BAYESIAN CLASSIFIERS WITH DISCRETE PREDICTORS

Pedro Larrañaga, Concha Bielza, Jose Luis Moreno

Computational Intelligence Group
Artificial Intelligence Department
Universidad Politécnica de Madrid

Machine Learning
Master in Data Science + Master HMDA

Outline

Discrete Bayesian network classifiers (Bielza and Larrañaga, 2014)

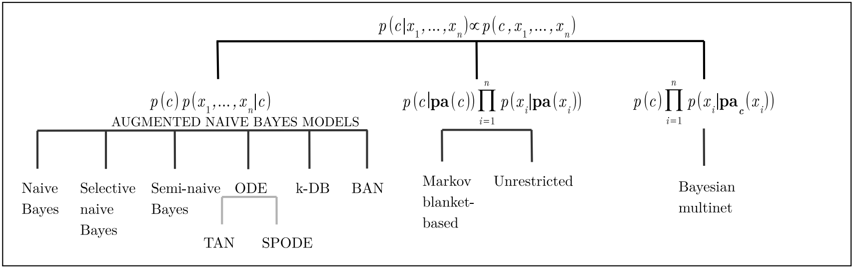
Bayes decision rule

$$p(\mathbf{x}, c) = p(c|\mathbf{pa}(c)) \prod_{i=1}^{n} p(x_i|\mathbf{pa}(x_i))$$

This method is really usefull when we have a loss function of this type. For any other, the precession is not guarantied

The Bayes decision rule (minimization of the expected loss) for a 0-1 loss function:

$$c^* = \arg\max_c p(c|\mathbf{x}) = \arg\max_c p(\mathbf{x}, c)$$



Categorization of discrete Bayesian network classifiers

Parameter Estimation

Maximun likelihood estimation

The mle estimator for $p(x_i|\mathbf{pa}(x_i))$ is given by $\frac{N_{ijk}}{N_{ij}}$

- N_{ijk} is the frequency in $\mathcal{D}$ of cases with $X_i = k$ and $\mathbf{Pa}(X_i) = j$
- N_{ij} is the frequency in $\mathcal{D}$ of cases with $\mathbf{Pa}(X_i) = j$ (i.e., $N_{ij} = \sum_{k=1}^{R_i} N_{ijk}$)

Bayesian estimation

Assuming a Dirichlet prior distribution over $(p(X_i = 1|\mathbf{Pa}(X_i) = j), ..., p(X_i = R_i|\mathbf{Pa}(X_i) = j))$ with all hyperparameters equal to α, the posterior distribution is Dirichlet with hyperparameters equal to $N_{ijk} + \alpha$, $k = 1, ..., R_i$

$p(X_i = k|\mathbf{Pa}(X_i) = j)$ is estimated by $\frac{N_{ijk}+\alpha}{N_{ij}+R_i\alpha}$ (Lindstone rule)

- Laplace estimation: $\alpha = 1$
- Schurmann-Grassberger rule: $\alpha = \frac{1}{R_i}$

Outline

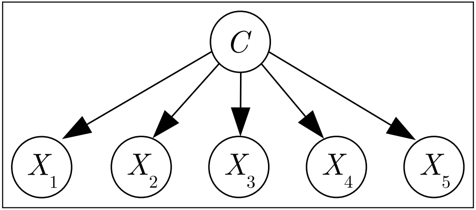
Naive Bayes as a Bayesian network

Under the assumtion of independece of conditionally probabilities, we can reduce grately the dimensionality of the problem

Naive Bayes (Minsky, 1961)

Predictor variables conditionally independent given C: $p(c|\mathbf{x}) \propto p(c) \prod_{i=1}^{n} p(x_i|c)$

$c^* = arg\max_c P(C = c) \prod_{i=1}^{n} P(X_i = x_i|C = c)$



Structure of a naïve Bayes

Decision boundary of a naive Bayes

Decision boundary = hyperplane (Minsky, 1961)

$$p(x_i|c) = p(X_i = 0|C = c)\left[\frac{p(X_i = 1|C = c)}{p(X_i = 0|C = c)}\right]^{x_i}$$

with $x_i = 0, 1$. Then, substituting this in $p(c|\mathbf{x}) \propto p(c)\prod_{i=1}^{n} p(x_i|c)$ and taking the natural log:

$$\begin{aligned}\ln p(c|\mathbf{x}) \propto \ln p(c) + \ln \prod_{i=1}^{n} p(X_i = 0|C = c) + \sum_{i=1}^{n} x_i \ln\left[\frac{p(X_i = 1|C = c)}{p(X_i = 0|C = c)}\right] \\ w_{c0} \quad = \quad \ln p(c) + \ln \prod_{i=1}^{n} p(X_i = 0|C = c) \\ w_{ci} \quad = \quad \ln\left[\frac{p(X_i = 1|C = c)}{p(X_i = 0|C = c)}\right]\end{aligned}$$

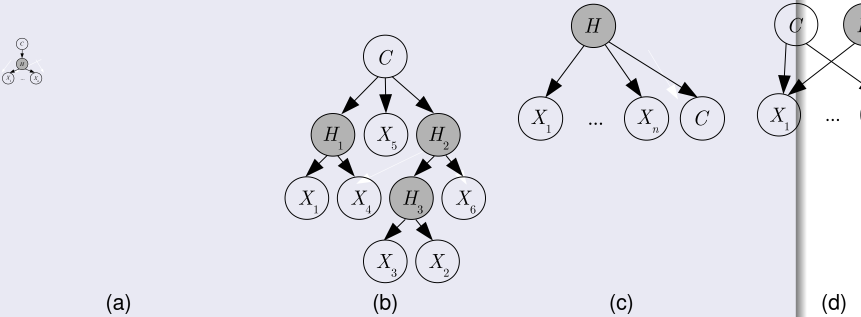
then $\ln p(c|\mathbf{x}) \propto w_{c0} + \mathbf{w}_c^T\mathbf{x}$ with $\mathbf{w}_c^T = (w_{c1}, \ldots, w_{cn})$
The decision boundary is

$$\ln p(C = 0|\mathbf{x}) - \ln p(C = 1|\mathbf{x}) = (w_{00} - w_{10}) + (\mathbf{w}_0 - \mathbf{w}_1)^T\mathbf{x} = 0$$

which defines a hyperplane

Naive Bayes con hidden variables

Violating the conditional independence assumption



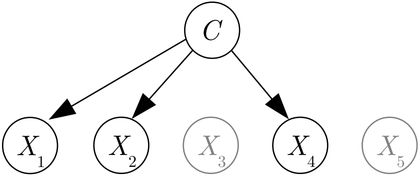
(a) Naive Bayes with a hidden variable H (Kwoh and Gilles 1996. (b) Hierarchical naive Bayes (Zhang et al., 2004; Langseth and Nielsen 2006). (c) Finite mixture model, with a hidden variable as a parent of the predictor variables and the class (Kontkanen et al., 1996). (d) Finite-mixture augmented naive Bayes (Monti and Cooper 1999)

Outline

Selective naive Bayes

Selective naive Bayes

Relevant and non-redundant predictors : $p(c|\mathbf{x}) \propto p(c|\mathbf{x}_F) = p(c)\prod_{i\in F} p(x_i|c)$
$\mathbf{X}_F$ denotes the projection of $\mathbf{X}$ onto the selected feature subset $F \subseteq \{1, 2, ..., n\}$



A selective naive Bayes structure for which $p(c|\mathbf{x}) \propto p(c)p(x_1|c)p(x_2|c)p(x_4|c)$

Filter and wrapper

- Filter: $\mathbb{I}(X_i, C)$ (Pazzani and Billsus, 1997)
- Wrapper: greedy forward (Langley and Sage, 1994), floating search (Pernkopf and O'Leary, 2003), genetic algorithms (Liu et al. 2001) and estimation of distribution algorithms (Inza et al., 2000)

Outline

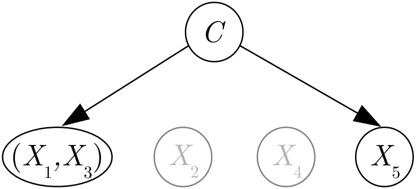
Semi-naive Bayes

Relaxing conditional independencies by Cartesian products

The new predictor variables (original ones or Cartesian products of originals) are still conditionally independent given the class variable

$$p(c|\mathbf{x}) \propto p(c) \prod_{j=1}^{K} p(\mathbf{x}_{S_j}|c),$$

where $S_j \subseteq \{1, 2, ..., n\}$ denotes the indices in the j-th feature (original or Cartesian product), $j = 1, ..., K$, $S_j \cap S_l = \varnothing$, for $j \neq l$



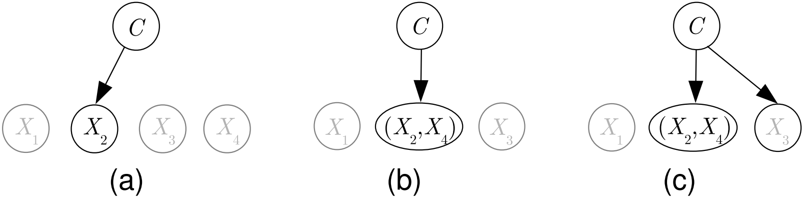
A semi-naive Bayes structure for which $p(c|\mathbf{x}) \propto p(c)p(x_1, x_3|c)p(x_5|c)$

Semi-naive Bayes

The forward sequential selection and joining (FSSJ) (Pazzani, 1996)

1. Starts from an empty structure. The accuracy is obtained by using the simple decision rule where the most likely label is assigned to all instances
2. Then the algorithm considers the best option between:
   - (a) Adding a variable not used by the current classifier as conditionally independent of the features (original or Cartesian products) used in the classifier, and
   - (b) Joining a variable not used by the current classifier with each feature (original or Cartesian products) present in the classifier

Building process (FSSJ)



(a) The selective naive Bayes with X_2 has yielded the best accuracy
(b) After building the models with these sets of predictor variables: $\{X_2, X_1\}$, $\{X_2, X_3\}$, $\{X_2, X_4\}$, $\{(X_2, X_1)\}$, $\{(X_2, X_3)\}$ and $\{(X_2, X_4)\}$, the last option is selected according to its accuracy
(c) The winner model out of $\{X_1, (X_2, X_4)\}$, $\{X_3, (X_2, X_4)\}$, $\{(X_1, X_2, X_4)\}$, and $\{(X_3, X_2, X_4)\}$. The accuracy does not improve with $\{X_1, X_3, (X_2, X_4)\}$, $\{(X_1, X_3), (X_2, X_4)\}$, and $\{X_3, (X_1, X_2, X_4)\}$, and the process stops

Outline

Tree augmented naive Bayes

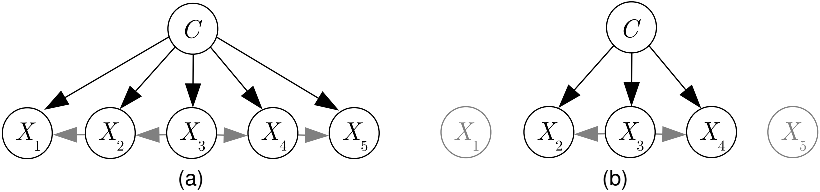
Tree augmented naive Bayes (Friedman et al., 1997)

The predictor subgraph is necessarily a tree: all predictor variables contain exactly one parent, except for one variable that has no parents, called the *root*

$$p(c|\mathbf{x}) \propto p(c)p(x_r|c) \prod_{i=1, i\neq r}^{n} p(x_i|c, x_{j(i)})$$

As we see in this one we have to compute a three order probability

where X_r denotes the root node and $\{X_{j(i)}\} = \mathbf{Pa}(X_i) \setminus C$, for any $i \neq r$



(a) A TAN structure, whose root node is X_3, for which $p(c|\mathbf{x}) \propto p(c)p(x_1|c, x_2)p(x_2|c, x_3)p(x_3|c)p(x_4|c, x_3)p(x_5|c, x_4)$. (b) Selective TAN (Blanco et al., 2005), for which $p(c|\mathbf{x}) \propto p(c)p(x_2|c, x_3)p(x_3|c)p(x_4|c, x_3)$

Learning algorithm for TAN

Tree Augmented Naive (Bayes)

Algorithm 1: Learning a TAN structure

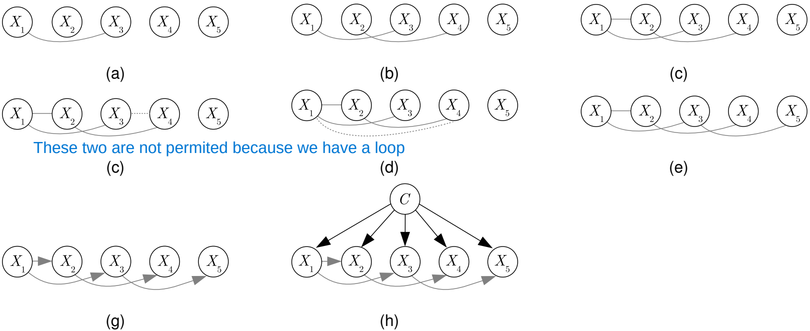
Input : A data set $\mathcal{D} = \{(\mathbf{x}^1, c^1), ..., (\mathbf{x}^N, c^N)\}$ with $\mathbf{X} = (X_1, ..., X_n)$
Output: A TAN structure

for $i < j, i, j = 1, ..., n$ **do**
  Compute $\mathbb{I}(X_i, X_j|C) = \sum_{i,j,r} p(x_i, x_j, c_r) \log_2 \frac{p(x_i,x_j|c_r)}{p(x_i|c_r)p(x_j|c_r)}$
endfor
Build a complete undirected graph where the nodes are $X_1, ..., X_n$. Annotate the weight of an edge connecting X_i and X_j by $\mathbb{I}(X_i, X_j|C)$
Build a maximum weighted spanning tree:
  Select the two edges with the heaviest weights
  while *The tree contains fewer than $n - 1$ edges* **do**
    if *They do not form a cycle with the previous edges* **then** Select the next heaviest edge
    else Reject the edge and continue
  endwhile
Transform the resulting undirected tree into a directed one by choosing a root node and setting the direction of all edges to be outward from this node
Construct a TAN structure by adding a node C and an arc from C to each X_i

This method provides the tree with the most amount of likelyhood

TAN building process

$\mathbb{I}(X_1, X_3|C) > \mathbb{I}(X_2, X_4|C) > \mathbb{I}(X_1, X_2|C) > \mathbb{I}(X_3, X_4|C) > \mathbb{I}(X_1, X_4|C) > \mathbb{I}(X_3, X_5|C) > \mathbb{I}(X_1, X_5|C) >$
$\mathbb{I}(X_2, X_3|C) > \mathbb{I}(X_2, X_5|C) > \mathbb{I}(X_4, X_5|C)$



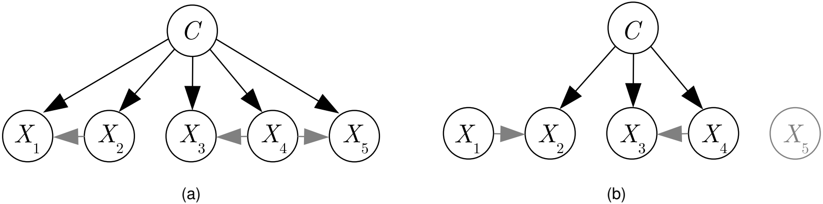
(a-c) Edges are added according to conditional mutual information quantities arranged in ascending order. (d-e) Edges $X_3 - X_4$ and $X_1 - X_4$ (dashed lines) cannot be added since they form a cycle. (f) Maximum weighted spanning tree. (g) The directed tree obtained by choosing X_1 as the root node. (h) Final TAN structure

Outline

TAN building process

Forest augmented naive Bayes (FAN) (Lucas, 2004)

- FAN: a forest –i.e., a disjoint union of trees– in the predictor subgraph, augmented with a naive Bayes. The forest is obtained using a maximum weighted spanning forest algorithm (Fredman and Tarjan, 1987)
- Selective FAN: allows the predictor variables to be optionally dependent on the class variable, that is, missing arcs from C to some X_i can be found (Ziebart et al., 2007)



(a) FAN with two root nodes X_2 and X_4: $p(c|\mathbf{x}) \propto p(c)p(x_1|c, x_2)p(x_2|c)p(x_3|c, x_4)p(x_4|c)p(x_5|c, x_4)$. (b) Selective FAN: $p(c|\mathbf{x}) \propto p(c)p(x_2|c, x_1)p(x_3|c, x_4)p(x_4|c)$

Outline

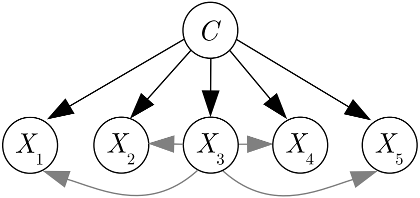
Superparent-one-dependence estimators

Superparent-one-dependence estimators (SPODE) (Keogh and Pazzani, 2002)

- One-dependence estimators (ODEs): each predictor variable is allowed to depend on at most one other predictor in addition to the class (is a particular case of a TAN model)
- SPODEs are an ODE where all predictors depend on the same predictor, called the superparent, in addition to the class

$$p(c|\mathbf{x}) \propto p(c)p(x_{sp}|c) \prod_{i=1, i \neq sp}^{n} p(x_i|c, x_{sp})$$

where X_{sp} denotes the superparent node



A SPODE structure, with X_3 as superparent, for which
$p(c|\mathbf{x}) \propto p(c)p(x_1|c, x_3)p(x_2|c, x_3)p(x_3|c)p(x_4|c, x_3)p(x_5|c, x_3)$

Superparent-one-dependence estimators

Averaged one-dependence estimator (AODE) (Webb et al., 2005)

- AODE averages the predictions of all qualified SPODEs (metaclassifier)
- 'Qualified" means including, for each instance $\mathbf{x} = (x_1, \ldots, x_{sp}, \ldots, x_n)$, only the SPODEs for which the probability estimates are accurate, that is, where the training data contain more than m cases satisfying $X_{sp} = x_{sp}$ ($m = 30$)

  $$p(c|\mathbf{x}) \propto p(c, \mathbf{x}) = \frac{1}{|\mathcal{SP}_{\mathbf{x}}^{m}|} \sum_{X_{sp} \in \mathcal{SP}_{\mathbf{x}}^{m}} p(c)p(x_{sp}|c) \prod_{i=1, i \neq sp}^{n} p(x_i|c, x_{sp})$$

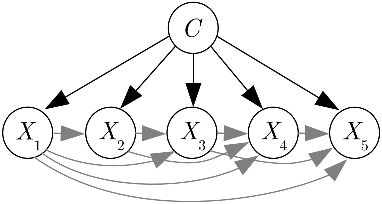
  where $\mathcal{SP}_{\mathbf{x}}^{m}$ denotes for each $\mathbf{x}$ the set of predictor variables qualified as superparents and $|\cdot|$ is its cardinality.
- AODE avoids model selection, thereby decreasing the variance component of the classifier

Outline

k-dependence Bayesian classifiers (*k*-DB) (Sahami, 1996)

k-DB

- *k*-DB allows each predictor variable to have a maximum of k parent variables apart from the class variable. Naive Bayes and TAN are particular cases
- $p(c|\mathbf{x}) \propto p(c) \prod_{i=1}^{n} p(x_i|c, x_{i_1}, ..., x_{i_k})$ where $X_{i_1}, ..., X_{i_k}$ are the parents of X_i



An example of a 3-DB structure for which
$p(c|\mathbf{x}) \propto p(c)p(x_1|c)p(x_2|c, x_1)p(x_3|c, x_1, x_2)p(x_4|c, x_1, x_2, x_3)p(x_5|c, x_1, x_3, x_4)$

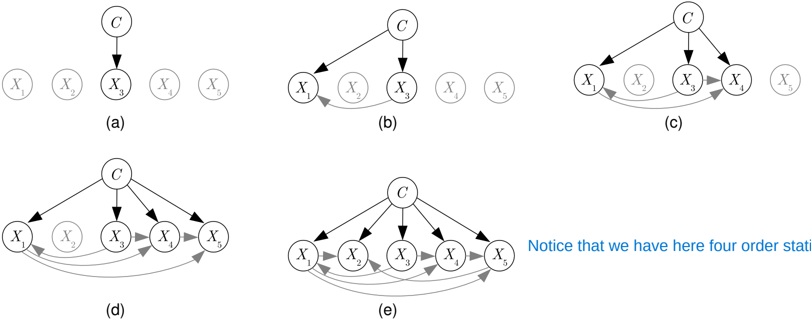
Learning a *k*-DB

- The inclusion order of the predictor variables X_i in the model is given by $\mathbb{I}(X_i, C)$, starting with the highest
- Once X_i enters the model, its parents are selected by choosing the k variables X_j in the model with the highest values of $\mathbb{I}(X_i, X_j|C)$

Building a *k*-DB with *k*=2

$$\mathbb{I}(X_3, C) > \mathbb{I}(X_1, C) > \mathbb{I}(X_4, C) > \mathbb{I}(X_5, C) > \mathbb{I}(X_2, C).$$

$$\begin{aligned}\mathbb{I}(X_3, X_4|C) &> \mathbb{I}(X_2, X_5|C) > \mathbb{I}(X_1, X_3|C) > \mathbb{I}(X_1, X_2|C) > \mathbb{I}(X_2, X_4|C) > \mathbb{I}(X_2, X_3|C)\\ &> \mathbb{I}(X_1, X_4|C) > \mathbb{I}(X_4, X_5|C) > \mathbb{I}(X_1, X_5|C) > \mathbb{I}(X_3, X_5|C)\end{aligned}$$



Notice that we have here four order statistics

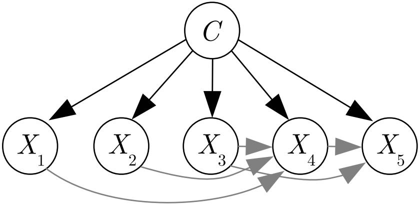
An example of *k*-DB structure learning with $k = 2$. (a-c) Variables X_3, X_1 and X_4 enter the model one by one, taking as parents the current predictor variables. (d) X_5 enters the model with parents X_1 and X_4. (e) X_2 enters the model with parents X_1 and X_5. This is the final *k*-DB structure

Outline

Bayesian network augmented naive Bayes (BAN) (Ezawa and Norton, 1996)

BAN

- Any Bayesian network structure as the predictor subgraph
- The posterior distribution is $p(c|\mathbf{x}) \propto p(c) \prod_{i=1}^{n} p(x_i|\mathbf{pa}(x_i))$



A BAN structure for which $p(c|\mathbf{x}) \propto p(c)p(x_1|c)p(x_2|c)p(x_3|c)p(x_4|c, x_1, x_2, x_3)p(x_5|c, x_3, x_4)$

Bayesian network augmented naive Bayes (BAN) (Ezawa and Norton, 1996)

Building a BAN

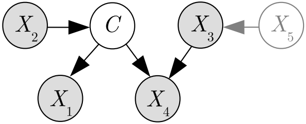
1. Ranks the n predictor variables based on $\mathbb{I}(X_i, C)$, and then it selects the minimum number of predictor variables k satisfying $\sum_{j=1}^{k} \mathbb{I}(X_j, C) \geq t_{CX} \sum_{j=1}^{n} \mathbb{I}(X_j, C)$, where $0 < t_{CX} < 1$ is the threshold
2. $\mathbb{I}(X_i, X_j|C)$ is computed for all pairs of the selected variables. The edges corresponding to the highest values are selected until a percentage t_{XX} of the overall conditional mutual information $\sum_{i<j}^{k} \mathbb{I}(X_i, X_j|C)$ is surpassed
3. Edge directionality is based on the variable ranking of the first step: higher-ranked variables point toward lower-ranked variables

Outline

Markov blanket-based Bayesian classifier (Koller and Sahami, 1996)

Markov blanket-based Bayesian classifier

- If C can have parents: $p(c|\mathbf{x}) \propto p(c|\mathbf{pa}(c)) \prod_{i=1}^{n} p(x_i|\mathbf{pa}(x_i))$
- The Markov blanket (its parents, its children and the parents of the children) of C is the only knowledge needed to predict its behavior



A Markov blanket structure for C for which $p(c|\mathbf{x}) \propto p(c|x_2)p(x_1|c)p(x_2)p(x_3)p(x_4|c, x_3)$
The Markov blanket of C is $\mathbf{MB}(C) = \{X_1, X_2, X_3, X_4\}$

Markov blanket-based Bayesian classifier (Koller and Sahami, 1996)

Building a MB(C)

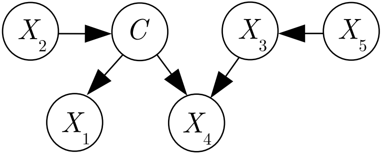
- Start from the set of all the predictor variables and eliminate a variable at each step (backward greedy strategy) until we have approximated **MB**(C)
- A feature is eliminated if it gives little or no additional information about C beyond what is subsumed by the remaining features
- Eliminates feature by feature trying to keep $p(C|\mathbf{MB}^{(t)}(C))$, the conditional probability of C given the current estimation of the Markov blanket at step t, as close to $p(C|\mathbf{X})$ as possible
- Closeness is defined by the Kullback-Leibler divergence

Outline

Unrestricted Bayesian classifiers

Unrestricted Bayesian classifiers

- Do not consider C as a special variable in the induction process
- Any existing Bayesian network structure learning algorithm can be used
- The corresponding Markov blanket of C can be used later for classification purposes



An unrestricted Bayesian network classifier structure for which $p(c|\mathbf{x}) \propto p(c|x_2)p(x_1|c)p(x_2)p(x_3)p(x_4|c, x_3)$

Outline

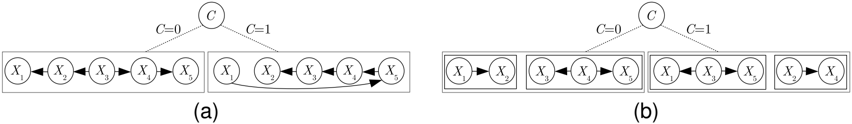
Bayesian multinets (Geiger and Heckerman, 1996)

Bayesian multinets

- Several (local) Bayesian networks associated with a subset of a partition of the domain of a variable H, called the hypothesis or distinguished variable
- Asymmetric conditional independence assertions are represented in each local network topology
- For classification problems, the distinguished variable is the class variable C

$$p(c|\mathbf{x}) \propto p(c) \prod_{i=1}^{n} p(x_i|\mathbf{pa}_c(x_i))$$

$\mathbf{Pa}_c(X_i)$ parent set of X_i in the local Bayesian network associated with $C = c$



(a) Bayesian multinet as a collection of trees:
$p(C=0|\mathbf{x}) \propto p(C=0)p(x_1|C=0,x_2)p(x_2|C=0,x_3)p(x_3|C=0)p(x_4|C=0,x_3)p(x_5|C=0,x_4)$ and
$p(C=1|\mathbf{x}) \propto p(C=1)p(x_1|C=1)p(x_2|C=1,x_3)p(x_3|C=1,x_4)p(x_4|C=1,x_5)p(x_5|C=1,x_1)$
(b) Bayesian multinet as a collection of forests:
$p(C=0|\mathbf{x}) \propto p(C=0)p(x_1|C=0)p(x_2|C=0,x_1)p(x_3|C=0,x_4)p(x_4|C=0)p(x_5|C=0,x_4)$ and
$p(C=1|\mathbf{x}) \propto p(C=1)p(x_1|C=1,x_3)p(x_2|C=1)p(x_3|C=1)p(x_4|C=1,x_2)p(x_5|C=1,x_3)$

Outline

Bayesian network based classifiers

- Provides a posterior probability for each possible value of the class
- Competitive results (accuracy, Brier, ROC) with the state of the art in supervised classifiers
- Knowledge discovery from the structure of the Bayesian network

References (i)

- C. Bielza and P. Larrañaga (2014a). Discrete Bayesian network classifiers: A survey. *ACM Computing Surveys*, 47 (1), Article 5
- K.J. Ezawa and S.W. Norton (1996). Constructing Bayesian networks to predict uncollectible telecommunications accounts. *IEEE Exper*t, 11(5), 45-51
- M.L. Fredman and R.E. Tarjan (1987). Fibonacci heaps and their uses in improved network optimization algorithms. *Journal ACM*, 34(3), 596-615
- N. Friedman, D. Geiger and M. Goldszmidt (1997). Bayesian network classifiers. *Machine Learning*, 29, 131-163
- D. Geiger and D. Heckerman (1996). Knowledge representation and inference in similarity networks and Bayesian multinets. *Artificial Intelligence*, 82, 45-74
- I. Inza, P. Larrañaga, R. Etxeberria and B. Sierra. Feature subset selection by Bayesian network-based optimization. *Artificial Intelligence*, 123(1-2), 157-184
- E.J. Keogh and M.J. Pazzani (2002). Learning the structure of augmented Bayesian classifiers. *International Journal on Artificial Intelligence Tools*, 11(4), 587-601
- D. Koller and M. Sahami (1996). Toward optimal feature selection. *Proceedings of the 13th International Conference on Machine Learning*, 284-292
- C. K. Kwoh and D. Gillies (1996). Using hidden nodes in Bayesian networks. *Artificial Intelligence*, 88, 1-38
- P. Langley and S. Sage (1994). Induction of selective Bayesian classifiers. *Proceedings of the 10th Conference on Uncertainty in Artificial Intelligence*, 399-406
- H. Langseth and T.D. Nielsen (2006). Classification using hierarchical naive Bayes models. *Machine Learning*, 63(2), 135-159

References (ii)

- J.N.K. Liu, N. L. Li and T. S. Dillon (2001). An improved naive Bayes classifier technique coupled with a novel input solution method. *IEEE Transactions on Systems, Man, and Cybernetics, Part C*, 31, 249-256
- P. Lucas (2004). Restricted Bayesian network structure learning. *Advances in Bayesian Networks*, 217-232
- M. L. Minsky (1961). Steps toward artificial intelligence. *Transactions on Institute of Radio Engineers*, 49, 8-30
- S. Monti and G. F. Cooper (1999). A Bayesian network classifier that combines a finite mixture model and a naive Bayes model. *Proceedings of the 15th Conference on Uncertainty in Artificial Intelligence*, 447-456
- M. Pazzani (1996). Constructive induction of Cartesian product attributes. *Proceedings of the Information, Statistics and Induction in Science Conference*, 66-77
- M. Pazzani and D. Billsus (1997). Learning and revising user profiles: The identification of interesting web sites. *Machine Learning*, 27, 313-331
- F. Pernkopf and P. O'Leary (2003). Floating search algorithm for structure learning of Bayesian network classifiers. *Pattern Recognition Letters*, 24, 2839-2848
- M. Sahami (1996). Learning limited dependence Bayesian classifiers. *Proceedings of the 2nd International Conference on Knowledge Discovery and Data Mining*, 335-338
- G. I. Webb and J. Boughton and Z. Wang (2005). Not so naive Bayes: Aggregating one-dependence estimators. *Machine Learning*, 58, 5-24
- N.L. Zhang, T.D. Nielsen and F.V. Jensen (2004). Latent variable discovery in classification models. *Artificial Intelligence in Medicine*, 30(3), 283-299
- B. Ziebart, A.K. Dey and J.A. Bagnell (2007). Learning selectively conditioned forest structures with applications to DBNs and classification. *Proceedings of the 23rd Conference Annual Conference on Uncertainty in Artificial Intelligence*, 458-465

BAYESIAN CLASSIFIERS WITH DISCRETE PREDICTORS

Pedro Larrañaga, Concha Bielza, Jose Luis Moreno

Computational Intelligence Group
Artificial Intelligence Department
Universidad Politécnica de Madrid

Machine Learning

Master in Data Science + Master HMDA