(Semi-)Predictive Data Discretization

Based on Steck, Jaakkola: (Semi-)Predictive Discretization During Model Selection

Arto Klami

20th March 2004

Contents

Motivation

Why discretization?

- Numerous methods are only applicable to discrete data, e.g. information bottleneck
- Sometimes the data is known to be discrete (e.g. binary on/off), though we only have real-valued data due to noise
- Computational reasons, discretization as preprocessing

"Traditional" methods

- Any (hard) clustering algorithm can be used for discretization, but they are not optimal
- Idea: minimize the loss of information that the given variable may contain about other variables
- Class-based methods: find discretization so that the entropy of class distributions is preserved maximally within the discretization levels
- A similar approach to unlabeled data by Monti and Cooper
- Number of possible discretizations is exponential in the number of samples $\longrightarrow$ greedy methods

Proposed approach

- Here a method that optimizes the discretization during the model learning is introduced
- Applied to graphical models, which can be relatively effectively computed for discrete data
- The task is to discretize the data and learn the structure of the graphical model (dependencies between variables) at the same time
- Some algorithms proposed earlier for the same task, but they are computationally too heavy
- In practice: maximize the likelihood of data with respect to the discretization and the model structure

Notation

- Denote by Λ a univariate discretization policy, that is, a sequence of $r_k - 1$ threshold values λ_j
- Discretization by mapping $f_\Lambda(y) = j$ if $\lambda_{j-1} \leq y < \lambda_j$
- N samples and n variables
- ith sample of kth variable is denoted by $y_k^{(i)}$ and the corresponding discrete value by $x_k^{(i)}$
- m is used to denote the model structure

Sequential approach

- Likelihood computed sequentially, given the discretization

$$p(D|\Lambda, m) = \prod_{i=1}^{N} p(y^{(i)}|D^{(i-1)}, \Lambda, m)$$

- The predictive distribution factors

$$p(y^{(i)}|D^{(i-1)}, \Lambda, m) = p(y^{(i)}|x^{(i)}, \Lambda)p(x^{(i)}|D^{(i-1)}, \Lambda, m)$$

- The second part is "easy" with discrete variables, first needs to be studied
- If m and x capture all dependencies, the variables y are independent given x, $p(y^{(i)}|x^{(i)}, \Lambda) = \prod_{k=1}^{n} p(y_k^{(i)}|x^{(i)}, \Lambda_k)$

Finest grid

- Consider the finest possible discretization (Ω) of the data set with exactly one data point in each discretization level
- This is called the finest grid implied by the data, and the thresholds can be freely selected between the data points
- "Restrict" the discretization policy Λ so that the thresholds are picked from the thresholds of Ω
- Denote by z the discretized value according to Ω and by x the discretized value according to Λ

$$p(y_k^{(i)}|x^{(i)}, \Lambda_k, \Omega_k) = p(y_k^{(i)}|z_k^{(i)}, \Omega_k)p(z_k^{(i)}|x^{(i)}, \Lambda_k, \Omega_k)$$

Conceptual summary

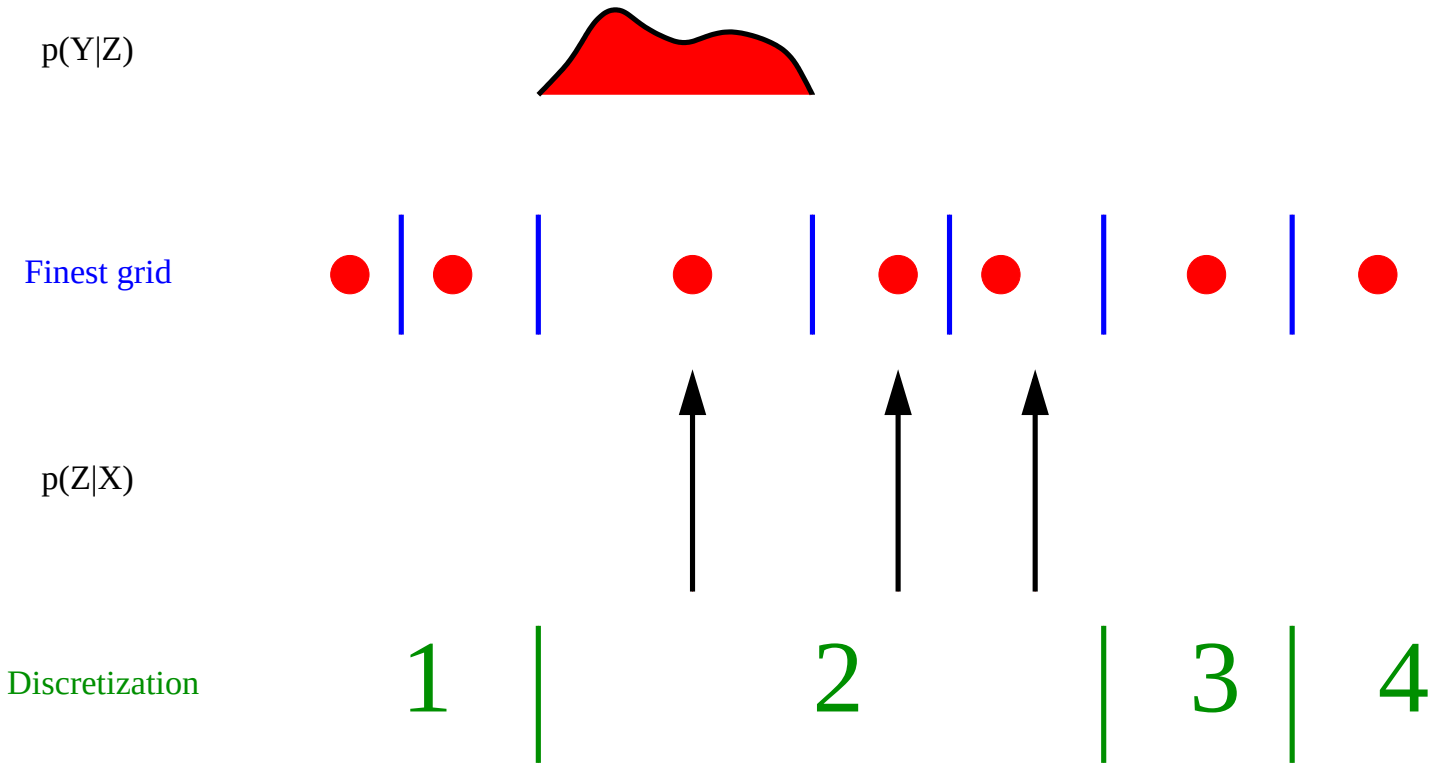
- The likelihood of the observed data can be factorized as

$$p(Y|D,\Lambda,m) = p(X|D,\Lambda,m)p(Z|X,\Lambda,\Omega)p(Y|Z,\Omega)$$

- The first term is familiar for discrete data
- Z is assumed to be evenly distributed given X

$$p(z_k^{(i)}|x^{(i)},\Lambda_k,\Omega_k) = \frac{1}{N(x_k^{(i)})}$$

- Any distribution is allowed for the third term


p(Y|Z)
Finest grid
p(Z|X)
Discretization
1
2
3
4

Semi-predictive discretization 1/2

- Optimize the likelihood

$$p(D|\Lambda, m, \Omega) = p(D_\Lambda|m) \left(\prod_{i=1}^{N} \prod_{k=1}^{n} \frac{1}{N(x_k^{(i)})} \right) \left(\prod_{i=1}^{N} \prod_{k=1}^{n} p(y_k^{(i)}|z_k^{(i)}, \Omega_k) \right)$$

- Semi-predictive because the finest grid (Ω) is computed using the whole data set, not just the previous samples
- First term: likelihood of the graph m given the discrete values — basic stuff
- Second term: Can be written as the reciprocal of the maximum likelihood of an empty graph (times constant), $p(D_\Lambda|\hat{\theta}, m_e)^{-1}$
- Third term: Independent of Λ and m and thus irrelevant

Semi-predictive Discretization 2/2

- The final cost function

$$L(\Lambda, m) = \log p(D_\Lambda|m) - \log p(D_\Lambda|\hat{\theta}, m_e) = \log p(D_\Lambda|m) + N \sum_{k=1}^{n} H(\hat{p}(X_k))$$

- Both likelihoods increase with diminishing number of discretization levels $\longrightarrow$ entropy penalizes for coarse discretization
- Depends only on counts of data, and is independent of the metric of the continuous space
- If all variables are independent, the discretization is chosen to optimize predictions $\longrightarrow$ one level is optimal

Predictive Discretization

- Also Ω is formed based only on the previous samples
- Leads to cost

$$L(\Lambda, m) = \log p(D_\Lambda | m) - \log G(D, \Lambda) \ ,$$

  where

$$G(D, \Lambda) = \prod_{k=1}^{n} \prod_{x_k} \frac{1}{\Gamma(N(x_k))}$$

- The difference between methods is relevant
- Predictive discretization favors slightly more discretization levels

Experiments

- Find the structure of pheromone response pathway in yeast
- Method is invariant to continuously differentiable, monotonic transformations, so no preprocessing of data is needed
- A greedy algorithm used for simplicity
  1. Given discretized data, optimize m locally
  2. Given m, optimize Λ iteratively for each variable at a time
- Heuristic to avoid local maxima: optimize m and Λ only slightly at each step
- The resulting network has clearly different structure than what has been found in earlier studies, and it seems biologically plausible

Conclusions

- A method for optimizing discretization while learning Bayes network structure was introduced
- The discretization seems to be important for the structure determination
- Computationally difficult task, here using the finest grid makes computations possible