Probabilistic Graphical Models

David Sontag

New York University

Lecture 1, January 26, 2012
One of the **most exciting advances** in machine learning (AI, signal processing, coding, control, ...) in the last decades
How can we gain **global insight** based on **local observations**?
Key idea

1. **Represent** the world as a collection of random variables $X_1, \ldots, X_n$ with joint distribution $p(X_1, \ldots, X_n)$

2. **Learn** the distribution from data

3. **Perform “inference”** (compute conditional distributions $p(X_i \mid X_1 = x_1, \ldots, X_m = x_m)$)
Reasoning under uncertainty

- As humans, we are continuously making predictions under uncertainty
- Classical AI and ML research ignored this phenomena
- Many of the most recent advances in technology are possible because of this new, *probabilistic*, approach
Applications: Deep question answering
The top U.S. general, visiting Israel at a delicate and dangerous moment in the global standoff with Tehran, is expected to press for restraint amid fears that the Jewish state is nearing a decision to attack Iran's nuclear program.

El máximo general de EE.UU., de visita en Israel en un momento delicado y peligroso en el enfrentamiento global con Teherán, se espera que presione a la moderación en medio de temores de que el estado judío se acerca a una decisión de atacar el programa nuclear de Irán.

New! Hold down the shift key, click, and drag the words above to reorder. Dismiss
Applications: Speech recognition

“I need to hide a body”

What kind of place are you looking for?

- reservoirs
- metal foundries
- mines
- dumps
- swamps
Applications: Stereo vision

**input:** two images

**output:** disparity
Key challenges

1. **Represent** the world as a collection of random variables $X_1, \ldots, X_n$ with joint distribution $p(X_1, \ldots, X_n)$
   - How does one *compactly describe* this joint distribution?
   - Directed graphical models (Bayesian networks)
   - Undirected graphical models (Markov random fields, factor graphs)

2. **Learn** the distribution from data
   - Maximum likelihood estimation. Other estimation methods?
   - How much data do we need?
   - How much computation does it take?

3. **Perform “inference”** (compute conditional distributions $p(X_i \mid X_1 = x_1, \ldots, X_m = x_m)$)
We will study Representation, Inference & Learning
First in the simplest case
  Only discrete variables
  Fully observed models
  Exact inference & learning
Then generalize
  Continuous variables
  Partially observed data during learning (hidden variables)
  *Approximate* inference & learning
Learn about algorithms, theory & applications
Logistics

- **Class webpage:**
  - [http://cs.nyu.edu/~dsontag/courses/pgm12/](http://cs.nyu.edu/~dsontag/courses/pgm12/)
  - Sign up for mailing list!
  - Draft slides posted before each lecture

- **Book:** *Probabilistic Graphical Models: Principles and Techniques* by Daphne Koller and Nir Friedman, MIT Press (2009)

- **Office hours:** Tuesday 5-6pm and by appointment. 715 Broadway, 12th floor, Room 1204

- **Grading:** problem sets (70%) + final exam (30%)
  - Grader is Chris Alberti (chris.alberti@gmail.com)
  - 6-7 assignments (every 2 weeks). Both theory and programming
  - First homework out **today**, due Feb. 9 at 5pm
  - See collaboration policy on class webpage
Quick review of probability

Reference: Chapter 2 and Appendix A

- What are the possible outcomes?
  Coin toss: $\Omega = \{\text{"heads"}, \text{"tails"}\}$
  Die: $\Omega = \{1, 2, 3, 4, 5, 6\}$

- An **event** is a subset of outcomes $S \subseteq \Omega$:
  Examples for die: $\{1, 2, 3\}, \{2, 4, 6\}, \ldots$

- We **measure** each event using a probability function
Probability function

- Assign non-negative weight, \( p(\omega) \), to each outcome such that
  \[
  \sum_{\omega \in \Omega} p(\omega) = 1
  \]

  - Coin toss: \( p(\text{“head”}) + p(\text{“tail”}) = 1 \)
  - Die: \( p(1) + p(2) + p(3) + p(4) + p(5) + p(6) = 1 \)

- Probability of event \( S \subseteq \Omega \):
  \[
  p(S) = \sum_{\omega \in S} p(\omega)
  \]

- Example for die: \( p(\{2, 4, 6\}) = p(2) + p(4) + p(6) \)

- Claim: \( p(S_1 \cup S_2) = p(S_1) + p(S_2) - p(S_1 \cap S_2) \)
Independence of events

Two events $S_1, S_2$ are independent if

$$p(S_1 \cap S_2) = p(S_1)p(S_2)$$
Conditional probability

- Let $S_1, S_2$ be events, $p(S_2) > 0$.

$$p(S_1 \mid S_2) = \frac{p(S_1 \cap S_2)}{p(S_2)}$$

- Claim 1: $\sum_{\omega \in S} p(\omega \mid S) = 1$

- Claim 2: If $S_1$ and $S_2$ are independent, then $p(S_1 \mid S_2) = p(S_1)$
Two important rules

1. **Chain rule** Let $S_1, \ldots, S_n$ be events, $p(S_i) > 0$.

   \[
p(S_1 \cap S_2 \cap \cdots \cap S_n) = p(S_1)p(S_2 | S_1) \cdots p(S_n | S_1, \ldots, S_{n-1})
   \]

2. **Bayes’ rule** Let $S_1, S_2$ be events, $p(S_1) > 0$ and $p(S_2) > 0$.

   \[
p(S_1 | S_2) = \frac{p(S_1 \cap S_2)}{p(S_2)} = \frac{p(S_2 | S_1)p(S_1)}{p(S_2)}
   \]
Discrete random variables

- Often each outcome corresponds to a setting of various attributes (e.g., “age”, “gender”, “hasPneumonia”, “hasDiabetes”)
- A random variable $X$ is a mapping $X: \Omega \rightarrow D$
  - $D$ is some set (e.g., the integers)
  - Induces a partition of all outcomes $\Omega$
- For some $x \in D$, we say
  \[
p(X = x) = p(\{ \omega \in \Omega : X(\omega) = x \})
  \]
  “probability that variable $X$ assumes state $x$”
- Notation: $\text{Val}(X) = \text{set } D$ of all values assumed by $X$
  (will interchangeably call these the “values” or “states” of variable $X$)
- $p(X)$ is a distribution: $\sum_{x \in \text{Val}(X)} p(X = x) = 1$
Multivariate distributions

- Instead of one random variable, have random vector
  \[X(\omega) = [X_1(\omega), \ldots, X_n(\omega)]\]

- \(X_i = x_i\) is an event. The joint distribution
  \[p(X_1 = x_1, \ldots, X_n = x_n)\]
  is simply defined as \(p(X_1 = x_1 \cap \cdots \cap X_n = x_n)\)

- We will often write \(p(x_1, \ldots, x_n)\) instead of \(p(X_1 = x_1, \ldots, X_n = x_n)\)

- Conditioning, chain rule, Bayes’ rule, etc. all apply
For example, the **conditional distribution**

\[
p(X_1 \mid X_2 = x_2) = \frac{p(X_1, X_2 = x_2)}{p(X_2 = x_2)}.
\]

This notation means

\[
p(X_1 = x_1 \mid X_2 = x_2) = \frac{p(X_1 = x_1, X_2 = x_2)}{p(X_2 = x_2)} \quad \forall x_1 \in \text{Val}(X_1)
\]

Two random variables are **independent**, \( X_1 \perp X_2 \), if

\[
p(X_1 = x_1, X_2 = x_2) = p(X_1 = x_1)p(X_2 = x_2)
\]

for all values \( x_1 \in \text{Val}(X_1) \) and \( x_2 \in \text{Val}(X_2) \).
Example

- Consider three binary-valued random variables
  
  \[ X_1, X_2, X_3 \quad \text{Val}(X_i) = \{0, 1\} \]

- Let outcome space \( \Omega \) be the cross-product of their states:
  
  \[ \Omega = \text{Val}(X_1) \times \text{Val}(X_2) \times \text{Val}(X_3) \]

- \( X_i(\omega) \) is the value for \( X_i \) in the assignment \( \omega \in \Omega \)

- Specify \( p(\omega) \) for each outcome \( \omega \in \Omega \) by a big table:

<table>
<thead>
<tr>
<th>( x_1 )</th>
<th>( x_2 )</th>
<th>( x_3 )</th>
<th>( p(x_1, x_2, x_3) )</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>.11</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
<td>.02</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>.05</td>
</tr>
</tbody>
</table>

- How many parameters do we need to specify?

\[ 2^3 - 1 \]
Marginalization

- Suppose $X$ and $Y$ are random variables with distribution $p(X, Y)$
  - $X$: Intelligence, $\text{Val}(X) = \{\text{"Very High"}, \text{"High"}\}$
  - $Y$: Grade, $\text{Val}(Y) = \{\text{"a"}, \text{"b"}\}$

- Joint distribution specified by:

<table>
<thead>
<tr>
<th></th>
<th>vh</th>
<th>h</th>
</tr>
</thead>
<tbody>
<tr>
<td>Y</td>
<td>a</td>
<td>0.7</td>
</tr>
<tr>
<td></td>
<td>b</td>
<td>0.1</td>
</tr>
</tbody>
</table>

- $p(Y = a) = ? = 0.85$

- More generally, suppose we have a joint distribution $p(X_1, \ldots, X_n)$. Then,

$$p(X_i = x_i) = \sum_{x_1} \sum_{x_2} \cdots \sum_{x_{i-1}} \sum_{x_{i+1}} \cdots \sum_{x_n} p(x_1, \ldots, x_n)$$
Suppose $X$ and $Y$ are random variables with distribution $p(X, Y)$

- $X$: Intelligence, $\text{Val}(X) = \{ \text{"Very High"}, \text{"High"} \}$
- $Y$: Grade, $\text{Val}(Y) = \{ \text{"a"}, \text{"b"} \}$

\[
\begin{array}{c|cc}
   & X & h \\
--- & --- & --- \\
Y & a & 0.7 & 0.15 \\
   & b & 0.1 & 0.05 \\
\end{array}
\]

Can compute the conditional probability

\[
p(Y = a \mid X = \text{vh}) = \frac{p(Y = a, X = \text{vh})}{p(X = \text{vh})} = \frac{p(Y = a, X = \text{vh})}{p(Y = a, X = \text{vh}) + p(Y = b, X = \text{vh})} = \frac{0.7}{0.7 + 0.1} = 0.875.
\]
Example: Medical diagnosis

- Variable for each **symptom** (e.g. “fever”, “cough”, “fast breathing”, “shaking”, “nausea”, “vomiting”)
- Variable for each **disease** (e.g. “pneumonia”, “flu”, “common cold”, “bronchitis”, “tuberculosis”)
- Diagnosis is performed by **inference** in the model:

  \[ p(\text{pneumonia} = 1 \mid \text{cough} = 1, \text{fever} = 1, \text{vomiting} = 0) \]

- One famous model, Quick Medical Reference (QMR-DT), has 600 diseases and 4000 findings
Naively, could represent multivariate distributions with table of probabilities for each outcome (assignment).

How many outcomes are there in QMR-DT? \(2^{4600}\)

Estimation of joint distribution would require a huge amount of data.

Inference of conditional probabilities, e.g.

\[
p(\text{pneumonia} = 1 \mid \text{cough} = 1, \text{fever} = 1, \text{vomiting} = 0)
\]

would require summing over exponentially many variables’ values.

Moreover, defeats the purpose of probabilistic modeling, which is to make predictions with previously unseen observations.
Structure through independence

- If $X_1, \ldots, X_n$ are independent, then
  \[ p(x_1, \ldots, x_n) = p(x_1)p(x_2) \cdots p(x_n) \]

- $2^n$ entries can be described by just $n$ numbers (if $|\text{Val}(X_i)| = 2$)!

- However, this is not a very useful model – observing a variable $X_i$ cannot influence our predictions of $X_j$

- If $X_1, \ldots, X_n$ are conditionally independent given $Y$, denoted as $X_i \perp X_{-i} \mid Y$, then
  \[ p(y, x_1, \ldots, x_n) = p(y)p(x_1 \mid y) \prod_{i=2}^{n} p(x_i \mid x_1, \ldots, x_{i-1}, y) \]
  \[ = p(y)p(x_1 \mid y) \prod_{i=2}^{n} p(x_i \mid y). \]

- This is a simple, yet powerful, model
Example: naive Bayes for classification

- Classify e-mails as spam ($Y = 1$) or not spam ($Y = 0$)
  - Let $1 : n$ index the words in our vocabulary (e.g., English)
  - $X_i = 1$ if word $i$ appears in an e-mail, and 0 otherwise
  - E-mails are drawn according to some distribution $p(Y, X_1, \ldots, X_n)$
- Suppose that the words are conditionally independent given $Y$. Then,

\[
p(y, x_1, \ldots x_n) = p(y) \prod_{i=1}^{n} p(x_i \mid y)
\]

Estimate the model with maximum likelihood. Predict with:

\[
p(Y = 1 \mid x_1, \ldots x_n) = \frac{p(Y = 1) \prod_{i=1}^{n} p(x_i \mid Y = 1)}{\sum_{y=\{0,1\}} p(Y = y) \prod_{i=1}^{n} p(x_i \mid Y = y)}
\]

- Are the independence assumptions made here reasonable?
- Philosophy: Nearly all probabilistic models are “wrong”, but many are nonetheless useful
A Bayesian network is specified by a directed acyclic graph $G = (V, E)$ with:

1. One node $i \in V$ for each random variable $X_i$
2. One conditional probability distribution (CPD) per node, $p(x_i \mid x_{Pa(i)})$, specifying the variable’s probability conditioned on its parents’ values

Corresponds 1-1 with a particular factorization of the joint distribution:

$$p(x_1, \ldots, x_n) = \prod_{i \in V} p(x_i \mid x_{Pa(i)})$$

Powerful framework for designing algorithms to perform probability computations
Consider the following Bayesian network:

$$p(x_1, \ldots, x_n) = \prod_{i \in V} p(x_i \mid x_{\text{Pa}(i)})$$

$$p(d, i, g, s, l) = p(d)p(i)p(g \mid i, d)p(s \mid i)p(l \mid g)$$

What is its joint distribution?
More examples

- Evidence is denoted by shading in a node
- Can interpret Bayesian network as a **generative process**. For example, to *generate* an e-mail, we
  1. Decide whether it is spam or not spam, by sampling $y \sim p(Y)$
  2. For each word $i = 1$ to $n$, sample $x_i \sim p(X_i \mid Y = y)$
Bayesian network structure implies conditional independencies!

The joint distribution corresponding to the above BN factors as

\[ p(d, i, g, s, l) = p(d)p(i)p(g \mid i, d)p(s \mid i)p(l \mid g) \]

However, by the chain rule, any distribution can be written as

\[ p(d, i, g, s, l) = p(d)p(i \mid d)p(g \mid i, d)p(s \mid i, d, g)p(l \mid g, d, i, g, s) \]

Thus, we are assuming the following additional independencies:

\[ D \indep I, \quad S \indep \{D, G\} \mid I, \quad L \indep \{I, D, S\} \mid G. \quad \text{What else?} \]
Bayesian network structure implies conditional independencies!

- Generalizing the above arguments, we obtain that a variable is independent from its non-descendants given its parents.

- **Common parent** – fixing B *decouples* A and C.

- **Cascade** – knowing B *decouples* A and C.

- **V-structure** – Knowing C *couples* A and B.
  - This important phenomenon is called *explaining away* and is what makes Bayesian networks so powerful.
We’ll show that $p(A, C \mid B) = p(A \mid B)p(C \mid B)$ for any distribution $p(A, B, C)$ that factors according to this graph structure, i.e.

$$p(A, B, C) = p(B)p(A \mid B)p(C \mid B)$$

Proof.

$$p(A, C \mid B) = \frac{p(A, B, C)}{p(B)} = p(A \mid B)p(C \mid B)$$
D-separation ("directed separated") in Bayesian networks

- Algorithm to calculate whether $X \perp Z \mid Y$ by looking at graph separation
- Look to see if there is active path between $X$ and $Y$ when variables $Y$ are observed:

(a)\[ X \rightarrow Y \rightarrow Z \]
(b)\[ X \rightarrow Y \leftarrow Z \]

\[ X \rightarrow Y \rightarrow Z \]
\[ X \rightarrow Y \leftarrow Z \]
Algorithm to calculate whether $X \perp Z \mid Y$ by looking at graph separation

Look to see if there is active path between $X$ and $Y$ when variables $Y$ are observed:

(a)

(b)
D-separation ("directed separated") in Bayesian networks

- Algorithm to calculate whether $X \perp Z \mid Y$ by looking at graph separation
- Look to see if there is **active path** between $X$ and $Y$ when variables $Y$ are observed:

![Diagram](image)

- If no such path, then $X$ and $Z$ are **d-separated** with respect to $Y$
- d-separation reduces statistical independencies (hard) to connectivity in graphs (easy)
- Important because it allows us to quickly prune the Bayesian network, finding just the relevant variables for answering a query
D-separation example 1
D-separation example 2
Bayesian networks given by \((G, P)\) where \(P\) is specified as a set of local conditional probability distributions associated with \(G\)’s nodes

One interpretation of a BN is as a generative model, where variables are sampled in topological order

Local and global independence properties identifiable via d-separation criteria

Computing the probability of any assignment is obtained by multiplying CPDs

- Bayes’ rule is used to compute conditional probabilities
- Marginalization or inference is often computationally difficult