CSE 150 Lecture 8 2.1.10.

Review

* Learning in BNs
* Maximum likelihood (ML) estimation

  Estimate CPTs that maximize probability of observed data (evidence)
* Complete data (a.k.a. fully observed)

  evidence $\{x_1^{(t)}, x_2^{(t)}, \ldots, x_n^{(t)}\}_{t=1}^{T}$ T complete instantiations of nodes $X_1, X_2, \ldots, X_n$

* ML estimates:

$$P_{ML}(X_i = x \mid pa_i = \pi) = \frac{\text{Count}(X_i = x, pa_i = \pi)}{\sum_{x'} \text{Count}(X_i = x', pa_i = \pi)}$$

Equivalently:

$$P_{ML}(X_i = x \mid pa_i = \pi) = \frac{\text{Count}(X_i = x, pa_i = \pi)}{\text{Count}(pa_i = \pi)}$$

* Other notation:

  Indicator function $I(x, x') = \begin{cases} 1 & \text{if } x = x' \\ 0 & \text{otherwise} \end{cases}$

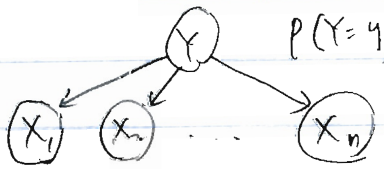
  $\text{Count}(X_i = x, pa_i = \pi) = \sum_{t=1}^{T} I(x_i^{(t)}, x)\, I(pa_i^{(t)}, \pi)$

Ex: naive Bayes model for document classification

* Variables

  $y \in \{1, 2, \ldots, m\}$ possible document topics

  $X_i \in \{0, 1\}$ does ith word in vocabulary (dictionary) appear in documen
* BN = DAG + CPTs



$P(Y = y)$

$P(X_i = 1 \mid Y = y)$ for $i = 1 \ldots n$

* Document classification

$$P(Y = y \mid \vec{X} = \vec{x}) = \frac{P(\vec{X} = \vec{x} \mid Y = y)\, P(Y = y)}{P(\vec{X} = \vec{x})}$$ Bayes rule

$$P(Y=y \mid \vec{X}=\vec{x}) = \frac{\left\{\prod_{i=1}^{n} P(X_i=x_i \mid Y=y)\right\} P(Y=y)}{\sum_{y'} \left\{\prod_{i=1}^{n} P(X_i=x_i \mid Y=y')\right\} P(Y=y')}$$

Conditional independence

"Naive Bayes" assumption

* Strengths of model

(1) easy to estimate from a large corpus of documents

$P_{ML}(Y=y)$ fraction of documents w/ topic y

$P_{ML}(X_i=1 \mid Y=y)$ fraction of documents w/ topic y that contain i-th word in vocabulary.

(2) Simplest baseline

* Weaknesses of model

(1) Naive Bayes assumption that words appear independently given topic.

(2) "bag-of-words" representation ignores word ordering

$\boxed{\text{Ex: Markov models of language}}$

* Let w_ℓ denote word at ℓ-th position in sentence.

How to model

$P(w_1, w_2, \ldots, w_{L-1}, w_L)$ probability of sentence with L words $w_1, \ldots, w_L$

* Simplifying assumption

(1) finite context / memory

$$P(w_\ell \mid w_1, w_2, \ldots, w_{\ell-1}) = P(w_\ell \mid \underbrace{w_{\ell-(k-1)}, w_{\ell-(k-2)}, \ldots, w_{\ell-2}, w_{\ell-1}}_{(k-1)\text{ previous words}})$$

"k-gram" model

$$P(w_\ell \mid w_1, w_2, \ldots, w_{\ell-1}) = P(w_\ell \mid w_{\ell-1}) \quad \text{"bi-gram" model}$$

(2) position invariance

$$P(w_{\ell+1}=w' \mid w_\ell = w) = P(w_\ell = w' \mid w_{\ell-1} = w)$$

* Belief network for bigram model of language.

$$\omega_1 \to \omega_2 \to \omega_3 \to \cdots \to \omega_L$$

Same CPTs at all non-root nodes in BN.

* Learning bigram model
  * Collect large corpus of text $\sim 10^8$ words
  * Vocabulary size: $V \sim 10^5$ dictionary entries.
* Count C_{ij} = # times that word j follows word i

  Count C_i = # times that word i appears (followed by any word)

  estimate $P_{ML}(\omega_\ell = j \mid \omega_{\ell-1} = i) = \frac{C_{ij}}{C_i}$

* Note: no generalization to unseen word combinations.
* n-gram model: Condition on previous n words

$$P(\omega_\ell \mid \omega_1, \ldots, \omega_{\ell-1}) = P(\omega_\ell \mid \omega_{\ell-(n-1)}, \cdots, \omega_{\ell-1})$$

$n=1$ unigram
$n=2$ bigram
$n=3$ trigram

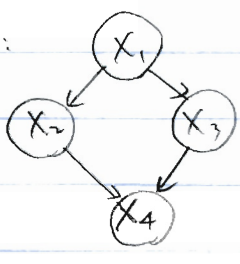
n-gram counts get increasingly sparse for large n.

ML esitimation from incomplete data

* Given: fixed DAG over discrete nodes $\{X_1, X_2, \ldots, X_n\}$

  Also: data set of T partial instantiations of $\{X_1, X_2, \ldots, X_n\}$

Ex:



t	X_1	X_2	X_3	X_4
1	0	?	1	0
2	1	?	?	1
3	0	?	0	?
⋮				
T	1	?	1	0

* Goal: estimate CPTs $P(X_i = x \mid Pa_i = \pi)$ that maximize <u>marginal</u> probability of <u>partially</u> observed data

* Variables in BN

$X =$ all nodes $\qquad X = H \cup V$

$H =$ hidden nodes

$V =$ visible nodes

* Log-likelihood

Assume that T examples are iid from joint distribution $P(X_1, X_2, \ldots, X_n)$:

visible nodes on t-th example

$$\begin{aligned}
\mathcal{L} &= \log P(\text{data}) \\
&= \log \left[\prod_{t=1}^{T} P(V = v^{(t)}) \right] \\
&= \sum_{t=1}^{T} \log P(V = v^{(t)}) \\
&= \sum_{t=1}^{T} \log \sum_{h} P(V = v^{(t)}, H = h) \quad \text{marginalizing over joint for } X = H \cup V \\
&= \sum_{t=1}^{T} \log \sum_{h} \prod_{i=1}^{n} P(X_i = x \mid pa_i = \pi) \Big|_{H=h,\, V=v^{(t)}}
\end{aligned}$$

* More complicated to optimize $\mathcal{L}$ for incomplete data

- No "closed form" solution.

  Alternative: iterative solution.

* Expectation-Maximization (EM) algorithm

iterative procedure to maximize $\mathcal{L}(\text{data})$ for incomplete data in terms of CPTs.

* Intuition - by analogy, ML estimates for complete data

$$P_{ML}(X_i = x \mid pa_i = \pi) = \frac{\text{Count}(X_i = x, pa_i = \pi)}{\text{Count}(pa_i = \pi)} = \frac{\sum_{t=1}^{T} I(x_i^{(t)}, x)\, I(pa_i^{(t)}, \pi)}{\sum_{t=1}^{T} I(pa_i^{(t)}, \pi)}$$

For incomplete data, we must "fill in" hidden values:

$$p_{ML}(X=x_i \mid pa_i=\pi) \leftarrow \frac{\sum_{t=1}^{T} p(X_i=x, pa_i=\pi \mid V=v^{(t)})}{\sum_{t=1}^{T} p(pa_i=\pi \mid V^{(t)})}$$

Intuition: expected statistics ("counts") under $p(H|V)$ substitute for observed counts in complete data case.