Predictive Discretization During Model Selection

Harald Steck[1] and Tommi S. Jaakkola[2]

[1] Institute for Computational Science, ETH Zurich, 8092 Zurich, Switzerland
hsteck@inf.ethz.ch
[2] MIT CSAIL, Stata Center, Bldg. 32-Gates 498, Cambridge, MA 02139, USA
tommi@csail.mit.edu

Abstract. We present an approach to discretizing multivariate continuous data while learning the structure of a graphical model. We derive a joint scoring function from the principle of predictive accuracy, which inherently ensures the optimal trade-off between goodness of fit and model complexity including the number of discretization levels. Using the so-called finest grid implied by the data, our scoring function depends only on the number of data points in the various discretization levels (independent of the metric used in the continuous space). Our experiments with artificial data as well as with gene expression data show that discretization plays a crucial role regarding the resulting network structure.

1 Introduction

Continuous data is often discretized as part of a more advanced approach to data analysis such as learning graphical models. Discretization may be carried out merely for computational efficiency, or because background knowledge suggests that the underlying variables are indeed discrete. While it is computationally efficient to discretize the data in a preprocessing step that is independent of the subsequent analysis [6,10,7], the impact of the discretization policy on the subsequent analysis is often unclear in this approach. Existing methods that optimize the discretization policy jointly with the graph structure [3,9] are computationally very involved and therefore not directly suitable for large domains.

We present a novel and more efficient scoring function for joint optimization of the discretization policy and the model structure. The objective relies on predictive accuracy, where predictive accuracy is assessed sequentially as in prequential validation [2] or stochastic complexity [12].

2 Sequential Approach

Let $Y = (Y_1, ..., Y_k, ..., Y_n)$ denote a vector of n continuous variables in the domain of interest, and y any specific instantiation of these variables. The discretization of Y is determined by a *discretization policy* $\Lambda = (\Lambda_1, ..., \Lambda_n)$: for each variable Y_k, let $\Lambda_k = (\lambda_{k,1}, ..., \lambda_{k,r_k-1})$ be ordered threshold values, and r_k be the

C.E. Rasmussen et al. (Eds.): DAGM 2004, LNCS 3175, pp. 1–8, 2004.

number of discretization levels. This determines the mapping $f_\Lambda : Y \mapsto X$, where $X = (X_1, ..., X_k, ..., X_n)$ is the corresponding discretized vector; for efficiency reasons we only consider *deterministic* discretizations, where each continuous value y is mapped to *exactly one* discretization level, $x_k = f_{\Lambda_k}(y_k)$.

We pretend that (continuous) *i.i.d.* data D arrive in a sequential manner, and then assess predictive accuracy regarding the data points along the sequence. This is similar to prequential validation or stochastic complexity [2,12]. We recast the joint marginal likelihood of the discretization policy Λ and the structure m of a graphical model in a sequential manner,

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

where $D^{(i-1)} = (y^{(i-1)}, y^{(i-2)}, ..., y^{(1)})$ denotes the data points seen *prior to* step i along the sequence.

For deterministic discretization we can assume that at each step i the predicted density regarding data point $y^{(i)}$ factors according to $\rho(y^{(i)}|D^{(i-1)}, \Lambda, m) = \rho(y^{(i)}|x^{(i)}, \Lambda)\ p(x^{(i)}|D^{(i-1)}, m, \Lambda)$, where $x^{(i)} = f_\Lambda(y^{(i)})$. It is desirable that the structure m indeed captures *all* the relevant (conditional) dependences among the variables $Y_1, ..., Y_n$. Assuming that the dependences among continuous Y_k are described by the *discretized* distribution $p(X|m, \Lambda, D)$, then any two continuous variables Y_k and $Y_{k'}$ are independent conditional on X: $\rho(y^{(i)}|x^{(i)}, \Lambda) = \prod_{k=1}^{n} \rho(y_k^{(i)}|x^{(i)}, \Lambda_k)$.

The computational feasibility of this approach depends crucially on the efficiency of the mapping between the discrete and continuous spaces. A simple approach may use the *same* density to account for points y and y' that are mapped to the same discretized state x, cf. [9]. Assuming a *uniform* probability density is overly stringent and degrades the predictive accuracy; moreover, this might also give rise to "empty states", cf. [15]. In contrast, we require only *independence* of the variables Y_k.

3 Finest Grid Implied by the Data

The *finest grid implied by the data* is a simple mapping between Y and X that retains the desired independence properties with non-uniform densities, and can be computed efficiently.

This grid is obtained by discretizing each variable Y_k such that the corresponding (new) discrete variable Z_k has as many states as there are data points, and exactly one data point is assigned to each of those states (an extension to the case with identical data points is straightforward; also note that this grid is not unique, as any threshold value between neighboring data points can be chosen). Note that, in our predictive approach, this grid is based on data $D^{(i-1)}$ at each step i.

Based on this grid, we can now obtain an efficient mapping between Y and X as follows: we assume that two points y_k and y'_k in the continuous space get assigned the same density if they map to the same state of Z_k; and that two states z_k and z'_k of Z_k get assigned the same probability mass if they map to the

same discretization level of X_k (we require that each state of Z_k is mapped to exactly one discretization level of X_k for computational efficiency). This immediately yields $\rho(y_k^{(i)}|x^{(i)}, \Lambda_k) = c/N^{(i)}_{x_k^{(i)}}$, where $N^{(i-1)}_{x_k^{(i)}}$ denotes the number of data points in discretization level $x_k^{(i)}$ of variable X_k before step i along the sequence $(N^{(i-1)}_{x_k^{(i)}} > 0)$. The constant c absorbs the mapping from Z to Y by means of the finest grid. Using the *same* grid for two models being compared, we have the important property that c is irrelevant for determining the optimal Λ and m. Unfortunately, details have to be skipped here due to lack of space, see also [15].

4 Predictive Discretization

In our sequential approach, the density at data point $y^{(i)}$ is predicted strictly without hindsight at each step i, i.e., only data $D^{(i-1)}$ is used. For this reason, this leads to a fair assessment of predictive accuracy. Since *i.i.d.* data lack an inherent sequential ordering, we may choose a *particular* ordering of the data points. This is similar in spirit to stochastic complexity [12], where also a *particular* sequential ordering is used. The basic idea is to choose an ordering such that, for all x_k, we have $N^{(i-1)}_{x_k} > 0$ for all $i \geq i_0$, where i_0 is minimal. The initial part of this sequence is thus negligible compared to the part where $i = i_0, ..., N$ when the number of data points is considerably larger than the number of discretization levels of any single variable, $N \gg \max_k |X_k|_\Lambda$. Combining the above equations, we obtain the following (approximate) predictive scoring function $\mathcal{L}(\Lambda, m)$:

$$\log \rho(D|\Lambda, m) \approx \mathcal{L}(\Lambda, m) + c' = \log p(D_\Lambda|m) - \log G(D, \Lambda) + c', \qquad (1)$$

where the approximation is due to ignoring the short initial part of the sequence; $p(D_\Lambda|m)$ is the marginal likelihood of the graph m in light of the data D_Λ discretized according to Λ. In a Bayesian approach, it can be calculated easily for various graphical models, e.g., see [1,8] concerning discrete Bayesian networks. The second term in Eq. 1 is given by

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

where Γ denotes the Gamma function, $\Gamma(N(x_k)) = [N(x_k) - 1]!$, and $N(x_k)$ is the number of data points in discretization level x_k.[1] It is crucial that the constant c', which collects the constants c from above, is irrelevant for determining the optimal Λ and m. Obeying lack of space, the reader is referred to [15] for further details.

Our scoring function $\mathcal{L}(\Lambda, m)$ has several interesting properties: First, the difference between the two terms in Eq. 1 determines the trade-off dictating the optimal number of discretization levels, threshold values and graph structure. As both terms increase with a diminishing number of discretization levels,

[1] Note that $N(x_k) > 0$ is ensured in our approach, i.e., there are no "empty states" [15].

the second term can be viewed as a penalty for small numbers of discretization levels. Second, $\mathcal{L}(\Lambda, m)$ depends on the number of data points in the different discretization levels only. This is a consequence of the *finest grid implied by the data*. It has several interesting implications. First, and most important from a practical point of view, it renders efficient evaluation of the scoring function possible. Second, and more interesting from a conceptual perspective, $\mathcal{L}(\Lambda, m)$ is independent of the particular choice of the finest grid. Apart from that, $\mathcal{L}(\Lambda, m)$ is independent of the metric in the continuous space, and thus invariant under monotonic transformations of the continuous variables. Obviously, this can lead to considerable loss of information, particularly when the (Euclidean) *distances* among the various data points in the continuous space govern the discretization (cf. left graph in Fig. 1). On the other hand, the results of our scoring function are not degraded if the data is given w.r.t. an inappropriate metric. In fact, the optimal discretization w.r.t. our scoring function is based on *statistical dependence* of the variables, rather than on the *metric*. This is illustrated in our toy experiments with artificial data, cf. Section 5. Apart from that, our approach includes as a special case quantile discretization, namely when all the variables are independent of each other.

5 Experiments

In our first two experiments, we show that our approach discretizes the data based on statistical dependence rather than on the metric in the continuous space. Consider the left two panels in Fig. 1: when the variables are *independent*, our approach may not find the discretization suggested by the clusters; instead, our approach assigns the same number of data points to each discretization level (with one discretization level being optimal). Note that discretization of independent variables is, however, quite irrelevant when learning graphical models: the optimal discretization of each variable Y_k depends on the variables in its Markov blanket, and Y_k is (typically strongly) dependent on those variables. When the variables are *dependent* in Fig. 1, our scoring function favours the "correct" discretization (solid lines), as this entails best predictive accuracy (even when disregarding the metric). However, dependence of the variables itself does not necessarily ensure that our scoring function favours the "correct" discretization, as illustrated in the right two panels in Fig. 1 (as a constraint, we require two discretization levels): given low noise levels, our scoring function assigns the same number of data points to each discretization level; however, a sufficiently *high* noise level in the data can actually be beneficial, permitting our approach to find the "correct" discretization, cf. Fig. 1 (right).

Our third experiment demonstrates that our scoring function favours less complex models (i.e., sparser graphs and fewer discretization levels) when given smaller data sets. This is desirable in order to avoid overfitting when learning from small samples, leading to optimal predictive accuracy. We considered a pair of normally distributed random variables Y_0 and Y_1 with correlation coefficient $corr(Y_0, Y_1) = 1/\sqrt{2}$. Note that this distribution does not imply a 'natural' number of discretization levels; due to the dependence of Y_0 and Y_1 one may

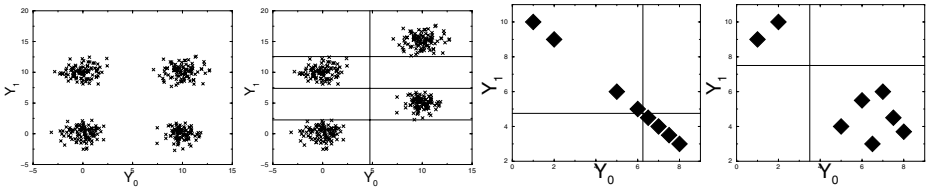

Fig. 1. Left two panels: each cluster comprises 100 points sampled from a Gaussian distribution; Y_0 and Y_1 are independent on the left, and dependent on the right. Right two panels: when Y_0 and Y_1 are dependent, *noise* may help in finding the 'correct' discretization.

hence expect the *learned* number of discretization levels to rise with growing sample size. Indeed, Fig. 2 shows exactly this behavior. Moreover, the learned graph structure implies independence of Y_0 and Y_1 when given very small samples (fewer than 30 data points in our experiment), while Y_0 and Y_1 are found to be dependent for all larger sample sizes.

In our fourth experiment, we were concerned with gene expression data. In computational biology, regulatory networks are often modeled by Bayesian networks, and their structures are learned from discretized gene-expression data, see, e.g., [6,11,7]. Obviously, one would like to recover the "true" network structure underlying the continuous data, rather than a degraded network structure due to a suboptimal discretization policy. Typically, the expression levels have been discretized in a preprocessing step, rather than jointly with the network structure, [6,11,7]. In our experiment, we employed our predictive scoring function (cf. Eq. 1) and re-analyzed the gene expression data concerning the pheromone response pathway in yeast [7], comprising 320 measurements concerning 32 continuous variables (genes) as well as the mating type (binary variable). Based on an error model concerning the micro-array measurements, a continuously differentiable, monotonic transformation is typically applied to the raw gene expression data in a preprocessing step. Since our predictive scoring function is invariant under this kind of transformation, this has no impact on our analysis, so that we are able to work directly with the raw data.

Instead of using a search strategy in the *joint* space of graphs and discretization policies — the theoretically best, but computationally most involved approach — we optimize the graph m and the discretization policy Λ alternately in a greedy way for simplicity: given the discretized data D_Λ, we use local search to optimize the graph m, like in [8]; given m, we optimize Λ iteratively by improving the discretization policy regarding a *single* variable given its Markov blanket at a time. The latter optimization is carried out in a hierarchical way over the number of discretization levels and over the threshold values of each variable. Local maxima are a major issue when optimizing the predictive scoring function due to the (strong) interdependence between m and Λ. As a simple heuristic, we alternately optimize Λ and m only slightly at each step.

The marginal likelihood $p(D_\Lambda|m)$, which is part of our scoring function, contains a free parameter, namely the so-called scale-parameter α regarding the

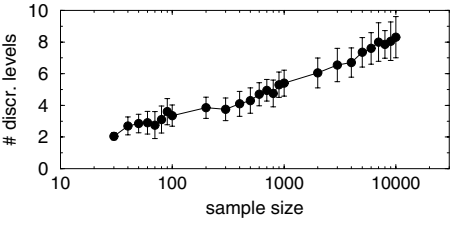

Fig. 2. The number of discretization levels (mean and standard deviation, averaged over 10 samples of each size) depends on the sample size (cf. text for details).

Dirichlet prior over the model parameters, e.g., cf. [8]. As outlined in [13], its value has a decisive impact on the resulting number of edges in the network, and must hence be chosen with great care. Assessing predictive accuracy by means of 5-fold cross validation, we determined $\alpha \approx 25$.

Fig. 3 shows the composite graph we learned from the used gene expression data, employing our predictive scoring function, cf. Eq. 1.[2] This graph is compiled by averaging over several Bayesian network structures in order to account for model uncertainty prevailing in the small data set. Instead of exploring model uncertainty by means of Markov Chain Monte Carlo in the model space, we used a non-parametric re-sampling method, as the latter is independent of any model assumptions. While the bootstrap has been used in [5,4,6,11], we prefer the jackknife when learning the graph structure, i.e., conditional independences. The reason is that the bootstrap procedure can easily induce spurious dependencies when given a small data set D; as a consequence, the resulting network structure can be considerably biased towards denser graphs [14]. The jackknife avoids this problem. We obtained very similar results using three different variants of the jackknife: delete-1, delete-30, and delete-64. Averaging over 320 delete-30 jackknife sub-samples, we found 65.7 ± 8 edges. Fig. 3 displays 65 edges: the solid ones are present with probability $> 50\%$, and the dashed ones with probability $> 34\%$. The orientation of an edge is indicated only if one direction is at least twice as likely as the contrary one. Apart from that, our predictive scoring function yielded that most of the variables have about 4 discretization levels (on average over the 320 jackknife samples), except for the genes MCM1, MFALPHA1, KSS1, STE5, STE11, STE20, STE50, SWI1, TUP1 with about 3 states, and the genes BAR1, MFA1, MFA2, STE2, STE6 with ca. 5 states.

In Fig. 3, it is apparent that the genes AGA2, BAR1, MFA1, MFA2, STE2, and STE6 (magenta) are densely interconnected, and so is the group of genes MFALPHA1, MFALPHA2, SAG1 and STE3 (red). Moreover, both of those groups are directly connected to the mating type, while the other genes in the network are (marginally) independent of the mating type. This makes sense

[2] We imposed no constraints on the network structure in Fig. 3. Unfortunately, the results we obtained when imposing constraints derived from location data have to be skipped due to lack of space.

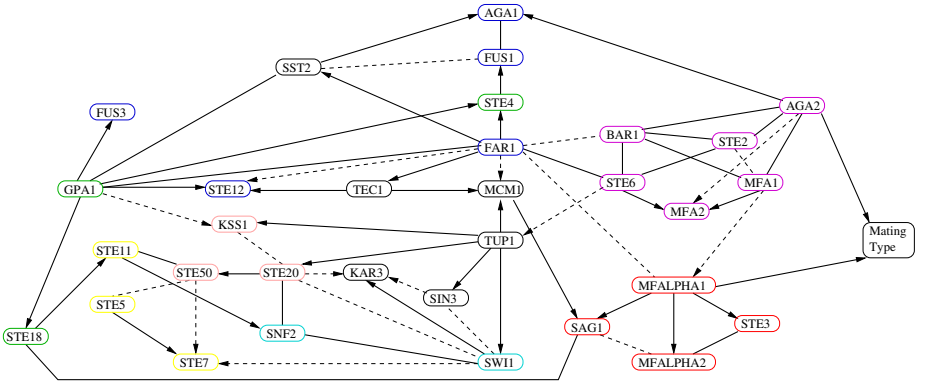

Fig. 3. This graph is compiled from 320 delete-30 jackknife samples (cf. [7] for the color-coding).

from a biological perspective, as the former genes (magenta) are only expressed in yeast cells of mating type A, while the latter ones (red) are only expressed in mating type ALPHA; the expression level of the other genes is rather unaffected by the mating type. Due to lack of space, a more detailed (biological) discussion has to be omitted here.

Indeed, this grouping of the genes is supported also when considering correlations as a measure of statistical dependence:[3] we find that the absolute value of the correlations between the mating type and each gene in either group from above is larger than 0.38, while any other gene is only weakly correlated with the mating type, namely less than 0.18 in absolute value.

The crucial impact of the used discretization policy Λ and scale-parameter α on the resulting network structure becomes apparent when our results are compared to the ones reported in [7]: their network structure resembles a naive Bayesian network, where the mating type is the root variable. Obviously, their network structure is notably different from ours in Fig. 3, and hence has very different (biological) implications. Unlike in [7], we have optimized the discretization policy Λ and the network structure m jointly, as well as the scale-parameter α. As the value of the scale-parameter α mainly affects the *number* of edges present in the learned graph [13], this suggests that the major differences in the obtained network structures are actually due to the discretization policy.

6 Conclusions

We have derived a principled yet efficient method for determining the resolution at which to represent continuous observations. Our discretization approach relies on predictive accuracy in the prequential sense and employs the so-called finest

[3] Note that correlations are applicable here, even though they measure only linear effects. This is because the mating type is a *binary* variable.

grid implied by the data as the basis for finding the appropriate levels. Our experiments show that a suboptimal discretization method can easily degrade the obtained results, which highlights the importance of the principled approach we have proposed.

Acknowledgements. We would like to thank Alexander Hartemink for making the pheromone data available to us. Harald Steck acknowledges support from the German Research Foundation (DFG) under grant STE 1045/1-2. Tommi Jaakkola acknowledges support from the Sloan Foundation in the form of the Sloan Research Fellowship.

References

1. G. Cooper and E. Herskovits. A Bayesian method for the induction of probabilistic networks from data. *Machine Learning*, 9:309–47, 1992.
2. A. P. Dawid. Statistical theory. The prequential approach. *Journal of the Royal Statistical Society, Series A*, 147:277–305, 1984.
3. N. Friedman and M. Goldszmidt. Discretization of continuous attributes while learning Bayesian networks. In *ICML*, pages 157–65, 1996.
4. N. Friedman, M. Goldszmidt, and A. Wyner. Data analysis with Bayesian networks: A bootstrap approach. In *UAI*, pages 196–205, 1999.
5. N. Friedman, M. Goldszmidt, and A. Wyner. On the application of the bootstrap for computing confidence measures on features of induced Bayesian networks. In *AI & STATS*, pages 197–202, 1999.
6. N. Friedman, M. Linial, I. Nachman, and D. Pe'er. Using Bayesian networks to analyze expression data. *Journal of Computational Biology*, 7:601–20, 2000.
7. A. J. Hartemink, D. K. Gifford, T. S. Jaakkola, and R. A. Young. Combining location and expression data for principled discovery of genetic regulatory networks. In *Pacific Symposium on Biocomputing*, 2002.
8. D. Heckerman, D. Geiger, and D. M. Chickering. Learning Bayesian networks: The combination of knowledge and statistical data. *Machine Learning*, 20:197–243, 1995.
9. S. Monti and G. F. Cooper. A multivariate discretization method for learning Bayesian networks from mixed data. *14^{th} UAI*, pages 404–13, 1998.
10. S. Monti and G. F. Cooper. A latent variable model for multivariate discretization. In *AI & STATS*, pages 249–54, 1999.
11. D. Pe'er, A. Regev, G. Elidan, and N. Friedman. Inferring subnetworks from perturbed expression profiles. *Bioinformatics*, 1:1–9, 2001.
12. J. Rissanen. Stochastic complexity and modeling. *The Annals of Statistics*, 14:1080–100, 1986.
13. H. Steck and T. S. Jaakkola. On the Dirichlet prior and Bayesian regularization. In *NIPS 15*, 2002.
14. H. Steck and T. S. Jaakkola. Bias-corrected bootstrap and model uncertainty. *NIPS 16*, 2003.
15. H. Steck and T. S. Jaakkola. (Semi-)predictive discretization during model selection. *AI Memo 2003-002, MIT*, 2003.