Bayesian network. Graphical models

Virgil Pavlu    September 30, 2008

1 Introduction to probabilities, statistics

- random variables $X, Y$ and density functions
- conditional probability $P[X|Y]
- joint probability $P[X, Y] = P[X|Y] \cdot P[Y]
- Bayes rule $P[X|Y] \cdot P[Y] = P[Y|X] \cdot P[X]
- independence $P[X, Y] = P[X] \cdot P[Y]
- marginalization $P[X] = \sum_{Y=y} P[X|Y = y] \cdot P[Y = y]

<table>
<thead>
<tr>
<th></th>
<th>red</th>
<th>blue</th>
<th>green</th>
</tr>
</thead>
<tbody>
<tr>
<td>square</td>
<td>0.25</td>
<td>0.10</td>
<td>0.21</td>
</tr>
<tr>
<td>round</td>
<td>0.17</td>
<td>0.04</td>
<td>0.23</td>
</tr>
<tr>
<td></td>
<td>0.42</td>
<td>0.14</td>
<td>0.44</td>
</tr>
</tbody>
</table>

Figure 1: probabilities
2 Bayesian networks. Inference example

Also called Belief networks or Probabilistic networks.

\[
\begin{align*}
p(G = 1|B = 1, F = 1) &= 0.8 \\
p(G = 1|B = 1, F = 0) &= 0.2 \\
p(G = 1|B = 0, F = 1) &= 0.2 \\
p(G = 1|B = 0, F = 0) &= 0.1 \\
p(B = 1) &= 0.9 \\
p(F = 1) &= 0.9 \\
\text{and hence} \\
p(F = 0) &= 0.1 \\
\end{align*}
\]

\(B = \text{Battery (0=flat, 1=fully charged)}\)

\(F = \text{Fuel Tank (0=empty, 1=full)}\)

\(G = \text{Fuel Gauge Reading (0=empty, 1=full)}\)

Figure 2: bayesian computation

\[
\begin{align*}
p(F = 0|G = 0) &= \frac{p(G = 0|F = 0)p(F = 0)}{p(G = 0)} \\
&\approx 0.257
\end{align*}
\]

Probability of an empty tank increased by observing \(G = 0\).

Figure 3: bayesian computation

\[
\begin{align*}
p(F = 0|G = 0, B = 0) &= \frac{p(G = 0|B = 0, F = 0)p(F = 0)}{\sum_{F \in \{0, 1\}} p(G = 0|B = 0, F)p(F)} \\
&\approx 0.111
\end{align*}
\]

Probability of an empty tank reduced by observing \(B = 0\).

This referred to as "explaining away".

Figure 4: bayesian computation
3 Bayesian networks. Factorization

\[ p(x_1, \ldots, x_7) = p(x_1) p(x_2) p(x_3) p(x_4|x_1, x_2, x_3) \]
\[ p(x_5|x_1, x_3) p(x_6|x_4) p(x_7|x_4, x_5) \]

General Factorization

\[ p(x) = \prod_{k=1}^{K} p(x_k|pa_k) \]

Figure 5: bayesian network factorization

\[ \bullet \mathbf{P}[A, B, C, D, E] = \]
\[ \mathbf{P}[A] \cdot \mathbf{P}[B|A] \cdot \mathbf{P}[C|A] \cdot \mathbf{P}[D|B, C] \cdot \mathbf{P}[E|D] \]

\[ \bullet \mathbf{P}[A, B, C, D, E] = \]
\[ \mathbf{P}[A] \cdot \mathbf{P}[B] \cdot \mathbf{P}[C|A, B] \cdot \mathbf{P}[D|C] \cdot \mathbf{P}[E|C] \]

Figure 6: bayesian network factorization
4 Complexity of generative models

• cut links or assume independence of components

\[
p(x_1, x_2 | \mu) = \prod_{k=1}^{K} \prod_{l=1}^{K} \mu_{kl}^{x_{1k} x_{2l}}
\]

**General joint distribution:** \( K^2 - 1 \) parameters

\[
p(x_1, x_2 | \mu) = \prod_{k=1}^{K} \mu_{1k} x_{1k} \prod_{l=1}^{K} \mu_{2l} x_{2l}
\]

**Independent joint distribution:** \( 2(K - 1) \) parameters

Figure 7: reduce parameters

• share parameters

\[
p(x_1, x_2, \ldots, x_m | \mu_1, \mu) = p(x_1 | \mu_1) p(\mu_1) \prod_{m=2}^{M} p(x_m | x_{m-1}, \mu_1) p(\mu)
\]

**Shared prior**

Figure 8: share parameters

• use parametrized models for conditional distribution instead of tables

If \( x_1, \ldots, x_M \) are discrete, \( K \)-state variables, \( p(y = 1 | x_1, \ldots, x_M) \) in general has \( O(K^M) \) parameters.

The parameterized form

\[
p(y = 1 | x_1, \ldots, x_M) = \sigma \left( w_0 + \sum_{i=1}^{M} w_i x_i \right) = \sigma(w^T x)
\]

requires only \( M + 1 \) parameters

Figure 9: parametrize conditional
5 Conditional independence

\[ p(a, b, c) = p(a | c)p(b | c)p(c) \]

\[ p(a, b) = \sum_c p(a | c)p(b | c)p(c) \]

\[ a \perp b \mid \emptyset \]

\[ p(a, b | c) = \frac{p(a, b, c)}{p(c)} = \frac{p(a, c)p(b | c)}{p(c)} \]

\[ a \perp b \mid c \]

Figure 10: conditional independence

\[ a \perp b \mid c \]

\[ a \perp b \mid f \]

Figure 11: conditional independence
6 Inference in graphical models

\[ p(y) = \sum_{x'} p(y|x') p(x') \quad p(x|y) = \frac{p(y|x)p(x)}{p(y)} \]

Figure 12: graphical inference

\[
p(x_n) = \frac{1}{Z} \left[ \sum_{x_{n-1}} \psi_{n-1,n}(x_{n-1}, x_n) \cdots \left[ \sum_{x_1} \psi_{1,2}(x_1, x_2) \right] \cdots \right] - \mu_\alpha(x_n) \\
\left[ \sum_{x_{n+1}} \psi_{n,n+1}(x_n, x_{n+1}) \cdots \left[ \sum_{x_N} \psi_{N-1,N}(x_{N-1}, x_N) \right] \cdots \right] - \mu_\beta(x_n)
\]

Figure 13: graphical inference
7 Factor graphs

\[ p(\mathbf{x}) = f_a(x_1, x_2)f_b(x_1, x_2)f_c(x_2, x_3)f_d(x_3) \]

\[ p(\mathbf{x}) = \prod_s f_s(\mathbf{x}_s) \]

Figure 14: Factor graph

Figure 15: Factor graph
8 Belief propagation: sum-product algorithm

\[ p(x) = \sum_{x \setminus x} p(x) \]
\[ p(x) = \prod_{s \in \text{ne}(x)} F_s(x, X_s) \]
\[ p(x) = \prod_{s \in \text{ne}(x)} \left[ \sum_{X_s} F_s(x, X_s) \right] = \prod_{s \in \text{ne}(x)} \mu_{f_s \to x}(x), \quad \mu_{f_s \to x}(x) = \sum_{X_s} F_s(x, X_s) \]

Figure 16: Sum-product algorithm

\[ F_s(x, X_s) = f_s(x, x_1, \ldots, x_M) G_1(x_1, X_{s1}) \cdots G_M(x_M, X_{sM}) \]

\[ \mu_{f_s \to x}(x) = \sum_{x_1} \cdots \sum_{x_M} f_s(x, x_1, \ldots, x_M) \prod_{s \in \text{in}(f_s) \setminus \{s\}} \sum_{X_s} G_m(x_m, X_m) \]

Figure 17: Sum-product algorithm

\[ \mu_{x \to f_s}(x_M) = \sum_{X_m} G_m(x_m, X_m) = \sum_{X_m} \prod_{l \in \text{in}(x_m) \setminus f_s} F_l(x_m, X_{ml}) \]
\[ = \prod_{l \in \text{in}(x_m) \setminus f_s} \mu_{f_l \to x_m}(x_m) \]

Figure 18: Sum-product algorithm
Let’s look at an example.

Figure 19: Sum-product example

\[
\bar{p}(x) = f_a(x_1, x_2)f_b(x_2, x_3)f_c(x_2, x_4)
\]

\[
\mu_{x_1 \rightarrow f_a}(x_1) = 1
\]
\[
\mu_{f_a \rightarrow x_2}(x_2) = \sum_{x_3} f_b(x_2, x_3)
\]
\[
\mu_{x_2 \rightarrow f_a}(x_2) = \mu_{f_a \rightarrow x_2}(x_2)\mu_{f_a \rightarrow x_3}(x_2)
\]
\[
\mu_{f_a \rightarrow x_1}(x_1) = \sum_{x_2} f_a(x_1, x_2)\mu_{x_2 \rightarrow f_a}(x_2)
\]
\[
\mu_{x_2 \rightarrow x_3}(x_2) = \mu_{f_a \rightarrow x_3}(x_2)\mu_{f_a \rightarrow x_2}(x_2)
\]
\[
\mu_{f_a \rightarrow x_4}(x_4) = \sum_{x_2} f_a(x_2, x_4)\mu_{x_2 \rightarrow f_a}(x_2)
\]

Figure 20: Sum-product example

\[
\bar{p}(x_2) = \mu_{f_a \rightarrow x_2}(x_2)\mu_{f_b \rightarrow x_2}(x_2)\mu_{f_c \rightarrow x_2}(x_2)
\]
\[
= \left[ \sum_{x_1} f_a(x_1, x_2) \right] \left[ \sum_{x_3} f_b(x_2, x_3) \right]
\]
\[
= \sum_{x_1} \sum_{x_2} \sum_{x_3} f_a(x_1, x_2) f_b(x_2, x_3) f_c(x_2, x_4)
\]
9   Max-sum algorithm
10  Junction trees. Loopy belief propagation