Chapter 3
On the Approximation of a Discrete Multivariate Probability Distribution Using the New Concept of t-Cherry Junction Tree

Edith Kovács and Tamás Szántai

Abstract Most everyday reasoning and decision making is based on uncertain premises. The premises or attributes, which we must take into consideration, are random variables, so that we often have to deal with a high dimensional discrete multivariate random vector. We are going to construct an approximation of a high dimensional probability distribution that is based on the dependence structure between the random variables and on a special clustering of the graph describing this structure. Our method uses just one-, two- and three-dimensional marginal probability distributions. We give a formula that expresses how well the constructed approximation fits to the real probability distribution. We then prove that every time there exists a probability distribution constructed this way, that fits to reality at least as well as the approximation constructed from the Chow–Liu dependence tree. In the last part we give some examples that show how efficient is our approximation in application areas like pattern recognition and feature selection.

3.1 Introduction

The goal of our paper is to approximate a high dimensional joint probability distribution using two and three-dimensional marginal distributions, only.

The idea of such kind of approximations was given by Chow and Liu [7]. In their work they construct a first order tree taking into account the mutual information gains of all pairs of random variables. They proved that their approximation is optimal in the sense of Kullback–Leibler divergence [12, 15].

In order to give an approximation that uses lower dimensional marginal probability distributions, there are many algorithms developed. Most of them first construct

Edith Kovács (✉)
Department of Mathematics, ÁVF College of Management of Budapest, Villányi út 11-13, 1114 Budapest, Hungary
e-mail: kovacs.edith@avf.hu

Tamás Szántai
Institute of Mathematics, Budapest University of Technology and Economics, Műegyetem rkp. 3, 1111 Budapest, Hungary

K. Marti et al. (eds.), *Coping with Uncertainty*, Lecture Notes in Economics and Mathematical Systems 633, DOI 10.1007/978-3-642-03735-1_3,

a Bayesian network (directed acyclic graph) see [1] and then they obtain from this in several steps a junction tree. See [11] for a good overview.

Many algorithms were developed for obtaining a Bayesian Network. These algorithms start with the construction of the Chow–Liu tree and then this graph is transformed, by adding edges and then delete the superfluous edges using conditional independence tests. The number of conditional independence tests is diminished by searching the minimal d-separating set [2, 6].

After the graphical structure is determined a number of quantitative operations have to be performed on it (see [13] and [9]).

In our paper we suppose to be known just the three-dimensional marginals of the joint probability distribution (indeed that implies that the second and first order marginals are known, too). From this information we first construct a graphical model, and then a junction tree, named t-cherry-junction tree.

The construction of our graphical model is inspired by the graphical structure, named cherry tree, and t-cherry tree introduced by Bukszár and Prékopa in [3].

We emphasize that our method does not use the construction of the Bayesian network first, we are going to use just the third order marginals and the information contents related to them, and to certain pairs of random variables involved.

The paper is organized as follows:

- The second section contains the theory, formulas, and connections between them that are used in the third section.
- The third section contains the introduction of the concept of the t-cherry-junction tree, the formula that gives the Kullback–Leibler divergence associated to the approximation introduced, and a proof that the approximation associated to the t-cherry-junction tree is at least as good as Chow–Liu's approximation is.
- The last section contains some applications of the approximation introduced in order to exhibit some advantages of this approach.

3.2 Preliminaries

3.2.1 Notations

Let $\mathbf{X} = (X_1, X_2, \ldots, X_n)^T$ be an n-dimensional random vector with the joint probability distribution

$$P\left(X_1 = x_{i_1}^1, \ldots, X_n = x_{i_n}^n\right), i_1 = 1, \ldots, m_1, \ldots, i_n = 1, \ldots, m_n.$$

For this we will use also the abbreviation

$$P(\mathbf{X}) = P(X_1, X_2, \ldots, X_n).$$

This shorter form will be applied in sums and products, too. So we will write

$$\sum_{\mathbf{X}} P(\mathbf{X}) = \sum_{i_1=1}^{m_1} \cdots \sum_{i_n=1}^{m_n} P(X_1 = x_{i_1}^1, \ldots, X_n = x_{i_n}^n) \, (= 1) \tag{3.1}$$

and for example if $H = \{j, k, l\}$ is a three element subset of the index (vertex) set $\{1, 2, \ldots, n\}$ then $\mathbf{X}_H = (X_j, X_k, X_l)^T$ and

$$\sum_{\mathbf{X}_H} P(\mathbf{X}_H) = \sum_{i_j=1}^{m_j} \sum_{i_k=1}^{m_k} \sum_{i_l=1}^{m_l} P(X_j = x_{i_j}^j, X_k = x_{i_k}^k, X_l = x_{i_l}^l) \, (= 1) \tag{3.2}$$

and

$$\prod_{\mathbf{X}_H} P(\mathbf{X}_H) = \prod_{i_j=1}^{m_j} \prod_{i_k=1}^{m_k} \prod_{i_l=1}^{m_l} P(X_j = x_{i_j}^j, X_k = x_{i_k}^k, X_l = x_{i_l}^l) \tag{3.3}$$

$P_a(\mathbf{X})$ will denote the approximating joint probability distribution of $P(\mathbf{X})$.

3.2.2 Cherry Tree and t-Cherry Tree

The cherry tree is a graph structure introduced by Bukszár and Prékopa (see [3]). A generalization of this concept, called hyper cherry tree can be found in [4] by Bukszár and Szántai. Let be given a nonempty set of vertices V.

Definition 3.1. The graph defined recursively by the following steps is called *cherry tree*:

1. Two vertices connected by an edge is the smallest cherry tree.
2. By connecting a new vertex of V, by two new edges to two existing vertices of a cherry tree, one obtains a new cherry tree.
3. Each cherry tree can be obtained from (1) by the successive application of (2).

Remark 3.1. A cherry tree is an undirected graph. We emphasize that it is not a tree.

Definition 3.2. We call *cherry*, a triplet of vertices formed from two existing vertices and a new one connected with them in step (2) of Definition 3.1.

For the cherry we will use the notation introduced in [4] by Bukszár and Szántai: $(\{l, m\}, k)$, where l and m are the existing vertices, k is the newly connected vertex and $\{k, l\}, \{k, m\}$ are the new edges.

Remark 3.2. From a set of n vertices we obtain a cherry tree with $n - 2$ cherries.

Remark 3.3. We denote by $\mathcal{H}$ the set of all cherries of the cherry tree and by ϵ the set of edges of the cherry tree.

Remark 3.4. A cherry tree is characterized by a set V of vertices, the set $\mathcal{H}$ of cherries and the set ϵ of edges. We denote a cherry-tree by $\Delta = (V, \mathcal{H}, \epsilon)$.

In our paper we need the concept of the *t*-cherry tree introduced by Bukszár and Prékopa in [3]. To get the concept of *t*-cherry tree one has to apply the more restrictive Step (2′) in Definition 3.1 instead of Step (2):

(2′) By connecting a new vertex of V by two new edges to two *connected* vertices of a cherry tree, one obtains a new cherry tree.

Definition 3.3. The graph defined recursively by (1), (2′) and (3) is called *t-cherry tree*.

Remark 3.5. A pair of adjacent vertices from the *t*-cherry tree may be used several times, for connecting new vertices to them.

3.2.3 Junction Tree

The junction tree is a very prominent and widely used structure for inference in graphical models (see [5] and [12]).

Let $X = \{X_1, \ldots, X_n\}$ be a set of random variables defined over the same probability field and $\mathbf{X} = (X_1, \ldots, X_n)^T$ an n-dimensional random vector.

Definition 3.4. A tree with the following properties is called junction tree over X:

1. To each node of the tree, a subset of X called cluster and the marginal probability distribution of these variables is associated.
2. To each edge connecting two clusters of the tree, the subset of X given by the intersection of the connected clusters and the marginal probability distribution of these variables is associated.
3. If two clusters contain a random variable, than all clusters on the path between these two clusters contain this random variable (running intersection property).
4. The union of all clusters is X.

Notations:

- $\mathcal{C}$ – the set of clusters
- C – a cluster
- $\mathbf{X}_C$ – the random vector with the random variables of C as components
- $P(\mathbf{X}_C)$ – the joint probability distribution of $\mathbf{X}_C$
- $\mathcal{S}$ – the set of separators
- S – a separator
- $\mathbf{X}_S$ – the random vector with the random variables of S as components
- $P(\mathbf{X}_S)$ – the joint probability distribution of $\mathbf{X}_S$

The junction tree provides a joint probability distribution of $\mathbf{X}$:

$$P(\mathbf{X}) = \frac{\prod\limits_{C \in \mathcal{C}} P(\mathbf{X}_C)}{\prod\limits_{S \in \mathcal{S}} P(\mathbf{X}_S)^{(\nu_S - 1)}},$$

where ν_S is the number of those clusters which contain all of the variables involved in S.

3.3 t-Cherry-Junction Tree

3.3.1 Construction of a t-Cherry-Junction Tree

Let $X = \{X_1, \ldots, X_n\}$ be a set of random variables defined over the same probability field and denote by $\mathbf{X} = (X_1, \ldots, X_n)^T$ the corresponding n-dimensional random vector. Together with the construction of the t-cherry tree over the set of vertices $V = \{1, \ldots, n\}$ we can construct a t-cherry-junction tree in the following way.

Algorithm 1. *Construction of a t-cherry-junction tree.*

1. The first cherry of the t-cherry tree identifies the first cluster of the t-cherry junction tree. The vertices of the first cherry give the indices of the random variables belonging to the first cluster.
2. Similar way each new cherry of the t-cherry tree identifies a new cluster of the t-cherry-junction tree. The separators contain pairs of random variables with indices corresponding to the connected vertices used in Step ($2'$) of Definition 3.3.

Theorem 3.1. *The t-cherry-junction tree constructed by Algorithm 1 is a junction tree.*

Proof. We can check the statements of Definition 3.4:

- The first statement of definition is obvious.
- The separator that connects two clusters contains the intersection of the two clusters. This follows from the construction of the separator sets.
- Let us suppose that there exists a variable X_m which belongs to two clusters on a path, so that on this path exist a cluster that does not contain X_m. This implies that on this past exist two neighboring cluster so that one of them contains X_m and the other one does not contain X_m. This means that in the t-cherry tree there are two cherries connected so that one of them contains the vertex m the other one does not contain the vertex m. According to the point (2) of Definition 3.1, m is a new vertex connected, but since X_m belongs already to another cluster, the vertex m belongs to another cherry, too. That is a contradiction, because that means that m is not a new vertex.

- The union of all sets associated to the clusters is X, because of the union of set of indices is V.

Remark 3.6. To each cluster $\{X_l, X_m, X_k\}$ a three-dimensional joint probability distribution $P(X_l, X_m, X_k)$ can be associated. To each separator $\{X_l, X_m\}$ a two-dimensional joint probability distribution $P(X_l, X_m)$ can be associated.

The joint probability distribution associated to the t-cherry-junction tree is a joint probability distribution over the variables $X_1, \ldots, X_n$, given by

$$P(\mathbf{X}) = \frac{\prod\limits_{\{X_l, X_m, X_k\} \in \mathcal{C}} P(X_l, X_m, X_k)}{\prod\limits_{\{X_l, X_m\} \in \mathcal{S}} P(X_l, X_m)^{\nu_{lm}-1}}, \tag{3.4}$$

where $\nu_{lm} = \#\{\{X_l, X_m\} \subset C \,|\, C \in \mathcal{C}\}$.

3.3.2 The Approximation of the Joint Distribution Over X by the Distribution Associated to a t-Cherry-Junction Tree

Chow and Liu introduced a method to approximate optimally an n-dimensional, discrete joint probability distribution by the one- and two-dimensional probability distributions using first-order dependence tree. It is shown that the procedure presented in their paper yields an approximation with minimum difference of information, in the sense of Kullback–Leibler divergence.

In this part, we first give a formula for the Kullback–Leibler divergence between the approximated distribution associated to the t-cherry-junction tree and the real joint probability distribution; we then conclude what we have to take into account to minimize the divergence. For the proof of this theorem we need a lemma:

Lemma 3.1. *In a t-cherry-junction tree for each variable $X_m \in X$:*

$$\#\{\{X_l, X_m\} \mid (\{X_l, X_m\}, X_k) \in \mathcal{C}\} = \#\{(\{X_l, X_m\}, X_k) \mid (\{X_l, X_m\}, X_k) \in \mathcal{C}\} - 1$$

Proof. Let denote $t = \#\{\{X_l, X_m\} \mid (\{X_l, X_m\}, X_k) \in \mathcal{C}\}$ for a given $X_m \in X$.

- Case $t = 0$.
  The statement is a consequence on one hand of Definition 3.4 that is the union of all sets associated to the nodes (clusters) is X, so every vertex from X, have to appear at least in one cluster. On the other hand, if X_m would be contained in two clusters than there must exist one separator set containing this vertex (point 3) in Definition 3.4), but we supposed $t = 0$.
- Case $t > 0$.
  If two clusters contain a variable X_m, than all clusters from the path between the two clusters contain X_m (running intersection property). From this results that

the clusters containing X_m are the nodes of a connected graph, and this graph is a first order tree. If this graph contains $t + 1$ clusters, than it contains t separator sets (definition of a tree in which we have separators instead of edges and clusters instead of nodes). From the definition of the junction tree (Definition 3.4) results that every separator set connecting two clusters contains the variables from the intersection of the clusters. So there are t separator sets that contain the variable X_m.

The goodness of the approximation of a probability distribution can be quantified by the Kullback–Leibler divergence between the real and the approximating probability distributions (see for example [8] or [15]). The Kullback–Leibler divergence expresses somehow the distance between two probability distributions. As smaller value it has as better the approximation is.

Theorem 3.2. *If we denote by $P_a(\mathbf{X})$ the approximating joint probability distribution associated to the t-cherry-junction tree [see formula (3.4)], then the Kullback–Leibler divergence between the real and the approximating joint probability distribution is given by:*

$$\begin{aligned} I(P(\mathbf{X}), P_a(\mathbf{X})) = -H(\mathbf{X}) - \sum_{(X_l, X_m, X_k) \in \mathcal{C}} I(X_l, X_m, X_k) \\ + \sum_{\{X_l, X_m\} \in \mathcal{S}} (\nu_{lm} - 1) I(X_l, X_m) + \sum_{k=1}^{n} H(X_k). \end{aligned} \tag{3.5}$$

Proof.

$$\begin{aligned} I(P(\mathbf{X}), P_a(\mathbf{X})) &= \sum_{\mathbf{X}} P(\mathbf{X}) \log_2 \frac{P(\mathbf{X})}{P_a(\mathbf{X})} \\ &= \sum_{\mathbf{X}} P(\mathbf{X}) \log_2 P(\mathbf{X}) - \sum_{\mathbf{X}} P(\mathbf{X}) \log_2 P_a(\mathbf{X}) \\ &= -H(\mathbf{X}) - \sum_{\mathbf{X}} P(\mathbf{X}) \log_2 \frac{\prod_{\{X_l, X_m, X_k\} \in \mathcal{C}} P(X_l, X_m, X_k)}{\prod_{\{X_l, X_m\} \in \mathcal{S}} P(X_l, X_m)^{\nu_{lm} - 1}} \\ &= -H(\mathbf{X}) - \sum_{\mathbf{X}} P(\mathbf{X}) \Bigg[\log_2 \prod_{\{X_l, X_m, X_k\} \in \mathcal{C}} P(X_l, X_m, X_k) \\ &\qquad - \log_2 \prod_{\{X_l, X_m\} \in \mathcal{S}} P(X_l, X_m)^{\nu_{lm} - 1} \Bigg] \\ &= -H(\mathbf{X}) - \sum_{\mathbf{X}} P(\mathbf{X}) \log_2 \prod_{\{X_l, X_m, X_k\} \in \mathcal{C}} P(X_l, X_m, X_k) \\ &\quad + \sum_{\mathbf{X}} P(\mathbf{X}) \log_2 \prod_{\{X_l, X_m\} \in \mathcal{S}} P(X_l, X_m)^{\nu_{lm} - 1} \end{aligned}$$

From Definition 3.4 follows that the union of the clusters of the junction tree is the set X. From the Lemma 3.1 we know that each vertex appears once more in clusters than in separator sets. So by adding and subtracting the sum

$$\sum_{\mathbf{X}} P(\mathbf{X}) \log_2 \left[\prod_{\{X_l, X_m, X_k\} \in \mathcal{C}} P(X_l) P(X_m) P(X_k) \right]$$

we obtain the following:

$$\begin{aligned}
&I(P(\mathbf{X}), P_a(\mathbf{X})) \\
&= -H(\mathbf{X}) - \sum_{\mathbf{X}} P(\mathbf{X}) \log_2 \frac{\prod\limits_{\{X_l, X_m, X_k\} \in \mathcal{C}} P(X_l, X_m, X_k)}{\prod\limits_{\{X_l, X_m, X_k\} \in \mathcal{C}} P(X_l) P(X_m) P(X_k)} \\
&\quad + \sum_{\mathbf{X}} P(\mathbf{X}) \log_2 \frac{\prod\limits_{\{X_l, X_m\} \in \mathcal{S}} P(X_l, X_m)^{\nu_{lm}-1}}{\prod\limits_{\{X_l, X_m\} \in \mathcal{S}} [P(X_l) P(X_m)]^{\nu_{lm}-1}} - \sum_{\mathbf{X}} P(\mathbf{X}) \log_2 \prod_{X_k \in X} P(X_k) \\
&= -H(\mathbf{X}) - \sum_{\mathbf{X}} P(\mathbf{X}) \sum_{\{X_l, X_m, X_k\} \in \mathcal{C}} \log_2 \frac{P(X_l, X_m, X_k)}{P(X_l) P(X_m) P(X_k)} \\
&\quad + \sum_{\mathbf{X}} P(\mathbf{X}) \sum_{\{X_l, X_m\} \in \mathcal{S}} (\nu_{lm} - 1) \log_2 \frac{P(X_l, X_m)}{P(X_l) P(X_m)} + \sum_{X_k \in X} H(X_k).
\end{aligned}$$

Since (X_l, X_m, X_k) and (X_l, X_m) are components of the random vector $\mathbf{X}$, we have the relations:

$$\begin{aligned}
&\sum_{\mathbf{X}} P(\mathbf{X}) \sum_{\{X_l, X_m, X_k\} \in \mathcal{C}} \log_2 \frac{P(X_l, X_m, X_k)}{P(X_l) P(X_m) P(X_k)} \\
&\quad = \sum_{\{X_l, X_m, X_k\} \in \mathcal{C}} \sum_{(X_l, X_m, X_k)^T} P(X_l, X_m, X_k) \log_2 \frac{P(X_l, X_m, X_k)}{P(X_l) P(X_m) P(X_k)}
\end{aligned}$$

and

$$\begin{aligned}
&\sum_{\mathbf{X}} P(\mathbf{X}) \sum_{\{X_l, X_m\} \in \mathcal{S}} (\nu_{lm} - 1) \log_2 \frac{P(X_l, X_m)}{P(X_l) P(X_m)} \\
&\quad = \sum_{\{X_l, X_m\} \in \mathcal{S}} \sum_{(X_l, X_m)^T} (\nu_{lm} - 1) P(X_l, X_m) \log_2 \frac{P(X_l, X_m)}{P(X_l) P(X_m)}.
\end{aligned}$$

Taking into account these relations and applying the notion of the mutual information content for two and three variables:

$$I(X_l, X_m) = \sum_{(X_l, X_m)^T} P(X_l, X_m) \log_2 \frac{P(X_l, X_m)}{P(X_l)P(X_m)},$$

$$I(X_l, X_m, X_k) = \sum_{(X_l, X_m, X_k)^T} P(X_l, X_m, X_k) \log_2 \frac{P(X_l, X_m, X_k)}{P(X_l)P(X_m)P(X_k)}$$

we obtain (3.5) and the statement of the theorem has been proved.

Observation 1. We can observe that for minimizing the Kullback–Leibler divergence between the real probability distribution and the approximation obtained from the t-cherry-junction tree we have to maximize the difference between the sum of the mutual divergence of the clusters and the sum of the mutual divergence of the separators denoted by S:

$$S = \sum_{\{X_l, X_m, X_k\} \in \mathcal{C}} I(X_l, X_m, X_k) - \sum_{\{X_l, X_m\} \in \mathcal{S}} (v_{lm} - 1) I(X_l, X_m)$$

Observation 2. If we wish to compare two approximations of a joint probability distribution associated to two different t-cherry-junction trees, we have just to:

- Sum the information contents of the clusters
- Sum the information contents of the separators
- Make the difference between them
- Claim a t-junction-cherry tree be better than another one, if it produces greater value of S

3.3.3 The Relation Between the Approximations Associated to the First-Order Dependence Tree and t-Cherry-Junction Tree

A natural question is: can the t-cherry-junction tree give better approximation than the first order tree given by Chow and Liu does? Let us remind that Chow and Liu introduced a method for finding an optimal first order tree that minimizes the Kullback–Leibler divergence.

If $pa(j)$ denotes the parent node of j, the joint probability distribution associated to the Chow–Liu first order dependence tree is given as follows:

$$P_{Ch-L}(\mathbf{X}) = \prod_{i=1}^{n} P(X_{m_i} | X_{pa(m_i)}),$$

where $\{m_1, \ldots, m_n\}$ is a permutation of the numbers $1, 2, \ldots, n$ and if $pa(j)$ is the empty set, then by definition $P(X_j | X_{pa(j)}) = P(X_j)$.

In the following Algorithm 2 we will show how one can construct a t-cherry-junction tree from a Chow–Liu first order dependence tree. After this in Theorem 3.3 we prove that the t-cherry-junction tree constructed this way gives at least as good approximation of the real probability distribution as the Chow–Liu first order dependence tree does.

Algorithm 2. *The construction of a t-cherry-junction tree from a Chow–Liu first order dependence tree.*

Let us regard the spanning tree behind the Chow–Liu first order dependence tree. It is sufficient to give an algorithm for constructing a t-cherry tree from this spanning tree. Then by Algorithm 1 we can assign a t-cherry-junction tree to this t-cherry tree:

1. The first cherry of the t-cherry tree let be defined by any three vertices of the spanning tree which are connected by two edges.
2. We add a new cherry to the t-cherry tree by taking a new vertex of the spanning tree adjacent to the so far constructed t-cherry tree.
3. We repeat step (2) till all vertices from the spanning tree become included in the t-cherry tree.

Theorem 3.3. *If* $P_{Ch-L}(\mathbf{X}) = \prod_{i=1}^{n} P(X_{m_i}|X_{pa(m_i)})$ *is the approximation associated to the Chow–Liu first order dependence tree there always exists a t-cherry-junction tree with associated probability distribution* $P_{t-ch}(\mathbf{X})$ *that approximates* $P(\mathbf{X})$ *at least as well as* $P_{Ch-L}(\mathbf{X})$ *does.*

Proof. We construct the t-cherry-junction tree from the Chow–Liu first order dependence tree using Algorithm 2.

The Kullback–Leibler divergence formally looks like it is given in (3.5).

In the case of the approximation obtained by the Chow–Liu method, the Kullback–Leibler divergence is given as follows (see [7]):

$$I(P(\mathbf{X}), P_{Ch-L}(\mathbf{X})) = -H(\mathbf{X}) - \sum_{i=1}^{n} I(X_{m_i}, X_{pa(m_i)}) + \sum_{i=1}^{n} H(X_i). \tag{3.6}$$

Because the first and last terms of the Kullback–Leibler divergences are the same in the case of the two approximations (3.5) and (3.6), we denote the sums that we have to compare by S_{Ch-L} in the case of Chow–Liu's approximation and S_{t-ch} in the case of t-cherry-junction tree approximation:

$$S_{Ch-L} = \sum_{X_{m_i} \in X} I(X_{m_i}, X_{pa(m_i)}) \tag{3.7}$$

and

$$S_{t-ch} = \sum_{\{X_l, X_m, X_k\} \in \mathcal{C}} I(X_l, X_m, X_k) - \sum_{\{X_l, X_m\} \in \mathcal{S}} (\nu_{lm} - 1) I(X_l, X_m) \tag{3.8}$$

In the case of formula (3.7) we can apply the formula

$$I(X,Y) = H(X) - H(X|Y)$$

(see [8], Formula (2.43) on p. 16), while in the case of formula (3.8) we can apply

$$I(X,Y,Z) = H(Z) + I(X,Y) - H(Z|X,Y)$$

which easily can be derived from formulae given in book [8].

So from (3.7) we get

$$\begin{aligned} S_{Ch-L} &= \sum_{X_{m_i} \in X} \left[H(X_{m_i}) - H(X_{m_i}|X_{pa(m_i)}) \right] \\ &= \sum_{X_{m_i} \in X} H(X_{m_i}) - \sum_{X_{m_i} \in X} H(X_{m_i}|X_{pa(m_i)}) \end{aligned} \tag{3.9}$$

and from (3.8) we get

$$\begin{aligned} S_{t-ch} &= \sum_{\{X_l, X_m, X_k\} \in \mathcal{C}} [H(X_k) + I(X_l, X_m) - H(X_k|X_l, X_m)] \\ &\quad - \sum_{\{X_l, X_m\} \in \mathcal{S}} (\nu_{lm} - 1) I(X_l, X_m) \\ &= \sum_{X_k \in X} H(X_k) - \sum_{\{X_l, X_m, X_k\} \in \mathcal{C}} H(X_k|X_l, X_m) \end{aligned} \tag{3.10}$$

From (3.9) and (3.10) we conclude that we have to compare only

$$\sum_{X_{m_i} \in X} H(X_{m_i}|X_{pa(m_i)})$$

and

$$\sum_{\{X_l, X_m, X_k\} \in \mathcal{C}} H(X_k|X_l, X_m)$$

To each X_k corresponds an X_{m_i} because these both are the vertices of X and each of them appears exactly once. The edge connecting X_{m_i} and $X_{pa(m_i)}$ is contained in the t-cherry tree as it was constructed according to Step (2) in the proof of Theorem 3.3. From this we can conclude that for each X_k contained in a cluster $\{X_l, X_m, X_k\}$ one of X_l and X_m is $X_{pa(k)}$. Now we can apply the inequality (see book [8], Formula (2.130) on p. 36):

$$H(X_k|X_i, X_j) \leq H(X_k|X_i)$$

with equality just in case X_k is conditionally independent of X_j. From this it is obvious that $S_{t-ch} \geq S_{Ch-L}$, and indeed the relation between the Kullback–Leibler

divergences of the two approximations is:

$$I(P(\mathbf{X}), P_{t-ch}(\mathbf{X})) \leq I(P(\mathbf{X}), P_{Ch-L}(\mathbf{X})).$$

This proves the statement of the theorem.

Observation 1. If the underlying dependence structure is a first order dependence tree than we got equality between the two divergences. (Because of the conditional independences that take place between the unlinked vertices of the first order dependence tree).

3.4 Some Practical Results of Our Approximation and Discussions

This section consists of three parts. In the first part we consider two different approximations of a given eight-dimensional discrete joint probability distribution:

- The approximation given by the Chow–Liu method
- The approximation corresponding to the t-cherry tree constructed by the algorithm given in the proof of Theorem 3.3

In the second part of this section we use the approximations to make some prediction, when the values taken by two out of eight random variables are known. From this information we are going to recognize the most probable values of the remaining six random variables. In the third part we use the influence diagram underlying the t-cherry-junction tree to make a feature selection.

Two Approximations of an Eight-Dimensional Discrete Probability Distribution

First we construct an eight-dimensional discrete probability distribution in the following way. The one-dimensional marginal distributions are generated randomly. Then the so called North-West corner algorithm was applied to determine an initial feasible solution of an eight-dimensional transportation problem, where the quantities to be transported were the marginal probability values. This way the "transported probabilities" are concentrated on 44 different directions, i.e. the constructed eight-dimensional discrete probability distribution is concentrated on 44 eight-dimensional vector instead of the possible 2,116,800. An other advantage is that this way we can get as high as possible positive correlations between the components of the random vector $\mathbf{X}$. For having also high negative correlations, we combined the North-West corners with South-West corners.

Now we suppose that only the two and three-dimensional marginals of the eight-dimensional joint probability distribution are known. Using these we are going to construct a probability distribution associated to the t-cherry-junction tree. First we construct the Chow–Liu spanning tree and then transform the correspondent Chow–Liu dependence tree in a t-cherry junction tree using the Algorithm 2.

Table 3.1 The mutual information gains in decreasing order for the construction of the Chow–Liu spanning tree

Index pair	Information gain
58	**1.660755**
18	**1.644658**
28	**1.641684**
25	1.611500
15	1.605311
56	**1.593626**
17	**1.558543**
47	**1.556713**
16	1.436039
14	1.423012
68	1.421927
26	1.416225
78	1.367303
13	**1.334538**

Table 3.2 The three variable mutual information contents ordered in decreasing way. The boldfaced ones are used in the *t*-cherry-junction tree

Index triplet	Information content
158	**3.615752**
156	**3.431799**
258	**3.414463**
168	3.384689
568	3.357803
128	3.322943
125	3.321265
256	3.278311
147	**3.276440**
178	**3.270665**
157	3.197210
578	3.167661
268	3.151382
148	3.137220
135	**3.120396**

The first step is to calculate the mutual information contents of every pair and triplet of random variables. We then order them in descending way.

The Chow–Liu tree is constructed by a greedy algorithm from the two-dimensional information gains. In Table 3.1 the ordered mutual information gains are given. The mutual information gains used by Chow–Liu's method are in boldface. In Fig. 3.1 the Chow–Liu tree can be seen.

The joint probability distribution associated to the constructed Chow–Liu dependence tree is:

Table 3.3 The two variable mutual information contents ordered in decreasing way. The boldfaced ones are used in the *t*-cherry-junction tree

Index pair	Information content
58	**1.660755**
18	**1.644658**
28	1.641684
25	1.611500
15	**1.605311**
56	1.593626
17	**1.558543**

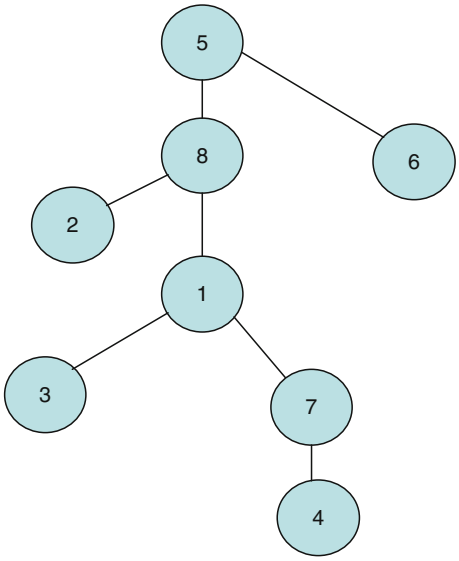

Fig. 3.1 The Chow–Liu spanning tree

$$
\begin{aligned}
&P_{Ch-L}(\mathbf{X}) \\
&\quad = P(X_5)P(X_8|X_5)P(X_2|X_8)P(X_6|X_5)P(X_1|X_8)P(X_3|X_1)P(X_7|X_1)P(X_4|X_7) \\
&\quad = \frac{P(X_5X_8)P(X_2X_8)P(X_5X_6)P(X_1X_8)P(X_1X_3)P(X_1X_7)P(X_4X_7)}{P(X_8)P(X_5)P(X_8)P(X_1)P(X_1)P(X_7)}.
\end{aligned}
$$

The Kullback–Leibler divergence corresponding to the divergence between the real probability distribution and the approximation given by Chow–Liu method can be calculated as follows:

$$
\begin{aligned}
&I(P(\mathbf{X}), P_{Ch-L}(\mathbf{X})) \\
&\quad = \sum_{i=1}^{8} H(X_i) - H(\mathbf{X}) - \big[I(X_5, X_8) + I(X_2, X_8) + I(X_5, X_6) + I(X_1, X_8) \\
&\qquad + I(X_1, X_3) + I(X_1, X_7) + I(X_4, X_7)\big] \\
&\quad = 16.908113 - 4.634363 - 10.990524 = 1.283226.
\end{aligned}
$$

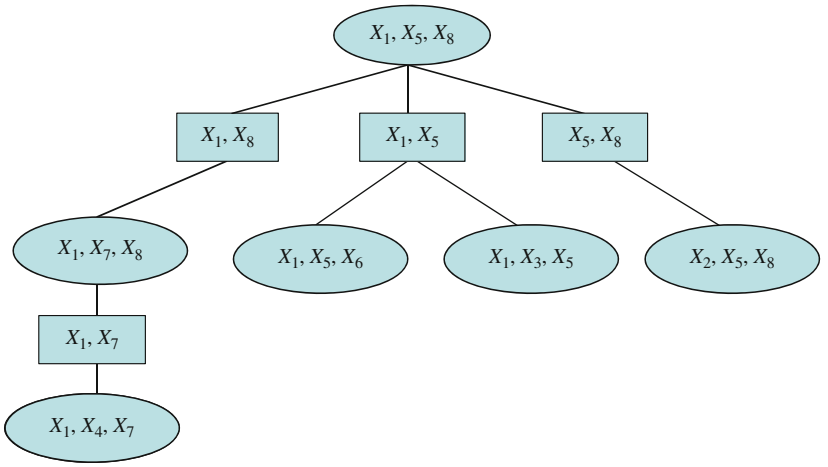

Fig. 3.2 The t-cherry-junction tree

The t-cherry-junction tree constructed by Algorithm 2 can be seen in Fig. 3.2. The three and the two variable mutual information contents corresponding to the clusters and separators of the t-cherry-junction tree are given in bold face in Tables 3.2 and 3.3

The joint probability distribution associated to the constructed t-cherry-junction tree is:

$$P_{t-ch}(\mathbf{X}) = \frac{P(X_1X_5X_8)P(X_1X_7X_8)P(X_1X_5X_6)P(X_1X_3X_5)P(X_2X_5X_8)P(X_1X_4X_7)}{P(X_1X_8)P(X_1X_5)P(X_1X_5)P(X_5X_8)P(X_1X_7)}.$$

The Kullback–Leibler divergence between the real probability distribution and the approximation associated to the t-cherry-junction tree is:

$$\begin{aligned} &I(P(\mathbf{X}), P_{t-ch}(\mathbf{X})) \\ &\quad = \sum_{i=1}^{8} H(X_i) - H(\mathbf{X}) - \big[\, I(X_1, X_5, X_8) + I(X_1, X_7, X_8) + I(X_1, X_5, X_6) \\ &\qquad + I(X_1, X_3, X_5) + I(X_2, X_5, X_8) + I(X_1, X_4, X_7) - I(X_1, X_8) \\ &\qquad - I(X_1, X_5) - I(X_1, X_5) - I(X_5, X_8) - I(X_1, X_7)\big] \\ &\quad = 16.908113 - 4.634363 - 20.129510 + 8.074578 = 0.218818. \end{aligned}$$

The real eight-dimensional distribution has 44 different vectors with probabilities different from 0. The Chow–Liu approximation has 453 vectors; the t-cherry-junction tree approximation has 93 vectors with probabilities different from zero. From the two Kullback–Leibler divergences calculated above we can observe easily that the approximation associated to the t-cherry-junction tree is much better ("closer" to the reality) than the approximation constructed from the Chow–Liu tree.

Table 3.4 Comparison of the predicted values with the real ones in the case of the two different approximations

The most probable vectors predicted by Ch–L approx.							$P(\mathbf{X})$	$P_a(\mathbf{X})$	The most probable vectors in the reality							$P(\mathbf{X})$
7	5	1	7	1	2	7	1	0.003487	0.003443	7	5	1	7	1	2	7	1	0.003487
6	5	1	7	1	2	7	3	0.006802	0.005149	6	5	1	7	1	2	7	3	0.006802
3	3	3	**4**	4	5	**3**	5	0.000000	0.001696	3	3	3	5	4	5	5	5	0.009853
2	2	3	4	5	5	3	7	0.011576	0.003718	2	2	3	4	5	5	3	7	0.011576
4	4	3	6	**2**	**3**	6	4	0.000000	0.004076	4	4	3	6	3	4	6	4	0.012450
3	2	3	4	4	5	3	6	0.019070	0.008043	3	2	3	4	4	5	3	6	0.019070
8	5	1	7	1	2	7	1	0.019399	0.019155	8	5	1	7	1	2	7	1	0.019399
3	2	3	4	**5**	5	3	7	0.010207	0.006777	3	2	3	4	4	5	3	7	0.025312
7	5	1	7	1	2	7	2	0.027852	0.027503	7	5	1	7	1	2	7	2	0.027852
9	5	1	7	1	2	8	1	0.034906	0.037489	9	5	1	7	1	2	8	1	0.034906
4	3	3	6	4	**5**	6	5	0.000000	0.033668	4	3	3	6	4	4	6	5	0.044913
5	4	3	7	**2**	**3**	7	4	0.000000	0.030836	5	4	3	7	3	4	7	4	0.049737
1	2	4	**1**	5	5	**2**	7	0.027323	0.033325	1	2	4	2	5	5	3	7	0.054170
6	5	**3**	7	1	2	7	2	0.000000	0.001522	6	5	1	7	1	2	7	2	0.128273
6	4	**2**	7	**3**	3	7	4	0.050302	0.009993	6	4	1	7	2	3	7	4	0.152472
The most probable vectors predicted by t–ch approx.							$P(\mathbf{X})$	$P_a(\mathbf{X})$	The most probable vectors in the reality							$P(\mathbf{X})$
7	5	1	7	1	2	7	1	0.003487	0.003487	7	5	1	7	1	2	7	1	0.003487
6	5	1	7	1	2	7	3	0.006802	0.006802	6	5	1	7	1	2	7	3	0.006802
3	3	3	5	4	5	5	5	0.009853	0.009497	3	3	3	5	4	5	5	5	0.009853
2	2	3	4	5	5	3	7	0.011576	0.009733	2	2	3	4	5	5	3	7	0.011576
4	4	3	6	3	4	6	4	0.012450	0.009917	4	4	3	6	3	4	6	4	0.012450
3	2	3	**5**	4	5	**4**	6	0.012682	0.018480	3	2	3	4	4	5	3	6	0.019070
8	5	1	7	1	2	7	1	0.019399	0.019399	8	5	1	7	1	2	7	1	0.019399
3	2	3	4	4	5	3	7	0.025312	0.020359	3	2	3	4	4	5	3	7	0.025312
7	5	1	7	1	2	7	2	0.027852	0.027852	7	5	1	7	1	2	7	2	0.027852
9	5	1	7	1	2	8	1	0.034906	0.034906	9	5	1	7	1	2	8	1	0.034906
4	3	3	6	4	4	6	5	0.044913	0.037797	4	3	3	6	4	4	6	5	0.044913
5	4	3	7	3	4	7	4	0.049737	0.058277	5	4	3	7	3	4	7	4	0.049737
1	2	4	2	5	5	3	7	0.054170	0.044752	1	2	4	2	5	5	3	7	0.054170
6	5	1	7	1	2	7	2	0.128273	0.128273	6	5	1	7	1	2	7	2	0.128273
6	4	1	7	2	3	7	4	0.152472	0.154778	6	4	1	7	2	3	7	4	0.152472

Application for Pattern Recognition

In this part we are testing our approximations for the following pattern recognition problem. We suppose that the values of X_1 and X_8 are known. For these given values we want to predict the most probable values of the other six random variables. In Table 3.4 one can see these predictions made with the help of the approximation.

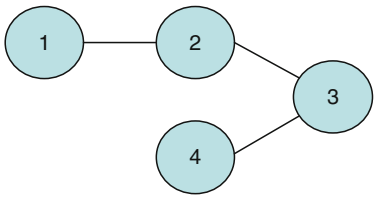

Fig. 3.3 A possible dependence diagram of four random variables

In the left side of the table the predicted values of the random variables $X_2, \ldots, X_7$ are given for the two different approximations. In the right side of the table the most probable values of the same random variables are given according to the real probability distribution (the same in the case of both approximations). The rows of the table are in ascending order according to the real probabilities. As the wrong predicted values are typed in boldface one easily can find them for each approximation. Let us observe that as long as the number of the wrong predicted values is 13 in the case of the Chow–Liu approximation, the same number equals only 2 in the case of the new t-cherry-junction tree approximation.

Feature Selection: Forecasting the Values of a Random Variable Which Depends on Many Others

The main idea of feature selection is to choose a subset of input random variables by eliminating features with little or no predictive information. In supervised learning the feature selection is useful when the main goal is to find feature subset that produces higher classification accuracy.

In practice many times we have a lot of attributes (random variables) that depend more or less on each other. The problem is how to select a few of them to make a good forecast of the variable we are interested in. The pairwise mutual information contents are not sufficient to make such a decision. To highlight this let us consider the following example. If we have four random variables with the relations between their pairwise mutual information contents: $I(X_2, X_3) > I(X_1, X_3) > I(X_3, X_4)$, and want to take into account only two random variables to forecast the values of X_3, we would use X_2 and X_1. But if we have the dependence diagram (see Fig. 3.3) we should decide in an other way. As X_1 influences X_3 only through X_2 we should rather use X_2 and X_4 for the forecast of the values of X_3. To solve such problems we give a method that uses the t-cherry-junction tree. If we are interested in forecasting a random variable X_i, we have to select from the cherry tree the clusters that contain X_i. We obtain in this way a sub junction tree. This results immediately from the properties of the junction tree. We have to take into consideration, just the variables that belong to this sub junction tree.

If in our earlier numerical example we are interested in the forecast of X_8 we select from the t-cherry-junction tree the clusters that contain X_8. From Fig. 3.2 these are (X_1, X_5, X_8), (X_1, X_7, X_8) and (X_2, X_5, X_8). Now we can conclude that

for our purpose it is important to know the joint probability distribution of the random vector $(X_1, X_2, X_5, X_7, X_8)^T$. This can be obtained as a marginal distribution of the distribution associated to the cherry tree, which can be also expressed as:

$$P(X_1, X_2, X_5, X_7, X_8) = \frac{P(X_1, X_5, X_8)P(X_1, X_7, X_8)P(X_2, X_5, X_8)}{P(X_1, X_8)P(X_5, X_8)}.$$

Now to test our method we calculate the value of the conditional entropy $H(X_8 \mid X_1, X_2, X_5, X_7) = 0.214946$ from the *real* probability distribution. If we choose another random vector $(X_1, X_2, X_5, X_7, X_8)^T$, then we get $H(X_8 \mid X_1, X_2, X_5, X_6) = 0.235572$.

It is interesting to see that $I(X_1, X_2, X_5, X_6, X_8) = 7.210594$ is greater than $I(X_1, X_2, X_5, X_7, X_8) = 7.087809$.

Acknowledgements This work was partly supported by the grant No. T047340 of the Hungarian National Grant Office (OTKA).

References

1. Acid, S., Campos, L.M.: BENEDICT: An algorithm for learning probabilistic belief networks. In: Sixth International Conference IPMU 1996, 979–984 (1996)
2. Acid, S., Campos, L.M.: An algorithm for finding minimum d-separating sets in belief networks. In: Proceedings of the 12th Conference on Uncertainty in Artificial Intelligence, pp. 3–10. Morgan Kaufmann, San Mateo (1996)
3. Bukszár, J., Prékopa, A.: Probability bounds with cherry trees. Math. Oper. Res. **26**, 174–192 (2001)
4. Bukszár, J., Szántai, T.: Probability bounds given by hypercherry trees. Optim. Methods Software **17**, 409–422 (2002)
5. Castillo, E., Gutierrez, J., Hadi, A.: Expert Systems and Probabilistic Network Models. Springer, Berlin (1997)
6. Cheng, J., Bell, D.A., Liu, W.: An algorithm for Bayesian belief network construction from data. In: Proceedings of AI&StAT'97, 83–90 (1997)
7. Chow, C.K., Liu, C.N.: Approximating discrete probability distribution with dependence trees. IEEE Trans. Inform. Theory **14**, 462–467 (1968)
8. Cover, T.M., Thomas, J.A.: Elements of Information Theory. Wiley, New York (1991)
9. Cowell, R.G., Dawid, A.Ph., Lauritzen, S.L., Spiegelhalter, D.J.: Probabilistic Networks and Expert Systems. Statistics for Engineering and Information Science. Springer, Berlin (1999)
10. Csiszar, I.: I-divergence geometry of probability distributions and minimization problems. Ann. Probab. **3**, 146–158 (1975)
11. Huang, C., Darwiche, A.: Inference in belief networks: A procedural guide. Int. J. Approx. Reason. **15**(3), 225–263 (1996)
12. Hutter, F., Ng, B., Dearden, R.: Incremental thin junction trees for dynamic Bayesian networks. Technical report, TR-AIDA-04-01, Intellectics Group, Darmstadt University of Technology, Germany, 2004. Preliminary version at http://www.fhutter.de/itjt.pdf
13. Jensen, F.V., Lauritzen, S.L., Olesen, K.: Bayesian updating in casual probabilistic networks by local computations. Comput. Stat. Q. **4** 269–282 (1990)
14. Jensen, F.V., Nielsen, T.D.: Bayesian networks and decision graphs, 2nd edn. Information Science and Statistics. Springer, New York (2007)
15. Kullback, S.: Information Theory and Statistics. Wiley, New York (1959)