DTU Library

Discrete Discriminant analysis based on tree-structured graphical models

Perez de la Cruz, Gonzalo; Eslava, Guillermina

Publication date:
2016

Document Version
Publisher's PDF, also known as Version of record

Citation (APA):
Perez de la Cruz, G., & Eslava, G. (2016). *Discrete Discriminant analysis based on tree-structured graphical models*. Technical University of Denmark. DTU Compute Technical Report-2016 No. 5

Discrete Discriminant analysis based on tree-structured graphical models

Gonzalo Perez-de-la-Cruz · Guillermina Eslava-Gomez

DTU Compute Technical Report-2016-05. ISSN: 1601-2321
May, 2016

Abstract The purpose of this paper is to illustrate the potential use of discriminant analysis based on tree–structured graphical models for discrete variables. This is done by comparing its empirical performance using estimated error rates for real and simulated data. The results show that discriminant analysis based on tree–structured graphical models is a simple nonlinear method competitive with, and sometimes superior to, other well–known linear methods like those assuming mutual independence between variables and linear logistic regression.

Keywords Discriminant analysis · Discrete variables · Error rates · Tree–structured graphical models · Minimum weight spanning tree · Multinomial distribution · Structure estimation

1 Introduction

The purpose of this paper is to illustrate the use and compare the empirical performance of discrete discriminant analysis with a tree-structured graphical model on each class, specifically with a tree–structured multinomial model.

Discriminant analysis in its parametric form assuming a full multinomial distribution on each class has been considered and presented, for example, in Goldstein and Dillon (1978). Its use, however, has been limited due to the large number of parameters involved for a not so large number of variables, as noted by Krzanowski and Marriott (1995).

A reduced order multinomial distribution on each class, where high order interactions are set to zero, has been proposed and used in order to diminish the number of parameters. For example,

Gonzalo Perez-de-la-Cruz
Department of Applied Mathematics and Computer Science, Statistics and Data Analysis section, Technical University of Denmark, Kgs. Lyngby 2800, Denmark.
E-mail: gonpec@dtu.dk

Guillermina Eslava-Gomez
Department of Mathematics, Faculty of Sciences, UNAM, Circuito Exterior, CU, 04510, D.F. Mexico.
E-mail: eslava@ciencias.unam.mx

Asparoukhov and Krzanowski (2001) considered a second–order log–linear and a second–order Bahadur model. In both cases, multinomial distributions with all pairwise interactions are involved.

A multinomial graphical model on each class is another option for a more parsimonious model. In a graphical model the structure of the distribution, or the set of non zero interactions, is determined by a graph. However, the graph is generally unknown and has to be estimated or identified in addition to the estimation of the parameters of the distribution. The identification of the graph is a problem of great computational complexity for large data sets or high dimensionality. For some subclasses of graphical models, for example some pairwise Markov models, methods for estimating or learning the structure of the graph are based on the lasso regularization, see Loh and Wainwright (2013) and Hastie et al. (2015); these methods involve regularization parameters to be set or determined.

Pairwise Markov models are a class of graphical models which consider only pairwise interactions between discrete variables and whose associated graph does not have cycles of size three. Tree–structured graphical multinomial models are instances of pairwise Markov models with a tree-structured graph. For these models, the identification of the tree structure could be straightforward, mainly due to the existence of efficient algorithms for finding the minimum weight spanning tree and the existence of closed-form expressions for the maximum likelihood estimators of the distribution.

The use of tree–structured multinomial models in classification problems for binary variables was originally suggested by Chow and Liu (1966) where they optimized the estimated error rate to find a tree structure. However, optimizing the estimated error rate over the set of all possible trees of certain order is computational demanding for a large number of variables. Later on, Chow and Liu (1968) identified the tree structure of each population by optimizing the Kullback-Leibler divergence between a tree–structured and the empirical probability function. They proved that this optimization problem was solvable by using minimum weight spanning tree algorithms, and so the exact solution can be found very efficiently even for a large number of variables. Once the tree structure of each distribution was found, they fitted a tree–structured graphical model to each population and used the posterior probabilities of belonging to each of the classes as criterion to classify observations.

Following Chow and Liu (1968), alternative methods for estimating the tree structure of the populations have been studied in the literature where the optimization problems are solvable by using minimum weight spanning tree algorithms. Friedman et al. (1997) considered a similar optimization problem as in Chow and Liu (1968), but restricting the tree to be the same for all classes. Tan et al. (2010) also used tree–structured graphical models for classification. They considered an approximation of the J-divergence to estimate the structure of the tree for each probability function, and used the likelihood ratio of the functions as classification criterion. Perez and Eslava (2016) used a similar procedure for the continuous case, but restricting the tree–structure of the probability functions to be the same for all classes. Additionally, the empirical log–likelihood ratio, either with two arbitrary trees or with the same tree for the distributions, has been considered for the identification of the trees in Tan et al. (2010) and Perez and Eslava (2016).

In section 2 we give relevant background on the multinomial discriminant analysis and on tree–structured graphical models. In section 3 we present some methods for tree structure estimation.

In section 4 we present the results of applying the discriminant methods to a real and simulated data sets. Finally in section 5 we offer some comments.

2 Discrete discriminant analysis

We consider the problem of discrimination between two well defined groups or classes of individuals, Π_1 and Π_2, on the basis of p discrete or categorical variables measured on a sample of individuals from each class.

We first introduce some notation. Let $Y \in \{1, 2\}$ be the class variable and $\mathbf{X} = (X_1, \ldots, X_p)$ the random vector of the p variables. Each variable X_i takes values in the space $\mathcal{X}_i = \{1, \ldots, |\mathcal{X}_i|\}$, where $|\mathcal{X}_i|$ denotes the number of categories or states that X_i takes. Low case letters denote particular elements of $\mathcal{X}_i$, so that $\{X_i = x_i\}$ corresponds to the event that the random variable X_i takes the value $x_i \in \mathcal{X}_i$, $i = 1, \ldots, p$. The vector $\mathbf{X}$ takes values in the discrete state space $\mathcal{X} = \mathcal{X}_1 \times \ldots \times \mathcal{X}_p$, and the number of states that $\mathcal{X}$ takes is $|\mathcal{X}| = \prod_{i \in V} |\mathcal{X}_i|$, where $V = \{1, \ldots, p\}$.

For any subset $A \subseteq V$, $\mathbf{X}_A$ denotes the subvector $(X_i, i \in A)$ which takes values in the space $\mathcal{X}_A = \prod_{i \in A} \mathcal{X}_i$, and $\mathbf{x}_A = (x_i, i \in A)$ refers to a particular element of the space $\mathcal{X}_A$. $P_{\mathbf{X}}(\mathbf{x}) = P(X_1 = x_1, \ldots, X_p = x_p)$ denotes the probability function of $\mathbf{X}$, $P_{\mathbf{X}_A}(\mathbf{x}_A)$ the marginal probability function of the subvector $\mathbf{X}_A$ and $P_{\mathbf{X}_A|\mathbf{X}_B}(\mathbf{x}_A|\mathbf{x}_B)$ the conditional probability function of the subvector $\mathbf{X}_A$ given the subvector $\mathbf{X}_B$, though we shall not use the subindices on the probability distributions, e.g. $P_{\mathbf{X}}(\mathbf{x}) = P(\mathbf{x})$. Finally, let π_1 and π_2 be the *a priori*, or class, probabilities that an observation $\mathbf{x}$ will belong to class Π_1 and Π_2, respectively, $\pi_1 = P(Y = 1)$ and $\pi_2 = P(Y = 2)$.

In discriminant analysis, given an observation $\mathbf{x}$, the optimal classification rule for discrete distributions, Welch (1939), is as follows:

$$\text{Classify } \mathbf{x} \begin{cases} \text{into class } \Pi_1 \quad \text{if } \dfrac{P_1(\mathbf{x})}{P_2(\mathbf{x})} > \dfrac{\pi_2}{\pi_1} \\[2ex] \text{into class } \Pi_2 \quad \text{if } \dfrac{P_1(\mathbf{x})}{P_2(\mathbf{x})} < \dfrac{\pi_2}{\pi_1} \\[2ex] \text{randomly into } \Pi_1 \text{ or } \Pi_2 \quad \text{if } \dfrac{P_1(\mathbf{x})}{P_2(\mathbf{x})} = \dfrac{\pi_2}{\pi_1}, \end{cases} \tag{1}$$

where P_1 and P_2 are the conditional probabilities given by $P_1(\mathbf{x}) = P(X_1 = x_1, \ldots, X_p = x_p|Y = 1)$ and $P_2(\mathbf{x}) = P(X_1 = x_1, \ldots, X_p = x_p|Y = 2)$.

2.1 Tree-structured graphical models

An undirected graphical model corresponds to a family of probability distributions over a random vector $\mathbf{X} = (X_1, \ldots, X_p)$ with an associated undirected graph $G = (V, E)$ which has vertex set $V = \{1, \ldots, p\}$ and edge set $E \subseteq \{(i, j)|\ i < j,\ i, j \in V\}$. The random vector $\mathbf{X}$ is indexed by the nodes of G, so that each variable X_i is represented by a node i, $i = 1, \ldots, p$. The structure of G encodes marginal and conditional independence properties of $\mathbf{X}$. We assume positive probability functions which ensures that they factorize with respect to their graph G, see for example

Lauritzen (1996).

Given a graph $G = (V, E)$, a clique $C \subseteq V$ is a subset whose induced graph is fully connected, that is, the edge $(s, t) \in E$ for all $s, t \in C, s < t$. A separator $S \subseteq V$ is a subset such that its removal splits the graph into two or more subgraphs. Let P_G be a member of a graphical model with associated graph G, and assume $P_G(\mathbf{x}) > 0 \; \forall \; \mathbf{x} \in \mathcal{X}$, then P_G can be factorized in terms of the set of cliques $\mathcal{C}$ of its graph:

$$P_G(\mathbf{x}) = \frac{1}{Z} \prod_{C \in \mathcal{C}} \psi_C(\mathbf{x}_C)$$

where $\psi_C : \mathcal{X}_C \to \mathbb{R}^+$ is a positive function for each $C \in \mathcal{C}$ and $Z = \sum_{\mathbf{x} \in \mathcal{X}} \prod_{C \in \mathcal{C}} \psi_C(\mathbf{x}_C)$ is the normalizing factor.

If, additionally, G is triangulated, the model is called decomposable and can be factorized in terms of marginal probability functions:

$$P_{dec}(\mathbf{x}) = \frac{\prod_{C \in \mathcal{C}} P(\mathbf{x}_C)}{\prod_{S \in \mathcal{S}} P(\mathbf{x}_S)^{v(S)}},$$

where $\mathcal{S}$ and $\mathcal{C}$ are the sets of separators and maximal cliques, respectively, and $v(S)$ is the multiplicity of S in a perfect sequence.

A subclass of decomposable graphical models which are also Markov pairwise models consists of those whose graph G is a tree $\tau = (V, E_\tau)$. They can be factorized in terms of marginal functions of one and two dimensions as

$$P_\tau(\mathbf{x}) = \prod_{i=1}^{p} P(X_i = x_i) \prod_{(i,j) \in E_\tau} \frac{P(X_i = x_i, X_j = x_j)}{P(X_i = x_i) P(X_j = x_j)}.$$

For example, if $\tau = (V, E_\tau)$ is a path $P_\tau(\mathbf{x}) = \prod_{i=1}^{p-1} P(x_i, x_{i+1}) / \prod_{i=2}^{p-1} P(x_i)$ with $E_\tau = \{(1, 2), \ldots, (p-1, p)\}$; or a star with center at node 1, $P_\tau(\mathbf{x}) = \prod_{i=2}^{p} P(x_1, x_i) / P(x_1)^{p-2}$ with $E_\tau = \{(1, i) |\; i = 2, \ldots, p\}$.

The maximum likelihood estimator (MLE) of P_τ is obtained by replacing the corresponding MLE of the marginal distributions. Assuming that P is a multinomial distribution and considering a sample of n independent multivariate observations, $\{\mathbf{x}^l\}_{l=1}^n$, the MLE of the marginal probability functions are the relative frequencies based on observed marginal counts $n(x_i)$ and $n(x_i, x_j)$. That is:

$$\widehat{P}(X_i = x_i) = \frac{n(x_i)}{n} \quad \text{and} \quad \widehat{P}(X_i = x_i, X_j = x_j) = \frac{n(x_i, x_j)}{n},$$

where $n(x_i) = \sum_{l=1}^n \mathbb{I}(x_i^l = x_i)$ and $n(x_i, x_j) = \sum_{l=1}^n \mathbb{I}(x_i^l = x_i) \mathbb{I}(x_j^l = x_j)$, for $x_i \in \mathcal{X}_i$, and where the indicator function $\mathbb{I}(x_i^l = x_i) = 1$ if $x_i^l = x_i$ and 0 otherwise, $l \in \{1, \ldots, n\}$; $i, j \in V$. Then,

$$
\begin{aligned}
\widehat{P}_\tau(\mathbf{x}) &= \prod_{i=1}^{p} \widehat{P}(X_i = x_i) \prod_{(i,j)\in E_\tau} \frac{\widehat{P}(X_i = x_i, X_j = x_j)}{\widehat{P}(X_i = x_i)\widehat{P}(X_j = x_j)} \\
&= \prod_{i=1}^{p} \frac{n(x_i)}{n} \prod_{(i,j)\in E_\tau} \frac{n(x_i, x_j)/n}{(n(x_i)/n)(n(x_j)/n)} \\
&= \frac{1}{n}\prod_{i=1}^{p} n(x_i) \prod_{(i,j)\in E_\tau} \frac{n(x_i, x_j)}{n(x_i)n(x_j)}.
\end{aligned}
\tag{2}
$$

2.2 Tree-structured discriminant analysis

Consider the classification rule given in (1) for a tree–structured multinomial probability function on each class, $P_{1_{\tau_1}}$ and $P_{2_{\tau_2}}$, respectively. Using the MLE given in (2) and estimating the *a priori* probabilities by the proportion of observations on each group, $\hat{\pi}_1 = n_1/n$ and $\hat{\pi}_2 = n_2/n$, with $n_1 + n_2 = n$, the estimated rule becomes:

$$
\text{Classify } \mathbf{x} \begin{cases}
\text{into class } \Pi_1 \quad \text{if } \ln \dfrac{\widehat{P}_{1_{\tau_1}}(\mathbf{x})}{\widehat{P}_{2_{\tau_2}}(\mathbf{x})} > \ln \dfrac{\hat{\pi}_2}{\hat{\pi}_1} \\[2ex]
\text{into class } \Pi_2 \quad \text{if } \ln \dfrac{\widehat{P}_{1_{\tau_1}}(\mathbf{x})}{\widehat{P}_{2_{\tau_2}}(\mathbf{x})} < \ln \dfrac{\hat{\pi}_2}{\hat{\pi}_1} \\[2ex]
\text{randomly into } \Pi_1 \text{ or } \Pi_2 \quad \text{if } \ln \dfrac{\widehat{P}_{1_{\tau_1}}(\mathbf{x})}{\widehat{P}_{2_{\tau_2}}(\mathbf{x})} = \ln \dfrac{\hat{\pi}_2}{\hat{\pi}_1},
\end{cases}
\tag{3}
$$

or equivalently expressed in terms of marginal counts using (2).

Notice that in order to use rule (3), τ_1 and τ_2 have to be identified. In section 3 we formulate the estimation or identification of the tree structure associated with each of the two probability functions.

2.3 The saturated and the independence model

The full multinomial distribution can be seen as a decomposable graphical model with the complete graph $\kappa = (V, E_\kappa)$ as associated graph. This model is also known as the saturated model. In this case the MLE of P_κ for each state is simply the observed relative frequency in the state, that is,

$$
\widehat{P}_\kappa(\mathbf{x}) = \frac{n(\mathbf{x})}{n} = \frac{n(x_1, \ldots, x_p)}{n} \quad \forall\, \mathbf{x} \in \mathcal{X}. \tag{4}
$$

On the other hand, the multinomial distribution under the assumption of independence between any pair of variables, referred as the independence model, can be seen as a graphical model with the edgeless graph $\bar{\kappa} = (V, E_{\bar{\kappa}})$ as associated graph. The MLE of $P_{\bar{\kappa}}$ is the product of the observed relative frequencies for each variable

$$
\widehat{P}_{\bar{\kappa}}(\mathbf{x}) = \prod_{i\in V} \frac{n(x_i)}{n} \quad \forall\, \mathbf{x} \in \mathcal{X}. \tag{5}
$$

3 Tree structure estimation

In this section we describe some methods for identifying the tree structures τ_1 and τ_2 required for the estimated rule (3). Once the tree structures are estimated rule (3) can be applied to the data.

Chow and Liu (1966) proposed the use of the estimated error rate as a function f to be optimized in order to find the tree structures, considering binary variables. They restricted the tree structures to be the same in all classes. For two groups, $\tau_1 = \tau_2 = \tau$, the estimated tree structure τ^* is such that

$$\tau^* = \underset{\tau \in T_p}{\operatorname{argmin}}\ f(\tau) \tag{6}$$

with

$$f(\tau) = \left\{ \sum_{l=1}^{n_1} \mathbb{I}\left(\ln \frac{\widehat{P}_{1_\tau}(\mathbf{x}^l)}{\widehat{P}_{2_\tau}(\mathbf{x}^l)} \leq \ln \frac{n_2}{n_1} \right) + \sum_{l=n_1+1}^{n_1+n_2} \mathbb{I}\left(\ln \frac{\widehat{P}_{2_\tau}(\mathbf{x}^l)}{\widehat{P}_{1_\tau}(\mathbf{x}^l)} \geq \ln \frac{n_2}{n_1} \right) \right\},$$

where T_p is the set of all trees with p nodes, and observations belonging to population one and two have indices ranging from 1 to n_1 and from $n_1 + 1$ to $n_1 + n_2$, respectively.

Since $|T_p| = p^{p-2}$, the evaluation of f for each $\tau \in T_p$ is computationally demanding when the number of variables is large. For this reason, they gave a stepwise routine to find an approximated solution of the problem in (6). Later, in Chow and Liu (1968), the tree structure for each population was found separately by minimizing the Kullback–Leibler divergence between a tree-structured and the empirical distribution. For two populations the estimated tree τ_y^* associated with each population is such that

$$\tau_y^* = \underset{\tau \in T_p}{\operatorname{argmin}}\ f_y(\tau) \tag{7}$$

with $f_y(\tau) = I(\widehat{P}_{y_\kappa}, \widehat{P}_{y_\tau}) = \sum_{\mathbf{x}\in\mathcal{X}} \widehat{P}_{y_\kappa}(\mathbf{x}) \ln(\widehat{P}_{y_\kappa}(\mathbf{x})/\widehat{P}_{y_\tau}(\mathbf{x}))$, $y \in \{1, 2\}$. Notice that the empirical distribution corresponds to the estimated full multinomial distribution in each population $\widehat{P}_{y_\kappa}(\mathbf{x})$ as given in (4).

They proved that the exact solution of (7) can be found efficiently due to the equivalence between this optimization problem and the one of finding a minimum weight spanning tree (MWST). Additionally, they proved that this problem is equivalent to the one of finding the maximum likelihood tree for each population, that is, to finding τ_y^* such that $f_y(\tau_y^*) \geq f_y(\tau)\ \forall\ \tau \in T_p$, with $f_y(\tau) = \sum_{l=1}^{n_y} \ln \widehat{P}_{y_\tau}\left(\mathbf{x}^l\right)$, $y \in \{1, 2\}$.

In the MWST problem there is no need to evaluate $f(\tau)$ for each $\tau \in T_p$ in order to find the exact solution. The solution is found by computing a weight for each possible edge and using efficient algorithms developed for solving the MWST problem, for example Kruskal's (Kruskal 1956) or Prim's algorithm (Prim 1957). In the MWST problem a spanning tree τ^* of a graph G is being found such that

$$\tau^* = \underset{\tau \in T_G}{\operatorname{argmin}} \sum_{(i,j)\in E_\tau} \lambda(i, j)\ , \tag{8}$$

where $\lambda(i, j)$ is a specific weight given to the edge $(i, j) \in E_G$ and $T_G \subseteq T_p$ is the set of all the spanning trees of G. For the problem in (7) and its equivalence with the one in (8), $G = \kappa$ since the searching is over all trees in T_p and the weights are found by noticing that for a given tree $\tau = (V, E_\tau)$

$$I(\widehat{P}_{y_\kappa}, \widehat{P}_{y_\tau}) = -\sum_{(i,j)\in E_\tau} \sum_{x_i\in\mathcal{X}_i, x_j\in\mathcal{X}_j} \widehat{P}_y(x_i, x_j) \ln \frac{\widehat{P}_y(x_i, x_j)}{\widehat{P}_y(x_i)\widehat{P}_y(x_j)} + D,$$

with D a constant that does not depend on τ. The weight for any edge is:

$$\begin{aligned}\lambda_y(i,j) &= -\sum_{x_i\in\mathcal{X}_i, x_j\in\mathcal{X}_j} \widehat{P}_y(x_i, x_j) \ln \frac{\widehat{P}_y(x_i, x_j)}{\widehat{P}_y(x_i)\widehat{P}_y(x_j)} \\ &= -\sum_{x_i\in\mathcal{X}_i, x_j\in\mathcal{X}_j} \frac{n_y(x_i, x_j)}{n_y} \ln \frac{n_y(x_i, x_j)}{n_y(x_i)n_y(x_j)/n_y} \quad \forall\,(i,j) \in E_\kappa.\end{aligned}$$

Other structure estimation methods based on alternative functions $f(\tau_1, \tau_2)$, whose optimization is equivalent to finding MWSTs, have also been considered in the literature. We restrict the comparison to a set of six existing methods with this property. In these methods, the associated optimization problem is defined by maximizing one the following three functions and assuming either two arbitrary trees or of a single one: Maximum Likelihood ($ML_{\tau_1\tau_2}$ and ML_τ); Approximated J-divergence ($AJD_{\tau_1\tau_2}$ and ADJ_τ); and Empirical Log Likelihood Ratio ($ELLR_{\tau_1\tau_2}$ and $ELLR_\tau$).

When $\tau_1 = \tau_2 = \tau$ only one MWST problem is involved. When τ_1 and τ_2 are arbitrary the optimization problem

$$(\tau_1^*, \tau_2^*) = \operatorname*{argmax}_{\tau_1,\tau_2\in T_p} f(\tau_1, \tau_2) = \operatorname*{argmax}_{\tau_1,\tau_2\in T_p} f_1(\tau_1) + f_2(\tau_2) \tag{9}$$

can equivalently be expressed as two independent MWST problems:

$$\tau_1^* = \operatorname*{argmin}_{\tau\in T_p} \sum_{(i,j)\in E_\tau} \lambda_1(i,j) \quad \text{and} \quad \tau_2^* = \operatorname*{argmin}_{\tau\in T_p} \sum_{(i,j)\in E_\tau} \lambda_2(i,j), \tag{10}$$

for specific weights $\lambda_1(i,j)$ and $\lambda_2(i,j)$ which depend on $f_1(\tau_1)$ and $f_2(\tau_2)$, respectively. The functions and associated weights for each of the six methods are described in Table 1.

The $ML_{\tau_1\tau_2}$ method corresponds to problem in (7), introduced in Chow and Liu (1968). ML_τ was proposed by Friedman et al. (1997) when considering a graph structure called TAN-Network which has the property that the subgraph induced by removing the class variable is a tree. $AJD_{\tau_1\tau_2}$ and $ELLR_{\tau_1\tau_2}$ were introduced in Tan et al. (2010); these methods are equivalent when the group sample sizes are the same, since (12) and (13) become proportional. AJD_τ and $ELLR_\tau$ were studied in Perez and Eslava (2016); AJD_τ considers the J-divergence $J(\widehat{P}_{1_\tau}(\mathbf{x}), \widehat{P}_{2_\tau}(\mathbf{x}))$, though an equivalent expression is given in Table 1. AJD_τ and $ELLR_\tau$ are equivalent when the group sample sizes are the same.

The empirical comparison of the rule in (3) with trees estimated by these methods is presented in the following section.

Table 1 Functions to be maximized to estimate the structure of the trees, and corresponding weights associated with the MWST problems.

Method	Function / Weights
$ML_{\tau_1\tau_2}$	$f(\tau_1,\tau_2) = \sum_{l=1}^{n_1} \ln \widehat{P}_{1_{\tau_1}}\left(\mathbf{x}^l\right) + \sum_{l=n_1+1}^{n_2} \ln \widehat{P}_{2_{\tau_2}}\left(\mathbf{x}^l\right)$ $\lambda_y(i,j) = -\sum_{x_i\in\mathcal{X}_i, x_j\in\mathcal{X}_j} n_y(x_i,x_j) \ln \frac{n_y(x_i,x_j)}{n_y(x_i)n_y(x_j)},\ y\in\{1,2\}. \quad (11)$
ML_{τ}	$f(\tau) = \sum_{l=1}^{n_1} \ln \widehat{P}_{1_{\tau}}\left(\mathbf{x}^l\right) + \sum_{l=n_1+1}^{n_2} \ln \widehat{P}_{2_{\tau}}\left(\mathbf{x}^l\right)$ $\lambda(i,j) = \lambda_1(i,j) + \lambda_2(i,j);\ \text{ with } \lambda_y(i,j) \text{ as in (11).}$
$AJD_{\tau_1\tau_2}$	$f(\tau_1,\tau_2) = \sum_{\mathbf{x}\in\mathcal{X}} \left(\widehat{P}_{1_\kappa}(\mathbf{x}) - \widehat{P}_{2_\kappa}(\mathbf{x})\right) \ln \frac{\widehat{P}_{1_{\tau_1}}(\mathbf{x})}{\widehat{P}_{2_{\tau_2}}(\mathbf{x})}$ $\lambda_y(i,j) = -\sum_{x_i\in\mathcal{X}_i, x_j\in\mathcal{X}_j} \left(\frac{n_y(x_i,x_j)}{n_y} - \frac{n_{3-y}(x_i,x_j)}{n_{3-y}}\right) \ln \frac{n_y(x_i,x_j)}{n_y(x_i)n_y(x_j)},\ y\in\{1,2\}. \quad (12)$
AJD_{τ}	$f(\tau) = \sum_{\mathbf{x}\in\mathcal{X}} \left(\widehat{P}_{1_\kappa}(\mathbf{x}) - \widehat{P}_{2_\kappa}(\mathbf{x})\right) \ln \frac{\widehat{P}_{1_{\tau}}(\mathbf{x})}{\widehat{P}_{2_{\tau}}(\mathbf{x})}$ $\lambda(i,j) = \lambda_1(i,j) + \lambda_2(i,j);\ \text{ with } \lambda_y(i,j) \text{ as in (12).}$
$ELLR_{\tau_1\tau_2}$	$f(\tau_1,\tau_2) = \sum_{l=1}^{n_1} \ln \frac{\widehat{P}_{1_{\tau_1}}(\mathbf{x}^l)}{\widehat{P}_{2_{\tau_2}}(\mathbf{x}^l)} + \sum_{l=n_1+1}^{n_1+n_2} \ln \frac{\widehat{P}_{2_{\tau_2}}(\mathbf{x}^l)}{\widehat{P}_{1_{\tau_1}}(\mathbf{x}^l)}$ $\lambda_y(i,j) = -\sum_{x_i\in\mathcal{X}_i, x_j\in\mathcal{X}_j} \left(n_y(x_i,x_j) - n_{3-y}(x_i,x_j)\right) \ln \frac{n_y(x_i,x_j)}{n_y(x_i)n_y(x_j)},\ y\in\{1,2\}. \quad (13)$
$ELLR_{\tau_1}$	$f(\tau) = \sum_{l=1}^{n_1} \ln \frac{\widehat{P}_{1_{\tau}}(\mathbf{x}^l)}{\widehat{P}_{2_{\tau}}(\mathbf{x}^l)} + \sum_{l=n_1+1}^{n_1+n_2} \ln \frac{\widehat{P}_{2_{\tau}}(\mathbf{x}^l)}{\widehat{P}_{1_{\tau}}(\mathbf{x}^l)}$ $\lambda(i,j) = \lambda_1(i,j) + \lambda_2(i,j);\ \text{ with } \lambda_y(i,j) \text{ as in (13).}$

4 Empirical performance

In this section we illustrate the performance of discriminant analysis with trees estimated by different methods, in an intensive care unit problem and in a simulation study. In addition to the rule given in (3) with trees selected by the six methods presented in Table 1, we consider four alternative possibilities: rule (3) with an edgeless graph (independent model), with a decomposable graph (decomposable model), and with a complete graph (saturated model) on each class; and linear logistic regression; referred respectively as Independent, $ML_{dec_1dec_2}$, Saturated, and Logistic–Reg.

The naive classifier, which corresponds to rule (3) with edgeless graphs, and linear logistic regression are linear classifiers commonly used due to their simplicity and often good classification performance in practical applications. In particular, the independent model is useful in high-dimensional settings or in the presence of sparse data, in these settings discriminant analysis with tree-structured graphical models is a simple nonlinear alternative to use.

The numerical results were obtained using R (R Core Team 2013), mainly packages *gRapHD* (Abreu et al. 2010) and *gRain* (Højsgaard 2012). For estimating the tree structures, the weights associated with the MWST problem for each method presented in Table 1 were implemented as a user defined function which is called by function *minForest* in *gRapHD*. In this sense, the implementation of any of the six methods for finding the structure of the trees is straightforward. The saturated and decomposable models have been fitted using function *extractPOT* in *gRain* with the option smooth set to .1 which corresponds to add .1 in each state $\mathbf{x} \in \mathcal{X}$. A decomposable structure was estimated, separately for each population, using function *stepw* in *gRapHD* with maximum likelihood as the function to be optimized and with the corresponding tree structure found with the $ML_{\tau_1\tau_2}$ method as initial solution.

4.1 Intensive care unit data

This example relates to a problem of discrimination between surviving and non surviving patients admitted into the intensive care unit (ICU) in two hospitals in Mexico city. The data consists of measurements made on 859 patients of which 480 died between the second day of admission into the ICU and three months after hospital discharge while 379 survived. Ten variables are considered in the analysis, seven of which are binary and three with three categories. Some of these variables were obtained by categorizing variables originally continuous.

The variables recorded at the time of admission into or during stay in the ICU are the following. Quality of life prior to admission (X_1 :1 good, 0 bad); use of mechanical ventilation (X_2 : 1 yes, 0 no); score of the predicted death rate based on APACHE II (X_3 : 1 score $>.19$, 0 score $\leq .18$); age (X_4 : 2 [76, 93]; 1 [41, 75]; 0 [16, 40]); number of surgeries while in ICU (X_5 : 2 two or more, 1 one, 0 none); sepsis (X_6 : 2 acquired within hospital, 1 acquired prior entering hospital, 0 none); Glasgow score (X_7 : 1 seven or less, 0 eight or more); Cardiac failure (X_8 : 1 yes, 0 no); Brussels score (X_9 : 1 six or more, 0 five or less); years of school attendance (X_{10} : 1 five or less, 0 six or more), and the indicator variable for nonsurvival (Y : 1 yes, 0 no).

In this example, with seven binary and three ternary variables, the state space on each class is $\mathcal{X} = \{0, 1\}^7 \times \{0, 1, 2\}^3$ and $|\mathcal{X}| = 3456$. Therefore, to ensure that all states have at least one observation, there should be at least 3456 observations from each population. Considering the saturated model, a very large number of observations relative to the number of variables is required for a precise estimation of all the state probabilities. Another difficulty is when a state $\mathbf{x} \in \mathcal{X}$ has zero observations in both populations, then the allocation is forced to be random. These aspects reflect the problem of sparseness, particularly for the application of the full multinomial model, and the need of more parsimonious models. For some examples and discussion on the problem of sparseness see Goldstein and Dillon (1978). For instance, the number of non empty states for this data set is very low, only 200 (5.8%) for the survival and 261 (7.6%) for the nonsurvival group, and only 93 (2.7%) states have non zero observations in both.

Error rates were estimated by the repeated holdout method. The data set was randomly split into two sets in proportions 4/5 and 1/5 in each population, one for training and one for testing. The discriminant methods were applied and the proportion of misclassified observations for each group were calculated. The procedure was repeated 100 times, and the mean and standard deviations of the observed proportions give the estimated error rate and an estimate of its standard error. Numerical results are displayed in Figure 1, and in Table 2 in the appendix.

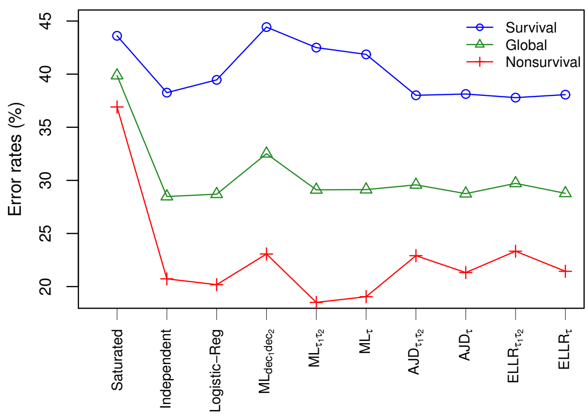

Fig. 1 Error rates estimated by repeated holdout method with 100 random training samples of size $4n/5$

The results show that for all methods but the saturated, global error rates are the same within uncertainty. All methods perform better for the large group (480) than for the small (379); the saturated model has the worst and a bad performance, respectively.

4.2 Simulated data

For a set of correlated variables an independent model might not perform well. In particular for binary variables, Hand (1981) illustrates the case for two binary variables and two groups. We have designed a simulation experiment considering a set of ten correlated binary variables. Each variable has the same marginal distribution within each population, and similar between the two, $P(X_i = 1|\Pi_1) = .5$ and $P(X_i = 1|\Pi_2) = .4$, $i \in \{1, \dots, 10\}$. This is an instance where the independent model is expected not to perform well.

Two multinomial models with different dependence structure among variables were considered to generate the data. In the first one, binary variables were obtained by dichotomizing observations from Gaussian graphical distributions with a random dependence graph. In the second one, the binary variables follow a graphical model associated with a path-structured graph.

4.2.1 Random structure

Samples $\{\mathbf{x}^l\}_{l=1}^{n_y}$, $y \in \{1, 2\}$, were obtained by dichotomizing random samples from a Gaussian graphical model whose associated graph is random.

One set $\{\mathbf{z}^l\}_{l=1}^{n_1}$ from population $\Pi_1 = N_{10}(0, \Sigma_{Rand_1})$ and one $\{\mathbf{z}^l\}_{l=1}^{n_2}$ from $\Pi_2 = N_{10}(0, \Sigma_{Rand_2})$ were generated. Each univariate observation $z_j^l, l = 1, \ldots, n_1$, form Π_1 was dichotomized as $x_j^l = \mathbb{I}(z_j^l < \Phi(.5))$; and each $z_j^l, l = 1, \ldots, n_2$ from Π_2 as $x_j^l = \mathbb{I}(z_j^l < \Phi(.4))$; $j \in \{1, \ldots, 10\}$; Φ denotes the inverse of a standard normal distribution.

The concentration matrices, $\mathbf{\Sigma}_{Rand_1}^{-1}$ and $\mathbf{\Sigma}_{Rand_2}^{-1}$, were taken to have on average 30% of the $p(p-1)/2 = 45$ off–diagonal elements with a non zero value and the rest with a zero value, resulting in 14 and 15 non zero elements on each matrix, respectively.

For each group, a symmetric 10×10 matrix $\mathbf{A} = \{a_{ij}\}$ was generated using random numbers u_{ij} from a uniform distribution $U(-1,1)$, with $a_{ij} = u_{ij}\mathbb{I}(|u_{ij}| > .7)$, $i \neq j$, and $a_{ii} = 1.01 \times \sum_{j \neq i} |a_{ij}|$, $i, j = 1, ..., 10$. The covariance matrix $\mathbf{\Sigma} = \{\sigma_{ij}\}$ is then determined by $\sigma_{ij} = a^{ij}/\sqrt{a^{ii}a^{jj}}$, where a^{ij} is the entry ij of the matrix $\mathbf{A}^{-1}$.

4.2.2 Path structure

Random samples were generated from path–structured multinomial distributions in each population: $P_{y_\tau}(\mathbf{x}) = \prod_{i=1}^{9} P_y(x_i, x_{i+1}) / \prod_{i=2}^{9} P_y(x_i)$, $y = 1, 2$; where $P_1(x_i, x_{i+1}) = .1\mathbb{I}(x_i = x_{i+1}) + .4\mathbb{I}(x_i \neq x_{i+1})$, $P_2(x_i, x_{i+1}) = .1\mathbb{I}(x_i \neq x_{i+1}) + .3\mathbb{I}(x_i = x_{i+1} = 0) + .5\mathbb{I}(x_i = x_{i+1} = 1)$, $i = 1, ..., 9$, and $x_i \in \mathcal{X}_i = \{0, 1\}$.

4.2.3 Results

Data sets were generated considering two cases: equal and different group sample sizes. In each case independent samples were generated, one set for training and one for testing the models: i) training: $n_1 = n_2 = 100$; test: $n_1 = n_2 = 1000$, and ii) training: $n_1 = 2n_2 = 200$; test: $n_1 = 1333, n_2 = 667$. The procedure was repeated 400 times in order to estimate error rates and corresponding standard deviations. Figures 2 and 3, and Tables 3 and 4 in the appendix, display the numerical results for each case.

For equal group sample sizes. For discriminant analysis using trees or decomposable graphs, the results are the same within uncertainty, their performance is better than the rest. Logistic regression and the independent model have an equally bad performance for both the path and the random dependence structures. The saturated model has in general a bad performance particularly for the populations with a random structure.

For different group sample sizes. The performance of the global error rate is as the corresponding in the case for equal group sample sizes. The error rates for the small group are much larger than for those of the large group for the populations with a random structure. The saturated method has the worst performance for the populations with a random structure and a bad one for the path—structured populations; the independent and logistic regression models have a bad performance in both populations.

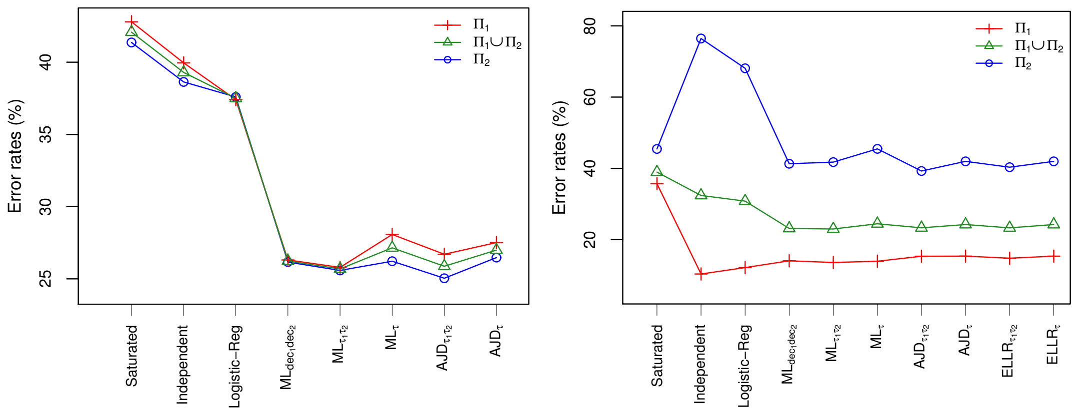

Fig. 2 Random structure. Average of 400 error rates estimated from independent training and test samples. Left: training $n_1 = n_2 = 100$ and test $n_1 = n_2 = 1000$; $AJD_{\tau_1\tau_2} \equiv ELLR_{\tau_1\tau_2}$; $AJD_\tau \equiv ELLR_\tau$. Right: training $n_1 = 2n_2 = 200$ and test $n_1 = 1333$ and $n_2 = 667$

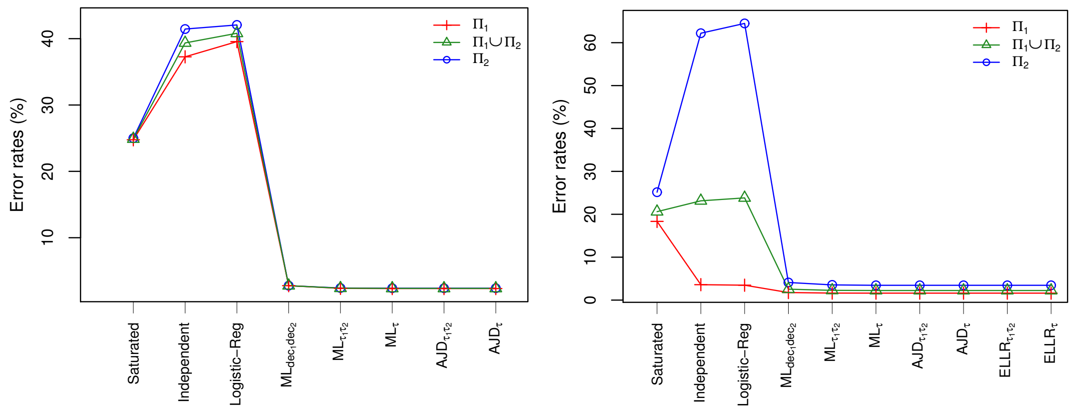

Fig. 3 Path structure. Average of 400 error rates estimated from independent training and test samples. Left: training $n_1 = n_2 = 100$ and test $n_1 = n_2 = 1000$; $AJD_{\tau_1\tau_2} \equiv ELLR_{\tau_1\tau_2}$; $AJD_\tau \equiv ELLR_\tau$. Right: training $n_1 = 2n_2 = 200$ and test $n_1 = 1333$ and $n_2 = 667$

Before giving some comments we give some remarks.

Remark 1 The performance of the classification rule given in (3), for any of the data sets, has not favoured any of the methods given in Table 1 for selecting the structure of the trees. Any of the six methods could be used. However, it should be noted that $ML_{\tau_1\tau_2}$ and ML_τ methods aim at approximating each probability function P_1 and P_2 with $\widehat{P}_{1_{\tau_1}}$ and $\widehat{P}_{2_{\tau_2}}$, either with two arbitrary trees or with a single one, respectively. These estimated functions can be helpful to explain the relationship among the variables in each class. Whereas the other methods aim at finding $\widehat{P}_{1_{\tau_1}}$ and $\widehat{P}_{2_{\tau_2}}$ that maximize the divergence between the two.

Remark 2 The study has been done for two classes, its extension to more than two is straightforward. For $K > 2$ classes, the methods for estimating the structure of the trees have to be implemented considering either K arbitrary trees or a single one. The function to be optimized for each method in Table 1 can be easily modified to include K populations, and its optimization would still be equivalent to K or to one MWST problem. Once the trees are identified, the classification rule given in (3) is equivalent to simply classify $\mathbf{x}$ to the class with the highest estimated posterior probability $\hat{\pi}_k \widehat{P}_{k_{\tau_k}}(\mathbf{x}) / \sum_{i=1}^{K} \hat{\pi}_i \widehat{P}_{i_{\tau_i}}(\mathbf{x}),\ k \in \{1, \ldots, K\}$.

Remark 3 In the context of classification, discriminant analysis with a tree-structured graphical model for mixed variables, continuos and discrete, in each population is a natural extension to the work presented here. For a single population some research has already been done. For instance, Lee and Hastie (2015) study the estimation of the graph–structure for pairwise graphical models, and Cheng et al. (2016) study more complicated mixed graphical models, both using the lasso. Whereas Edwards et al. (2010) and Højsgaard et al. (2012) have considered the class of tree–structured graphical models, using maximum likelihood to estimate the structure of the tree with algorithms for finding the exact tree.

5 Discussion

We have illustrated the use and performance of discrete discriminant analysis with tree–structured graphical distributions on each of two classes. The results show that this method offers a good tool for practitioners for classifying multivariate discrete observations. The method takes into account a subset of pairwise interactions between variables making it a parsimonious model useful to deal with some of the effects of sparseness in discrete data sets.

Discriminant analysis using a full multinomial model on each class, although being very simple to compute, in practice demands large sample sizes to perform well and to avoid classify observations randomly into one of the classes. The simple method which assumes an independent model on each class is also very simple to compute and its use, like the one assuming full multinomial models, does not require the estimation of a graph structure. Although the assumption of mutual independence is often not tenable in practice, this method performed well for the intensive care unit data; however when the main difference between the two groups is due to the interaction of the variables, as in the simulated data, this method is not effective.

Discrimination based on tree–structured multinomial models falls between methods using a full and an independent multinomial distribution with respect to the kind of correlation structure. It considers only $p-1$ of all $p(p-1)/2$ pairwise interactions between p variables in each population. Its performance for the intensive care unit data, regardless of the method used to estimate the structure of the trees, was as good as the best method which assumes mutual independence. For the simulated data, its performance was superior. There are, of course, structured populations where interactions among variables cannot be captured, not even approximated, by a tree–structured distribution. However, the use of these models with some pairwise interactions might help to improve the classification performance of discriminant analysis without increasing the computational complexity when compared with linear classifiers.

The method using a decomposable multinomial model on each class performed equally well within uncertainty, with estimated error rates marginally higher, than the method that best performed. These models consider higher order interactions, but the estimation of the structure of the decomposable graph cannot be solved in an exact way as for tree–structured models.

Logistic discrimination performed equally well as the other methods and better than the one using a saturated model for the intensive care unit data. For the simulated data, its performance was worst than discrimination based on tree–structured models, this was expected since logistic discrimination was taken as a linear logistic regression with no interaction terms.

The main conclusion is that discriminant analysis with a tree–structured multinomial distribution, also referred as tree–structured graphical model, in each class is a good competitor to simple methods which are robust against the effects of sparseness, like the one assuming mutual independent variables, with the advantage of taking into account some of the dependence structure of the variables through pairwise interactions.

Acknowledgements This work was written while GEG was at the Department of Applied Mathematics and Computer Science, Technical University of Denmark, on Sabbatical leave from the Faculty of Sciences at the National Autonomous University of Mexico (UNAM). We are very grateful to Drs. H. Avila Rosas and L. D. Sánchez Velázquez for providing the ICU data, and for helpful discussions concerning the codification and guided choice of variables. GPC received a postdoctoral grant (252737) by the National Council of Science and Technology (CONACYT) of Mexico.
Guillermina Eslava-Gómez acknowledges the receipt of a grant from the program PASPA from DGAPA, UNAM, for six months Sabbatical leave.

References

Asparoukhov OK, Krzanowski WJ (2001) A comparison of discriminant procedures for binary variables. Compt Stat Data An 38: 139-160
Cheng J, Li T, Levina E, Zhu J (2016) High-Dimensional Mixed Graphical Models. arXiv:1304.2810v2 [stat.ML]
Chow C, Liu C (1966) An approach to structure adaptation in pattern recognition. IEEE T Syst Sci Cyb 2: 73-80
Chow C, Liu C (1968) Approximating discrete probability distributions with dependence trees. IEEE T Inform Theory 14: 462-467
Edwards D, Abreu G, Labouriau R (2010) Selecting High-Dimensional Mixed Graphical Models Using Minimal AIC or BIC Forests. BMC Bioinformatics: 11-18
Friedman N, Geiger D, Goldszmidt M (1997) Bayesian network classifiers. Mach Learn 29: 131-163
Goldstein M, Dillon WR (1978) Discrete discriminant analysis. Wiley
Hand DJ (1981) Discrimination and classification. Wiley
Hastie T, Tibshirani R, Wainwright M (2015) Statistical learning with sparsity. The lasso and generalizations. CRC Press
Højsgaard S (2012) Graphical Independence Networks with the gRain Package for R. J Stat Softw 46(10): 1-26
Højsgaard S, Lauritzen SL, Edwards D (2012) Graphical Models with R. Springer
Krzanowski WJ, Marriott FHC (1995) Multivariate analysis Part 2: Classification, Covariance, Structures and Repeated Measurements. Arnold, London
Kruskal JB (1956) On the Shortest Spanning Subtree of a Graph and the Traveling Salesman Problem. P Am Math Soc 7: 48-50
Lauritzen SL (1996) Graphical Models. Oxford University Press
Lee JD, Hastie TJ (2015) Learning the Structure of Mixed Graphical Models. J Comput Graph Stat 24(1): 230-253
Loh PL, Wainwright MJ (2013) Structure estimation for discrete graphical models: generalized covariance matrices and their inverses. Ann Stat 41: 3022-3049
Perez-de-la-Cruz G, Eslava-Gomez G (2016) Discriminant analysis with Gaussian graphical tree models. AStA Adv Stat Anal 100: 161-187
Prim RC (1957) Shortest connection networks and some generalizations. Bell Syst Tech J 36: 1389-1401
R Core Team (2013) R:A Language and Environment for Statistical Computing. R Foundation for Statistical Computing, Vienna, Austria. http://www.R-project.org/
Tan VYF, Sanghavi S, Fisher JW, Willsky AS (2010) Learning Graphical Models for Hypothesis Testing and Classification. IEEE T Signal Proces 58: 5481-5495
Welch BL (1939) Note on discriminant functions. Biometrika 31: 218–220

A Appendix

Table 2 Intensive care unit data. Error rates estimated by repeated holdout method with 100 random training samples of size $4n/5$

Method	Nonsurvival $n_1 = 480$	Survival $n_2 = 379$	Global $n = 859$ (se)
Saturated	36.9	43.6	39.9 (3.4)
Independent	20.7	38.3	**28.5** (3.3)
Logistic-Reg	20.2	39.5	28.7 (3.1)
$ML_{dec_1 dec_2}$	23.1	44.4	32.5 (3.9)
$ML_{\tau_1\tau_2}$	**18.5**	42.5	29.1 (3.0)
ML_{τ}	19.1	41.9	29.1 (3.0)
$AJD_{\tau_1\tau_2}$	22.9	38.0	29.6 (3.1)
AJD_{τ}	21.3	38.1	28.8 (3.2)
$ELLR_{\tau_1\tau_2}$	23.3	**37.8**	29.7 (2.9)
$ELLR_{\tau}$	21.4	38.1	28.8 (3.1)

Table 3 Random structure. Average of 400 error rates estimated from independent training and test samples. Left panel: training $n_1 = n_2 = 100$ and test $n_1 = n_2 = 1000$; $AJD_{\tau_1\tau_2} \equiv ELLR_{\tau_1\tau_2}$; $AJD_{\tau} \equiv ELLR_{\tau}$. Right panel: training $n_1 = 2n_2 = 200$ and test $n_1 = 1333$ and $n_2 = 667$

	$n_1 = n_2 = 100$			$n_1 = 200,\ n_2 = 100$		
Method	Π_1	Π_2	$\Pi_1 \cup \Pi_2$(se)	Π_1	Π_2	$\Pi_1 \cup \Pi_2$ (se)
Saturated	42.80	41.37	42.08 (1.4)	35.70	45.45	38.95 (1.4)
Independent	39.96	38.63	39.29 (2.0)	**10.36**	76.44	32.40 (1.6)
Logistic-Reg	37.41	37.60	37.50 (1.7)	12.16	68.09	30.81 (1.4)
$ML_{dec_1 dec_2}$	26.31	26.16	26.23 (1.7)	14.07	41.30	23.15 (1.4)
$ML_{\tau_1\tau_2}$	**25.79**	25.58	**25.69** (1.6)	13.60	41.76	**22.99** (1.3)
ML_{τ}	28.07	26.22	27.15 (1.9)	13.94	45.47	24.46 (1.4)
$AJD_{\tau_1\tau_2}$	26.70	**25.04**	25.87 (1.4)	15.35	**39.26**	23.32 (1.6)
AJD_{τ}	27.51	26.47	26.99 (1.6)	15.38	41.97	24.25 (1.7)
$ELLR_{\tau_1\tau_2}$				14.79	40.33	23.31 (1.5)
$ELLR_{\tau}$				15.37	41.97	24.24 (1.7)

Table 4 Path structure. Average of 400 error rates estimated from independent training and test samples. Left panel: training $n_1 = n_2 = 100$ and test $n_1 = n_2 = 1000$; $AJD_{\tau_1\tau_2} \equiv ELLR_{\tau_1\tau_2}$; $AJD_{\tau} \equiv ELLR_{\tau}$. Right panel: training $n_1 = 2n_2 = 200$ and test $n_1 = 1333$ and $n_2 = 667$

	$n_1 = n_2 = 100$			$n_1 = 200,\ n_2 = 100$		
Method	Π_1	Π_2	$\Pi_1 \cup \Pi_2$ (se)	Π_1	Π_2	$\Pi_1 \cup \Pi_2$ (se)
Saturated	24.76	25.00	24.88 (1.2)	18.34	25.15	20.61 (1.2)
Independent	37.26	41.45	39.36 (3.7)	3.59	62.21	23.14 (2.2)
Logistic-Reg	39.55	42.07	40.81 (3.7)	3.47	64.49	23.82 (1.9)
$ML_{dec_1 dec_2}$	2.77	2.77	2.77 (0.5)	1.76	4.10	2.54 (0.5)
$ML_{\tau_1\tau_2}$	2.37	2.42	2.39 (0.3)	1.63	3.54	2.27 (0.4)
ML_{τ}	**2.34**	**2.39**	**2.36** (0.3)	**1.62**	**3.45**	**2.23** (0.3)
$AJD_{\tau_1\tau_2}$	**2.34**	**2.39**	**2.36** (0.3)	**1.62**	**3.45**	**2.23** (0.3)
AJD_{τ}	**2.34**	**2.39**	**2.36** (0.3)	**1.62**	**3.45**	**2.23** (0.3)
$ELLR_{\tau_1\tau_2}$				**1.62**	**3.45**	**2.23** (0.3)
$ELLR_{\tau}$				**1.62**	**3.45**	**2.23** (0.3)