Discovering Latent Structures: Experience with the CoIL Challenge 2000 Data Set

Nevin L. Zhang

Hong Kong University of Science and Technology, Hong Kong, China
lzhang@cse.ust.hk

Abstract. We present a case study to demonstrate the possibility of discovering complex and interesting latent structures using hierarchical latent class (HLC) models. A similar effort was made earlier [6], but that study involved only small applications with 4 or 5 observed variables. Due to recent progress in algorithm research, it is now possible to learn HLC models with dozens of observed variables. We have successfully analyzed a version the CoIL Challenge 2000 data set that consists of 42 observed variable. The model obtained consists of 22 latent variables, and its structure is intuitively appealing.

Keywords: Latent structure discovery, Bayesian networks, learning, case study.

1 Introduction

Hierarchical latent class (HLC) models [7] are tree-structured Bayesian networks where variables at leaf nodes are observed and are hence called *manifest variables*, while variables at internal nodes are hidden and hence are called *latent variables.* All variables are assumed discrete. HLC models generalize latent class (LC) models [3] and were first identified as a potentially useful class of Bayesian networks (BN) by Pearl [4].

HLC models can used for latent structure discovery. Often, observed variables are correlated because they are influenced by some common hidden causes. HLC models can be seen as hypotheses about how latent causes influence observed variables and how they are correlated among themselves. Finding an HLC model that fits data amounts to finding a latent structure that can explain data well.

The CoIL Challenge 2000 data set [5] contains information on customers of a Dutch insurance company. The data consists of 86 variables, around half of which are about ownership of various insurance products. Different product ownership variables are correlated. One who pays a high premium on one type of insurance is more likely, than those who do not, to also purchase other types of insurance. Intuitively, such correlations are due to people's (latent) attitudes toward risks. The more risk-aversion one is toward one category of risks, the more likely one is to purchase insurance products in that category. Therefore, the CoIL Challenge 2000 data set is a good testbed for latent structure discovery methods.

Y. Shi et al. (Eds.): ICCS 2007, Part IV, LNCS 4490, pp. 26–34, 2007.
© Springer-Verlag Berlin Heidelberg 2007

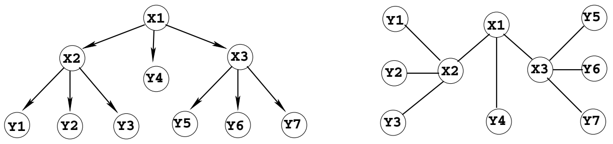

Fig. 1. An example HLC model and the corresponding unrooted HLC model. The X_i's are latent variables and the Y_j's are manifest variables.

We have analyzed the CoIL Challenge 2000 data set using HLC models. The structure of the model obtained is given in Section 4. There are 42 manifest variables and 22 latent variables, and the structure is intuitively very appealing. Latent structure discovery is very difficult. It is hence exciting to know that we are able to discover such a complex and meaningful structure.

HLC models can also be used simply for probabilistic modeling. They possess two nice properties for this purpose. First, they have low inferential complexity due to their tree structures. Second, they can model complex dependencies among the observed. In Section 5, the reader will see the implications of the second property on prediction and classification accuracy in the context of the CoIL Challenge 2000 data.

We begin with a review of HLC models in Section 2 and a description of the CoIL Challenge 2000 data set in Section 3.

2 Hirarchical Latent Class Models

Figure 1 shows an example HLC model (left diagram). A *latent class (LC) model* is an HLC model where there is only one latent node. We usually write an HLC model as a pair $M = (m, \theta)$, where θ is the collection of parameters. The first component m consists of the model structure and cardinalities of the variables. We will sometimes refer to m also as an HLC model. When it is necessary to distinguish between m and the pair (m, θ), we call m an *uninstantiated HLC model* and the pair (m, θ) an *instantiated HLC model.*

Two instantiated HLC models $M{=}(m, \theta)$ and $M'{=}(m', \theta')$ are *marginally equivalent* if they share the same manifest variables Y_1, Y_2, ..., Y_n and

$$P(Y_1, \ldots, Y_n|m, \theta) = P(Y_1, \ldots, Y_n|m', \theta'). \tag{1}$$

An uninstantiated HLC models m *includes* another m' if for any parameterization θ' of m', there exists parameterization θ of m such that (m, θ) and (m', θ') are marginally equivalent, i.e. if m can represent any distributions over the manifest variables that m' can. If m includes m' and vice versa, we say that m and m' are *marginally equivalent.* Marginally equivalent (instantiated or uninstantiated) models are *equivalent* if they have the same number of independent parameters. One cannot distinguish between equivalent models using penalized likelihood scores.

Let X_1 be the root of an HLC model m. Suppose X_2 is a child of X_1 and it is a latent node. Define another HLC model m' by reversing the arrow $X_1 \rightarrow X_2$. In m', X_2 is the root. The operation is hence called *root walking*; the root has walked from X_1 to X_2. Root walking leads to equivalent models [7]. This implies that it is impossible to determine edge orientation from data. We can learn only *unrooted HLC models*, which are HLC models with all directions on the edges dropped. Figure 1 also shows an example unrooted HLC model. An unrooted HLC model represents a class of HLC models. Members of the class are obtained by rooting the model at various nodes. From now on when we speak of HLC models we always mean unrooted HLC models unless it is explicitly stated otherwise.

Assume that there is a collection **D** of i.i.d samples on a given set of manifest variables that were generated by an unknown regular HLC model. The learning task is to reconstruct the unrooted HLC models that corresponds to the generative model.

The first principled algorithm for learning HLC models was developed by Zhang [7]. The algorithm consists of two search routines, one optimizes model structure while the other optimizes cardinalities of latent variables in a given model structure. It is hence called *double hill-climbing (DHC).* It can deal with data sets with about one dozen manifest variables. Zhang and Kočka [8] recently proposed another algorithm called *heuristic single hill-climbing (HSHC).* HSHC combines the two search routines of DHC into one and incorporates the idea of structural EM [2] to reduce the time spent in parameter optimization. HSHC can deal with data sets with dozens of manifest variables.

Results presented in this paper were obtained using the HSHC algorithm. The algorithm hill-climbs in the space of all unrooted regular HLC models for the given manifest variables. We assume that the BIC score is used to guide the search. The BIC score of a model m is:

$$BIC(m|\mathbf{D}) = logP(\mathbf{D}|m, \theta^*) - \frac{d(m)}{2} logN$$

where θ^* is the ML estimate of model parameters, $d(m)$ is the *standard dimension* of m, i.e. the number of independent parameters, and N is the sample size.

3 The Coil Challenge 2000 Data Set

The training set of the COIL Challenge 2000 data consists of 5,822 customer records. Each records consists of 86 attributes, containing sociodemographic information (Attributes 1-43) and insurance product ownerships (Attributes 44-86). The sociodemographic data is derived from zip codes. In previous analyses, these variables were found more or less useless. In our analysis, we include only three of them, namely Attributes 43 (purchasing power class), 5 (customer main type), and 4 (average age). All the product ownership attributes are included in the analysis.

The data was preprocessed as follows: First, similar attribute values were merged so that there are at least 30 cases for each value. Thereafter, the attributes have 2 to 9 values. In the resultant data set, there are fewer than 10 cases where Attributes 50, 60, 71 and 81 take "nonzero" values. Those attributes were therefore excluded from further analysis. This leaves us with 42 attributes.

We analyzed the data using a Java implementation HSHC algorithm. In each step of search, HSHC runs EM on only one model to optimize all its parameters. However, it may run local EM on several candidate models to optimize the parameters that are affected by search operators. The number of such candidate models is denoted by K, and K is a parameter for the algorithm. We tried four values for K, namely 1, 5, 10, and 20. The experiments were run on a Pentium 4 PC with a clock rate of 2.26 GHz. The running times and the BIC scores of the resulting models are shown in the following table. The best model was found in the case of K=10. We denote the model by M^*. The structure of the model is shown in Figure 3.[1]

K	1	5	10	20
Time (hrs)	51	99	121	169
BIC	-52,522	-51,625	-51,465	-51,592

4 Latent Structure Discovery

Did HSHC discover interesting latent structures? The answer is positive. We will explain this by examining different aspects of Model M^*. First of all, the data contains two variables for each type of insurance. For bicycle insurance, for instance, there are "contribution to bicycle insurance policies (v_{62})" and "number of bicycle insurance policies (v_{83})". HSHC introduced a latent variable for each such pair. The latent variable introduced for v_{62} and v_{83} is h_{11}, which can be interpreted as "aversion to bicycle risks". Similarly, h_{10} can be interpreted as "aversion to motorcycle risks", h_9 as "aversion to moped risks", and so on.

Consider the manifest variables below h_{12}. Besides "social security", all the other variables are related to heavy private vehicles. HSHC concluded that they are influenced by one common latent variable. This is clearly reasonable and h_{12} can be interpreted as "aversion to heavy private vehicle risks". Besides "social security", all the manifest variables below h_8 are related to private vehicles. HSHC concluded that they are influence by one common latent variable. This is reasonable and h_8 can be interpreted as "aversion to private vehicle risks".

All the manifest variables below h_{15}, except "disability", are agriculture-related; while the manifest variables below h_1 are firm-related. It is therefore reasonable for HSHC to conclude that those two groups of variables are respectively influenced by two latent variables h_1 and h_{15}, which can be interpreted as "aversion to firm risks" and "aversion to agriculture risks" respectively.

It is interesting to note that, although delivery vans and tractors are vehicles, HSHC did not conclude that they are influenced by h_8. HSHC reached the correct

[1] Note that what HSHC obtains is an unrooted HLC model. The structure of the model is visually shown as a rooted tree in Figure 3 partially for readability and partially due to the discussions of the following section.

conclusion that the decisions to buy insurance for tractors, for delivery vans, or for other private vehicles are influenced by different latent factors.

The manifest variables below h_3 intuitively belong to the same category; those below h_6 are also closely related to each other. It is therefore reasonable for HSHC to conclude that those two groups of variables are respectively influenced by latent variables h_3 and h_6.

The three sociodemographic variables (v_{04}, v_{05}, and v_{43}) are connected to latent variable h_{21}. Hence h_{21} can be viewed as a venue for summarizing information contained in those three variables. Latent variable h_0 can interpreted as "general attitude toward risks". Under this interpretation, the links between h_0 and its neighbors are all intuitively reasonable: One's general attitude toward risks should be related to one's sociodemographic status (h_{21}), and should influence one's attitudes toward specific risks (h_8, h_1, h_{15}, …, etc).

There are also aspects of Model M^* that do not match our intuition well. For example, since there is a latent variable (h_{12}) for heavy private vehicles under h_8, we would naturally expect a latent variable for light private vehicles. But there is no such variable. Below h_3, we would expect a latent variable specifically for life insurance. Again, there is no such variable. The placement of the variables about social insurance and disability is also questionable. With an eye on improvements, we have considered a number of alterations to M^*. However, none resulted in models better than M^* in terms of BIC score.

Those mismatches are partially due to the limitations of HLC models. Disability is a concern in both agriculture and firms. We naturally would expect h_{17} (aversion to disability risks) be connected to both h_1 (aversion to firm risks) and h_{15} (aversion to agriculture risks). But that would create a loop, which is not allowed in HLC models. Hence, there is a need to study generalizations of HLC models in the future. As mentioned in Section 2, it would also be interesting to study the impact of standard model dimensions versus effective model dimensions.

5 Probabilistic Modeling

We have so far mentioned two probabilistic models for the CoIL Challenge 2000 data, namely the HLC model M^* and the latent class model produced during latent class analysis. In this section, we will denote M^* as M_{HLC} and the latent class model as M_{LC}. For the sake of comparison, we have also used the greedy equivalence search algorithm [1] to obtain a Bayesian network model that do not contain latent variables. This model will be denoted as M_{GES}. This structure of M_{GES} is shown in Figure 2. In general, we refer to Bayesian networks that do not contain latent variables *observed BN models.*

The structure of M_{HLC} is clearly more meaningful than that of M_{LC} and M_{GES}. The structure of M_{LC} is too simplistic to be informative. The relationships encoded in M_{GES} are not as interpretable as those encoded in M_{HLC}.

How do the models fit the data? Before answering this question, we note that HLC models and observed BN models both have their pros and cons when it

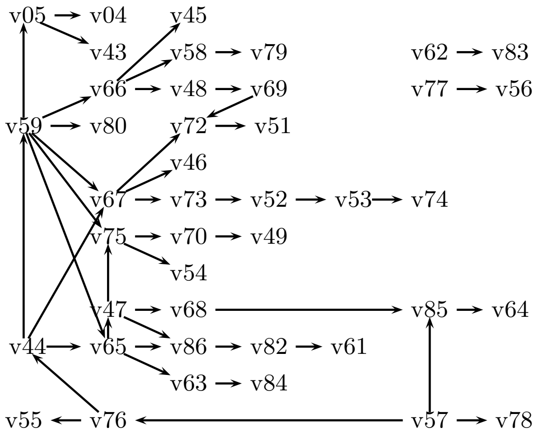

Fig. 2. Bayesian network model without latent variables

comes to represent interactions among manifest variables. The advantage of HLC models over observed BN models is that they can model high-order interactions. In M_{HLC}, latent variable h_{12} models some of the interactions among the heavy private vehicle variables; h_8 models some of the interactions among the private vehicle variables; while h_0 models some of the interactions among all manifest variables. On the other hand, observed BN models are better than HLC models in modeling details of variable interactions. In M_{GES}, the conditional probability distributions $P(v_{59}|v_{44})$ and $P(v_{67}|v_{59}, v_{44})$ contain all information about the interactions among the three variables v_{44}, v_{59}, and v_{67}.

As can be seen from the table below, the logscore of M_{HLC} on training data is slightly higher than that of M_{GES}. On the other hand, M_{GES} is less complex than M_{HLC}, and its BIC score is higher than that of M_{HLC}. In COIL Challenge 2000, there is a test set of 4,000 records. The logscore of M_{HLC} on test data is higher than that of M_{GES} and the difference is larger than that on the training data. In other words, M_{HLC} is better than M_{GES} when it comes to predicting the test data.

Model	Logscore	Complexity	BIC	Logscore
				(test data)
M_{LC}	-62328	739	-65532	-43248
M_{GES}	-49792	284	-51023	-34627
M_{HLC}	-49688	410	-51465	-34282

Because HLC models represent high-order variable interactions, M_{HLC} should perform better than M_{GES} in classification tasks. Out of the 4,000 customers in the COIL Challenge 2000 test data, 238 own mobile home policies (v_{86}). The

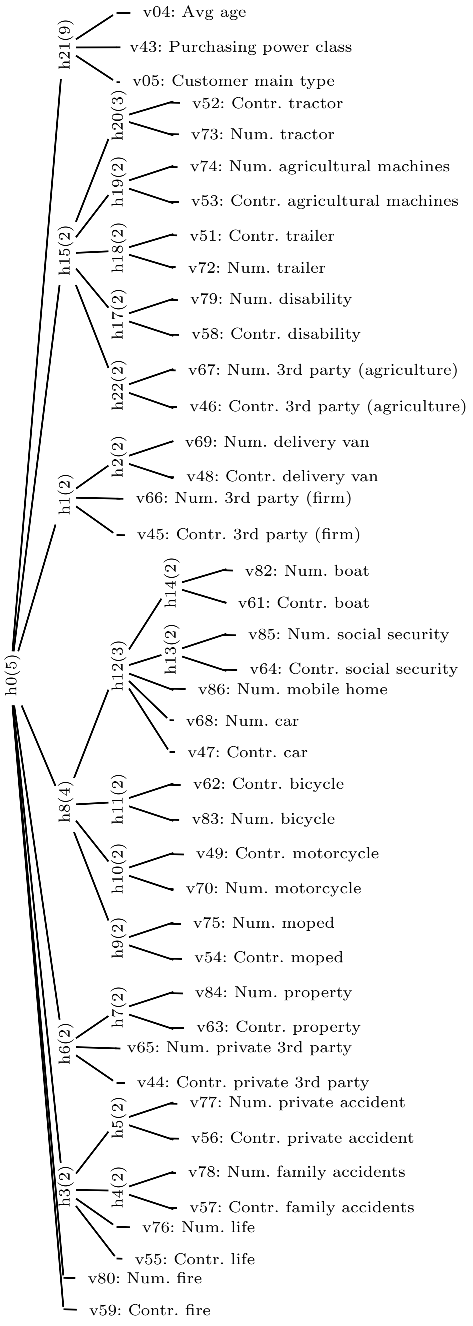

Fig. 3. Structure of Model M^*. The number next to a latent variable is the cardinality of that variable.

classification task is to identify a subset of 800 that contains as many mobile home policy owners as possible. As can be seen from the following table, M_{HLC} does perform significantly better than M_{GES}.

Model/Method	# of Mobile Home Policy Holders Identified	Hit Ratio
Random	42	17.6%
G_{GES}	83	34.9%
G_{LC}	105	44.1%
G_{HLC}	110	46.2%
CoIL 2000 Best	121	50.8%

The classification performance of M_{HLC} ranks at Number 5 among the 43 entries to the CoIL Challenge 2000 contest [5], and it is not far from the performance of the best entry. This is impressive considering that no attempt was made to minimize classification error when learning M_{HLC}. In terms of model interpretability, M_{HLC} would rank Number 1 because all the 43 entries focus on classification accuracy rather than data modeling.

6 Conclusions

Through the analysis of the CoIL Challenge 2000 data set, we have demonstrated that it is possible to infer complex and meaningful latent structures from data using HLC models.

Acknowledgements

Research on this work was supported by Hong Kong Grants Council Grant #622105. We thank Tao Chen, Yi Wang and Kin Man Poon for valuable discussions.

References

1. Chickering, D. M. (2002). Learning equivalence classes of Bayesian-network structures. *Journal of Machine Learning Research*, 2: 445-498.
2. Friedman, N. (1997). Learning belief networks in the presence of missing values and hidden variables. In *Proc. of 14th Int. Conf. on Machine Learning (ICML-97)*, 125-133.
3. Lazarsfeld, P. F., and Henry, N.W. (1968). *Latent structure analysis.* Boston: Houghton Mifflin.
4. Pearl, J. (1988). *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference* Morgan Kaufmann Publishers, Palo Alto.
5. van der Putten, P. and van Someren, M. (2004). A Bias-Variance Analysis of a Real World Learning Problem: The CoIL Challenge 2000. *Machine Learning*, Kluwer Academic Publishers, 57, 177-195.

6. Zhang, N. L. (2002). Hierarchical Latent Class models for Cluster Analysis, AAAI-02.
7. Zhang, N. L. (2004). Hierarchical latent class models for cluster analysis. *Journal of Machine Learning Research*, 5:697–723.
8. Zhang, N. L. and Kocka, T. K. (2004). Efficient Learning of Hierarchical Latent Class Models. In *Proc. of the 16th IEEE International Conference on Tools with Artificial Intelligence (ICTAI-2004)*.