the oracle several times to build up our BN. The following algorithm constructs such a BN -

```

1 Variables:  $x_1, x_2, \dots, x_n \leftarrow$  ordered variables in  $\mathcal{X}$ 
2 Independencies:  $\mathcal{I} \leftarrow$  set of independencies
3  $\mathcal{G} \leftarrow$  Empty graph over  $\mathcal{X}$ 
4 for  $i = 1$  to  $n$  do
5    $\mathbf{U} \leftarrow \{x_1, \dots, x_{i-1}\}$  // Set of candidate parents of  $x_i$ 
6   for  $U' \subseteq \{x_1, \dots, x_{i-1}\}$  do
7     if  $U' \subset \mathbf{U}$  and  $(x_i \perp\!\!\!\perp \{x_1, \dots, x_{i-1}\} - U' | U') \in \mathcal{I}$  then
8        $\mathbf{U} \leftarrow U'$ 
9     end
10  end
11  // Now we have the minimal set  $\mathbf{U}$  satisfying
12     $(x_i \perp\!\!\!\perp \{x_1, \dots, x_{i-1}\} - \mathbf{U} | \mathbf{U})$ 
13  // Now we set  $\mathbf{U}$  to be the parents of  $x_i$ 
14  for  $x_j \in \mathbf{U}$  do
15    Add  $x_j \rightarrow x_i$  in  $\mathcal{G}$ 
16  end
17 return  $\mathcal{G}$ 

```

Algorithm 1: Minimal Bayesian Network Construction (I-Map)

We know sketch rough proofs for the claims of the algorithm.

Theorem 9. The BN \mathcal{G} constructed by algorithm 1 is minimal, i.e we cannot remove any edge from the BN while maintaining the correctness of the BN for P .

Proof. By construction. A subset of $\text{ND}(x_i)$ were available when we chose parents of \mathbf{U} minimally. \square

Theorem 10. \mathcal{G} constructed by the above algorithm is correct, i.e, the local-CIs induced by \mathcal{G} hold in P .

Proof. The construction is such that $\text{Factorize}(P, \mathcal{G})$ holds everytime. Since $\text{Factorize}(P, \mathcal{G}) \implies \text{Local-CI}(P, \mathcal{G})$, the constructed BN satisfies the local-CIs of P . \square

Remark 12 (Importance of ordering). It is possible that a different ordering in \mathcal{X} gives rise to a different BN, which although may be minimal, but may not be *optimal*. A minimal BN is defined for a given ordering, while an optimal BN is defined over all orderings. Example 11 shows such a case.

Example 11. To be added.

D-Separation

Our goal is to know when we can guarantee $\mathbf{X} \perp\!\!\!\perp \mathbf{Y} | \mathbf{Z}$ holds given a BN \mathcal{G} . The further discussion provides some cases where we can guarantee $\mathbf{X} \not\perp\!\!\!\perp \mathbf{Y} | \mathbf{Z}$.

1. **Direct Connection:** If there is an edge $X \rightarrow Y$, then regardless of any \mathbf{Z} , we can find examples where they influence each other.
2. **Indirect Connection:** [To be added]