Factored joint distribution - Preview

\[ P(F, A, S, H, N) = P(F) \cdot P(A) \cdot P(S|F, A) \cdot P(H|S) \cdot P(N|H) \]

\[ 2^5 - 1 = 31 \text{ possible states} \]
What about probabilities?

Conditional probability tables (CPTs)

Key: Independence assumptions

Knowing sinus separates the variables from each other
The independence assumption

Local Markov Assumption:
A variable $X$ is independent of its non-descendants given its parents

Explaining away

Local Markov Assumption:
A variable $X$ is independent of its non-descendants given its parents

Flu $\rightarrow$ Allergy $\rightarrow$ Sinus
Flu $\rightarrow$ Headache
Flu $\rightarrow$ Nose

Flu $\rightarrow$ Allergy $\rightarrow$ Sinus
Flu $\rightarrow$ Headache
Flu $\rightarrow$ Nose

Flu Allergy Sinus Headache Nose

$F$ $A$ $S$ $H$ $N$

Notation:
- $F$: Flu
- $A$: Allergy
- $S$: Sinus
- $H$: Headache
- $N$: Nose

P(F=t | S=t) ≠ P(F=t | S=t, A=t)
Naïve Bayes revisited

Local Markov Assumption:
A variable X is independent of its non-descendants given its parents

Joint distribution

Why can we decompose? Markov Assumption!
The chain rule of probabilities

- $P(A, B) = P(A)P(B|A) = P(B)P(A|B)$
  
  For any dist
  
  $P(F, S) = P(F)P(S|F)$
  
  $P(F, A, S) = P(F)P(A|F)P(S|F, A)$

- More generally:
  
  $P(X_1, \ldots, X_n) = P(X_1)P(X_2|X_1)\ldots P(X_n|X_1, \ldots, X_{n-1})$

---

Chain rule & Joint distribution

**Local Markov Assumption:** A variable $X$ is independent of its non-descendants given its parents.

Proof by example: If $P(FASH) = P(F)P(A|F)P(S|FA)P(H|FAS)P(N|FASH)$, then

- $P(F)P(A|F)$
- $P(S|FA)$
- $P(H|FAS)$
- $P(N|FASH)$

Order matters? $P(FASH) = P(F)P(A|F)P(S|FA)P(N|FASH)$

Would not get $P(F)$ if $X$ depends on $F$ and $H$.

Follow typographical order:

- $A \perp F \iff P(A|F) = P(A)$
- $H \perp FASH \iff P(H|FAS) = P(H|S)$
- $N \perp FASH \iff P(N|FASH) = P(N)$
The Representation Theorem – Joint Distribution to BN

**BN:**

**Joint probability distribution:**

\[
P(X_1, \ldots, X_n) = \prod_{i=1}^{n} P(X_i | \text{Pa}_X_i)
\]

If conditional independencies in BN are subset of conditional independencies in \( P \)

Obtain

Encodes independence assumptions

Two (trivial) special cases

**Edgeless graph**

- \( X_1, X_2, X_3, \ldots \)
- \( X_1 \indep \text{all others} \)

- Finest param
- High bias

**Fully-connected graph**

- \( X_1, X_2, X_3, X_4 \)
- No independence

- Structure learning
- Most param
- High variance
Review

- Bayesian Networks
  - Compact representation for probability distributions
  - Exponential reduction in number of parameters
- Fast probabilistic inference
  - As shown in demo examples
  - Compute $P(X|e)$
- Today
  - Learn BN structure

Flu → Allergy
Sinus
Headache → Nose
Learning Bayes nets

Data \(x^{(1)}\) … \(x^{(m)}\)

\[
\text{structure} \quad \text{parameters}
\]

\[
\text{MLE} \quad P(D \mid G, \Theta_c)
\]

Learning the CPTs

For each discrete variable \(X_i\)

\[
P(S = s \mid A = a, F = f) = \frac{\text{MLE} \ \text{Count}(s = s, a = a, f = f)}{\text{Count}(a = a, f = f)}
\]

\[
P(X_c = x_c \mid \text{Pa}_X = w) = \frac{\text{MLE} \ \text{Count}(x_c = x_c, \text{Pa}_X = w)}{\text{Count}(\text{Pa}_X = w)}
\]

Substitution:

\[
\text{Count}(\text{Pa}_X = w) = 0 \text{ or very small}
\]

→ add smoothing / L2 regularization / AKA Bayesian priors

MLE:

\[
P(X_i = x_i \mid X_j = x_j) = \frac{\text{Count}(X_i = x_i, X_j = x_j)}{\text{Count}(X_j = x_j)}
\]
Information-theoretic interpretation of maximum likelihood 1

Given structure, log likelihood of data:

\[
\log P(\mathcal{D} | \theta_G, G) = \log \prod_{j=1}^{m} \prod_{i=1}^{N} P(x_{i,j}^{(j)} | \text{Pa}_{x_{i,j}} = \mathcal{U}_{i,j}^{(j)})
\]

\[
= \sum_{i=1}^{N} \sum_{j=1}^{m} \log P(x_{i,j}^{(j)} | \text{Pa}_{x_{i,j}} = \mathcal{U}_{i,j}^{(j)})
\]

\[
= \sum_{i=1}^{N} \sum_{x_{i,j} \in \mathcal{X}_{i,j}} \sum_{\text{Pa}_{x_{i,j}} \in \mathcal{U}_{i,j}} \log \frac{P(x_{i,j}^{(j)} | \text{Pa}_{x_{i,j}} = \mathcal{U}_{i,j}^{(j)})}{\sum_{x_{i,j}' \in \mathcal{X}_{i,j}} P(x_{i,j}'^{(j)} | \text{Pa}_{x_{i,j}} = \mathcal{U}_{i,j}^{(j)})}
\]

\[
= \sum_{i=1}^{N} \sum_{x_{i,j} \in \mathcal{X}_{i,j}} \sum_{\text{Pa}_{x_{i,j}} \in \mathcal{U}_{i,j}} \log \frac{P(x_{i,j}^{(j)} | \text{Pa}_{x_{i,j}} = \mathcal{U}_{i,j}^{(j)})}{\sum_{x_{i,j}' \in \mathcal{X}_{i,j}} P(x_{i,j}'^{(j)} | \text{Pa}_{x_{i,j}} = \mathcal{U}_{i,j}^{(j)})}
\]

\[
= \sum_{i=1}^{N} \sum_{x_{i,j} \in \mathcal{X}_{i,j}} \sum_{\text{Pa}_{x_{i,j}} \in \mathcal{U}_{i,j}} \log \frac{P(x_{i,j}^{(j)} | \text{Pa}_{x_{i,j}} = \mathcal{U}_{i,j}^{(j)})}{\sum_{x_{i,j}' \in \mathcal{X}_{i,j}} P(x_{i,j}'^{(j)} | \text{Pa}_{x_{i,j}} = \mathcal{U}_{i,j}^{(j)})}
\]

\[
- \sum_{i=1}^{N} \mathcal{H}(X_i | \text{Pa}_{x_{i,j}})
\]

Information-theoretic interpretation of maximum likelihood 2

Given structure, log likelihood of data:

\[
\log P(\mathcal{D} | \theta_G, G) = \sum_{j=1}^{m} \sum_{i=1}^{N} \log P(x_{i,j}^{(j)} | \text{Pa}_{x_{i,j}} = \mathcal{U}_{i,j}^{(j)})
\]

\[
= \sum_{i=1}^{N} \sum_{x_{i,j} \in \mathcal{X}_{i,j}} \sum_{\text{Pa}_{x_{i,j}} \in \mathcal{U}_{i,j}} \log \frac{P(x_{i,j}^{(j)} | \text{Pa}_{x_{i,j}} = \mathcal{U}_{i,j}^{(j)})}{\sum_{x_{i,j}' \in \mathcal{X}_{i,j}} P(x_{i,j}'^{(j)} | \text{Pa}_{x_{i,j}} = \mathcal{U}_{i,j}^{(j)})}
\]

\[
= \sum_{i=1}^{N} \sum_{x_{i,j} \in \mathcal{X}_{i,j}} \sum_{\text{Pa}_{x_{i,j}} \in \mathcal{U}_{i,j}} \log \frac{P(x_{i,j}^{(j)} | \text{Pa}_{x_{i,j}} = \mathcal{U}_{i,j}^{(j)})}{\sum_{x_{i,j}' \in \mathcal{X}_{i,j}} P(x_{i,j}'^{(j)} | \text{Pa}_{x_{i,j}} = \mathcal{U}_{i,j}^{(j)})}
\]

\[
= \sum_{i=1}^{N} \sum_{x_{i,j} \in \mathcal{X}_{i,j}} \sum_{\text{Pa}_{x_{i,j}} \in \mathcal{U}_{i,j}} \log \frac{P(x_{i,j}^{(j)} | \text{Pa}_{x_{i,j}} = \mathcal{U}_{i,j}^{(j)})}{\sum_{x_{i,j}' \in \mathcal{X}_{i,j}} P(x_{i,j}'^{(j)} | \text{Pa}_{x_{i,j}} = \mathcal{U}_{i,j}^{(j)})}
\]

\[
- \sum_{i=1}^{N} \mathcal{H}(X_i | \text{Pa}_{x_{i,j}})
\]
Information-theoretic interpretation of maximum likelihood 3

- Given structure, log likelihood of data:

\[
\max_G \log \hat{P}(D \mid \theta, G) = m \sum_{i} \sum_{x_i, \text{pa}_G} \hat{P}(x_i, \text{pa}_G) \log \hat{P}(x_i \mid \text{pa}_G)
\]

\[
\leq \max_G -m \sum_{i} \sum_{x_i} H(x_i) \leq \min_{G} m \sum_{i} H(x_i)
\]

\[
\leq \max_G -m \sum_{i} \sum_{x_i} \mathcal{I}(x_i; \text{pa}_G) - m \sum_{i} H(x_i)
\]

*Information Theoretic interpretation does not depend on G of MLE for \( \theta \)*

\[
\Rightarrow \max_G \equiv \text{choosing parents with max mutual info with var}
\]

Decomposable score

- Log data likelihood

\[
\log \hat{P}(D \mid \theta, G) = m \sum_i \mathcal{I}(X_i, \text{pa}_G) - m \sum_i H(X_i)
\]

- Decomposable score:
  - Decomposes over families in BN (node and its parents)
  - Will lead to significant computational efficiency!!!
  - Score(\( G \mid D \)) = \( \sum_{X_i} \mathcal{I}(X_i; \text{pa}_G) \)

\[
\sum_{X_i} \mathcal{I}(X_i; \text{pa}_G)
\]
How many trees are there?

Nonetheless – Efficient optimal algorithm finds best tree

Every var has at most one parent

For n vars, how many possible trees?

O(n \log n)

Exhaustive search is impossible

Scoring a tree 1: equivalent trees

\[ \log P(D \mid \theta, G) = m \sum_i \hat{I}(X_i, Pa_{X_i}, G) - m \sum_i \hat{H}(X_i) = \max_{E} \hat{I}(E_{\theta}, E_{G}) \]

No true for all edge directions:

A ↓ B ↣ ALC ∣ \emptyset

Not a tree, because B not a parent.
**Scoring a tree 2: similar trees**

\[
\log \hat{P}(D | \theta, G) = m \sum_i \hat{I}(X_i, \text{Pa}_X, G) - m \sum_i H(X_i)
\]

For each pair of variables \(X_i, X_j\):

- Compute empirical distribution:
  \[
  \hat{P}(x_i, x_j) = \frac{\text{Count}(x_i, x_j)}{m}
  \]
- Compute mutual information:
  \[
  \hat{I}(X_i, X_j) = \sum_{x_i, x_j} \hat{P}(x_i, x_j) \log \frac{\hat{P}(x_i, x_j)}{\hat{P}(x_i)\hat{P}(x_j)}
  \]

Define a graph:

- Nodes \(X_1, \ldots, X_n\)
- Edge \((i, j)\) gets weight \(\hat{I}(X_i, X_j)\)

A
down
B
down
C

Score = \(\hat{I}(A, B) + \hat{I}(B, C)\)

\[\text{Score} = \hat{I}(A, B) + \hat{I}(A, C)\]

**Chow-Liu tree learning algorithm 1**

- For each pair of variables \(X_i, X_j\):
  - Compute empirical distribution:
    \[
    \hat{P}(x_i, x_j) = \frac{\text{Count}(x_i, x_j)}{m}
    \]
  - Compute mutual information:
    \[
    \hat{I}(X_i, X_j) = \sum_{x_i, x_j} \hat{P}(x_i, x_j) \log \frac{\hat{P}(x_i, x_j)}{\hat{P}(x_i)\hat{P}(x_j)}
    \]
- Define a graph:
  - Nodes \(X_1, \ldots, X_n\)
  - Edge \((i, j)\) gets weight \(\hat{I}(X_i, X_j)\)

Run max spanning tree — complexity is about \(O(E \log E)\) or \(O(n^2 \log n)\).
Chow-Liu tree learning algorithm 2

\[ \log \hat{P}(D \mid \theta, \mathcal{G}) = m \sum_i I(X_i, \text{Pa}_X, G) - m \sum_i \hat{H}(X_i) \]

Optimal tree BN

- Compute maximum weight spanning tree
- Directions in BN: pick any node as root, breadth-first-search defines directions