CS 2750 Machine Learning
Lecture 17

Density estimation with hidden variables and missing values

Milos Hauskrecht
milos@cs.pitt.edu
5329 Sennott Square

Administration

Midterm: Wednesday, March 17, 2004
- **In class**
- **Closed book**
- **Material covered by Spring break, excluding learning of the BBN structures**
- **Last year midterm is on the web**

No new homework

Project proposals

Due: Wednesday, March 24, 2004

- **1-2 pages long**

Proposal

- **Written proposal:**
  1. Outline of a learning problem, type of data you have available. Why is the problem important?
  2. Learning methods you plan to try and implement for the problem. References to previous work.
  3. How do you plan to test, compare learning approaches
  4. Schedule of work (approximate timeline of work)
- **A PPT (3 slide) summary of points 1-4**

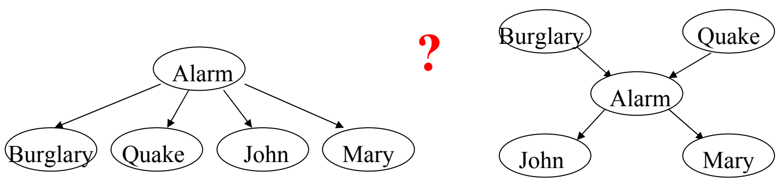
Learning the structure of the BBN

Model selection

- **BBN has two components**:
  - **Structure of the network** (models conditional independences)
  - **A set of parameters** (conditional child-parent distributions)

We already know how to learn the parameters for the fixed structure

But how to learn the structure of the BBN?



Learning the structure

Criteria we can choose to score the structure S

- **Marginal likelihood**

  maximize $P(D \mid S, \xi)$

  ξ - represents the prior knowledge

- **Maximum posterior probability**

  maximize $P(S \mid D, \xi)$

$$P(S \mid D, \xi) = \frac{P(D \mid S, \xi) P(S \mid \xi)}{P(D \mid \xi)}$$

How to compute marginal likelihood $P(D \mid S, \xi)$?

Learning of BBNs

- **Notation**:
  - i ranges over all possible variables $i=1,..,n$
  - $j=1,..,q$ ranges over all possible parent combinations
  - $k=1,..,r$ ranges over all possible variable values
  - $\mathbf{\Theta}$ - parameters of the BBN

  Θ_{ij} is a vector of Θ_{ijk} representing parameters of the conditional probability distribution; such that $\sum_{k=1}^{r} \Theta_{ijk} = 1$

  N_{ijk} - a number of instances in the dataset where parents of variable X_i take on values j and X_i has value k

  $N_{ij} = \sum_{k=1}^{r} N_{ijk}$

  α_{ijk} - prior counts (parameters of Beta and Dirichlet priors)

  $\alpha_{ij} = \sum_{k=1}^{r} \alpha_{ijk}$

Marginal likelihood

- Integrate over all possible parameter settings

$$P(D \mid S,\xi) = \int_{\Theta} P(D \mid S,\Theta,\xi)\, p(\Theta \mid S,\xi)\, d\Theta$$

- Using the assumption of parameter and sample independence

$$P(D \mid S,\xi) = \prod_{i=1}^{n} \prod_{j=1}^{q_i} \frac{\Gamma(\alpha_{ij})}{\Gamma(\alpha_{ij} + N_{ij})} \prod_{k=1}^{r_i} \frac{\Gamma(\alpha_{ijk} + N_{ijk})}{\Gamma(\alpha_{ijk})}$$

- We can use **log-likelihood score** instead

$$\log P(D \mid S,\xi) = \sum_{i=1}^{n} \left\{ \sum_{j=1}^{q_i} \log \frac{\Gamma(\alpha_{ij})}{\Gamma(\alpha_{ij} + N_{ij})} + \sum_{k=1}^{r_i} \log \frac{\Gamma(\alpha_{ijk} + N_{ijk})}{\Gamma(\alpha_{ijk})} \right\}$$

Score is decomposable along variables !!!

Computing the marginal likelihood

- **From the iid assumption:**

$$P(\boldsymbol{D} \mid S, \boldsymbol{\Theta}) = \prod_{h=1}^{N} \prod_{i=1}^{n} P(x_i^h \mid parents_i^h, \boldsymbol{\Theta})$$

- Let r_i = number of values that attribute x_i can take
  q_i= number of possible parent combinations
  N_{ijk}= number of cases in D where x_i has value k
  and parents with values j.

$$= \prod_{i}^{n} \prod_{j}^{q_i} \prod_{k}^{r_i} P(x_i = k \mid parents_i = j, \boldsymbol{\Theta})^{N_{ijk}}$$

$$= \prod_{i}^{n} \prod_{j}^{q_i} \prod_{k}^{r_i} \theta_{ijk}^{N_{ijk}}$$

Computing the marginal likelihood

- **From parameter independence**

$$p(\boldsymbol{\Theta} \mid S, \xi) = \prod_{i=1}^{n} \prod_{j=1}^{q_i} p(\Theta_{ij} \mid S, \xi)$$

- **Priors for** $p(\Theta_{ij} \mid S, \xi)$
  - $\Theta_{ij} = (\Theta_{ij1}, \ldots, \Theta_{ijr_i})$ is a vector of parameters;
  - we use a Dirichlet distribution with parameters $\boldsymbol{\alpha}$ to represent it

$$P(\Theta_{ij} \mid S, \xi) = P(\Theta_{ij1}, \ldots, \Theta_{ijr_i} \mid S, \xi) = Dirichlet(\Theta_{ij1}, \ldots, \Theta_{ijr_i} \mid \alpha)$$

$$= \frac{\Gamma(\sum_{k=1}^{r_i} \alpha_{ijk})}{\prod_{k=1}^{r_i} \Gamma(\alpha_{ijk})} \prod_{k=1}^{r_i} \Theta_{ijk}^{\alpha_{ijk}-1}$$

Computing the marginal likelihood

- **Combine things together:**

$$P(D \mid S_i) = \int_{\Theta} P(D \mid S_i, \Theta) P(\Theta \mid S_i) d\Theta$$

$$= \int \prod_{i}^{n} \prod_{j}^{q_i} \prod_{k}^{r_i} \Theta_{ijk}^{N_{ijk}} \cdot \frac{\Gamma(\sum_{k=1}^{r_i} \alpha_{ijk})}{\prod_{k=1}^{r_i} \Gamma(\alpha_{ijk})} \prod_{k=1}^{r_i} \Theta_{ijk}^{\alpha_{ijk}-1} d\Theta$$

$$= \prod_{i}^{n} \prod_{j}^{q_i} \frac{\Gamma(\sum_{k=1}^{r_i} \alpha_{ijk})}{\prod_{k=1}^{r_i} \Gamma(\alpha_{ijk})} \int \prod_{k=1}^{r_i} \Theta_{ijk}^{N_{ijk}+\alpha_{ijk}-1} d\Theta$$

$$= \prod_{i}^{n} \prod_{j}^{q_i} \frac{\Gamma(\alpha_{ij})}{\prod_{k=1}^{r_i} \Gamma(\alpha_{ijk})} \cdot \frac{\prod_{k=1}^{r_i} \Gamma(a_{ijk} + N_{ijk})}{\Gamma(\alpha_{ij} + N_{ij})}$$

Learning the structure

- **Likelihood of data for the BBN** (structure + parameters)

$$P(D \mid S, \Theta, \xi)$$

measures the goodness of fit of the BBN to data

- **Marginal likelihood** (for the structure only)

$$P(D \mid S, \xi)$$

- **Does not measure only a goodness of fit. It is:**
  - different for structures of different complexity
  - Incorporates preferences towards simpler structures, **implements Occam's razor** !!!!

Occam's Razor

- Why there is a preference towards simpler structures ?

Rewrite marginal likelihood as

$$P(D \mid S, \xi) = \frac{\int_{\Theta} P(D \mid S, \Theta, \xi)\, p(\Theta \mid S, \xi)\, d\Theta}{\int_{\Theta} p(\Theta \mid S, \xi)\, d\Theta}$$

We know that $\int_{\Theta} p(\Theta \mid S, \xi) d\Theta = 1$

Interpretation: in more complex structures there are more ways how parameters can be set badly

- **The numerator:** count of good assignments
- **The denominator:** count of all assignments

Approximations of probabilistic scores

Approximations of the marginal likelihood and posterior scores

- **Information based measures**
  - Akaike criterion
  - Bayesian information criterion (BIC)
  - Minimum description length (MDL)
- Reflect the tradeoff between the fit to data and preference towards simpler structures

Example: **Akaike criterion.**

Maximize: $score(S) = \log P(D \mid S, \Theta_{ML}, \xi) - \text{compl(S)}$

Bayesian information criterion (BIC)

Maximize:

$$score(S) = \log P(D \mid S, \Theta_{ML}, \xi) - \frac{1}{2} \text{compl(S) logN}$$

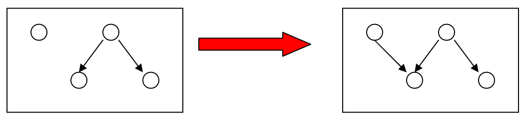
Optimizing the structure

Finding the best structure is a **combinatorial optimization** problem

- A good feature: the score **is decomposable along variables**:

$$\log P(D \mid S, \xi) = \sum_{i=1}^{n} \left\{ \sum_{j=1}^{q_i} \log \frac{\Gamma(\alpha_{ij})}{\Gamma(\alpha_{ij} + N_{ij})} + \sum_{k=1}^{r_i} \log \frac{\Gamma(\alpha_{ijk} + N_{ijk})}{\Gamma(\alpha_{ijk})} \right\}$$

Algorithm idea: Search the space of structures using local changes (additions and deletions of a link)



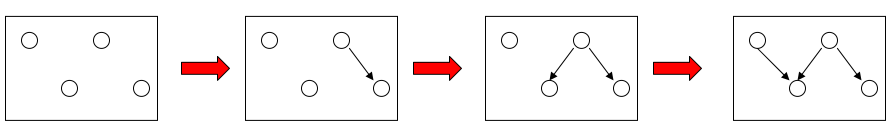
Advantage:

– we do not have to compute the whole score from scratch

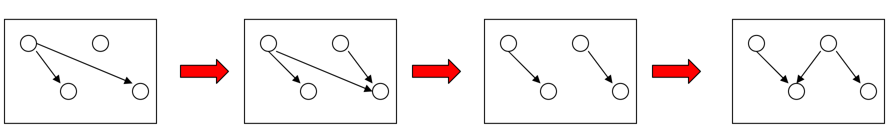
– Recompute the partial score for the affected variable

Optimizing the structure. Algorithms

- **Greedy search**
  – Start from structure with no links
  – Add a link that yields the best score improvement



- **Metropolis algorithm (with simulated annealing)**
  – Local additions and deletions
  – Avoids being trapped in “local” optimal



Density estimation with hidden variables and missing values

Learning probability distribution

Basic learning settings:

- A set of random variables $\mathbf{X} = \{X_1, X_2, \ldots, X_n\}$
- **A model of the distribution** over variables in $\boldsymbol{X}$ with parameters Θ
- **Data** $D = \{D_1, D_2, .., D_N\}$

  s.t. $D_i = (x_1^i, x_2^i, \ldots x_n^i)$

Objective: find parameters $\hat{\Theta}$ that describe the data

Assumptions considered so far:

- Known parameterizations
- No hidden variables
- No-missing values

Hidden variables

Modeling assumption:

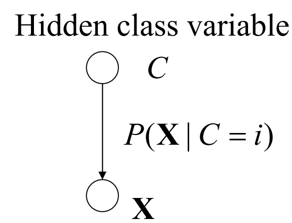
Variables $\mathbf{X} = \{X_1, X_2, \ldots, X_n\}$ are related through hidden variables

Why to add hidden variables?

- **More flexibility in describing the distribution $P(\mathbf{X})$**
- **Smaller parameterization of $P(\mathbf{X})$**
  - **New independences can be introduced via hidden variables**

Example:

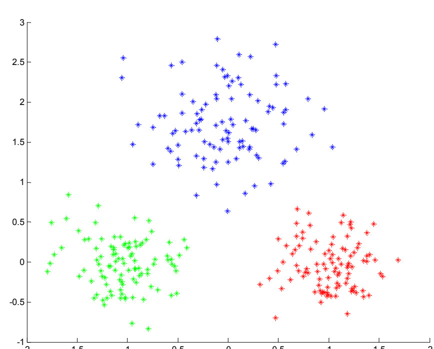
- Latent variable models
  - hidden classes (categories)



Hidden variable model. Example.

- We want to represent the probability model of a population in a two dimensional space $\mathbf{X} = \{X_1, X_2\}$

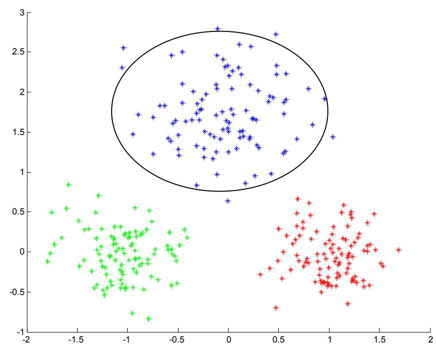
Observed data



Hidden variable model

- We want to represent the probability model of a population in a two dimensional space $\mathbf{X} = \{X_1, X_2\}$

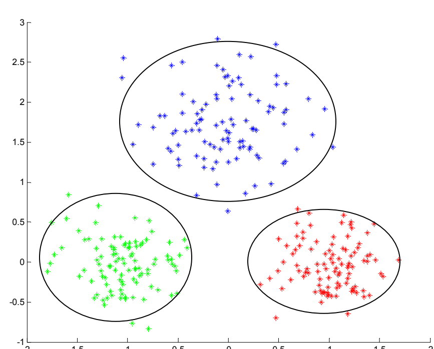
Observed data



Hidden variable model

- We want to represent the probability model of a population in a two dimensional space $\mathbf{X} = \{X_1, X_2\}$

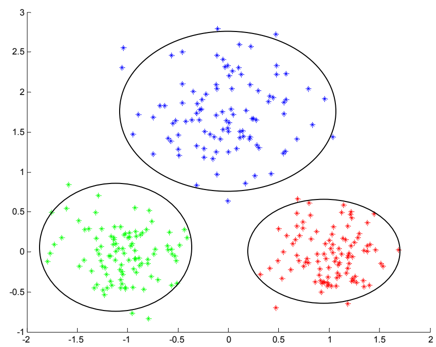
Observed data



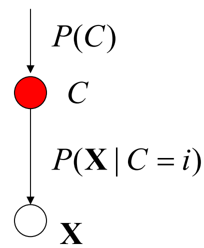
Hidden variable model

- We want to represent the probability model of a population in a two dimensional space $\mathbf{X} = \{X_1, X_2\}$

Observed data



Model : 3 Gaussians with a hidden class variable



Mixture of Gaussians

Probability of the occurrence of a data point $\boldsymbol{x}$ is modeled as

$$p(\mathbf{x}) = \sum_{i=1}^{k} p(C = i) p(\mathbf{x} \mid C = i)$$

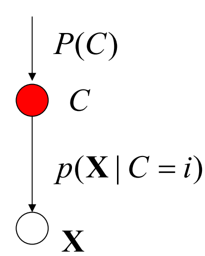
where

$p(C = i)$

= probability of a data point coming from class C=i

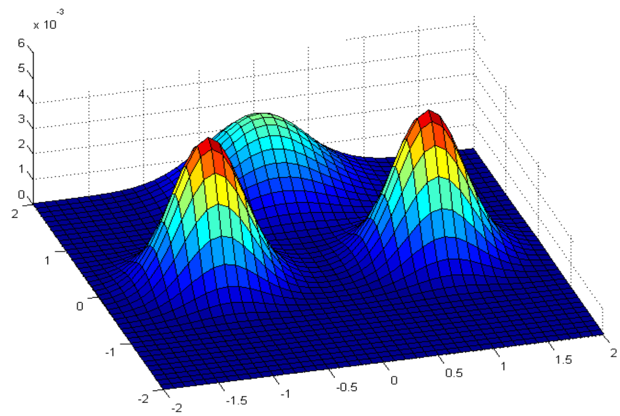
$p(\mathbf{x} \mid C = i) \approx N(\boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i)$

= class-conditional density (modeled as Gaussian) for class i



Mixture of Gaussians

- Density function for the Mixture of Gaussians model



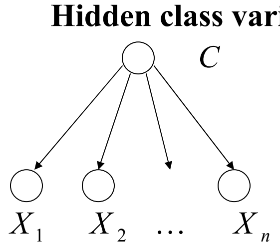
Naïve Bayes with a hidden class variable

Introduction of a hidden variable can reduce the number of parameters defining $P(\mathbf{X})$

Example:

- Naïve Bayes model with a hidden class variable

Hidden class variable



Attributes are independent given the class

- **Useful in customer profiles**
  - Class value = type of customers

Missing values

A set of random variables $\mathbf{X} = \{X_1, X_2, \ldots, X_n\}$

- **Data** $D = \{D_1, D_2, .., D_N\}$
- **But some values are missing**

$D_i = (x_1^i, x_3^i, \ldots x_n^i)$

Missing value of x_2^i

$D_{i+1} = (x_3^i, \ldots x_n^i)$

Missing values of x_1^i, x_2^i

Etc.

- **Example: medical records**
- **We still want to estimate parameters of** $P(\mathbf{X})$

Density estimation

Goal: Find the set of parameters $\hat{\Theta}$

Estimation criteria:

- **ML** $\max_{\mathbf{\Theta}} p(D \mid \mathbf{\Theta}, \xi)$
- **Bayesian** $p(\mathbf{\Theta} \mid D, \xi)$

Possible optimization methods for the ML: gradient-ascent, conjugate gradient, Newton-Rhapson, etc.

- **Problem:** No or very small advantage from the structure of the corresponding belief network

Expectation-maximization (EM) method

- An alternative optimization method
- Suitable when there are missing or hidden values
- **Takes advantage of the structure of the belief network**