Turkish Journal of Electrical Engineering & Computer Sciences

http://journals.tubitak.gov.tr/elektrik/

Research Article

Turk J Elec Eng & Comp Sci
(2019) 27: 4378 – 4389

doi:10.3906/elk-1811-4

Heart attack mortality prediction: an application of machine learning methods

Issam SALMAN*
Department of Software Engineering, Faculty of Nuclear Sciences and Physical Engineering,
Czech Technical University, The Czech Republic

Received: 07.11.2018 • **Accepted/Published Online:** 01.07.2019 • **Final Version:** 26.11.2019

Abstract: The heart is an important organ in the human body, and acute myocardial infarction (AMI) is the leading cause of death in most countries. Researchers are doing a lot of data analysis work to assist doctors in predicting the heart problem. An analysis of the data related to different health problems and its functions can help in predicting the wellness of this organ with a degree of certainty. Our research reported in this paper consists of two main parts. In the first part of the paper, we compare different predictive models of hospital mortality for patients with AMI. All results presented in this part are based on real data of about 603 patients from a hospital in the Czech Republic and about 184 patients from two hospitals in Syria. Although the learned models may be specific to the data, we also draw more general conclusions that we think are generally valid. In the second part of the paper, because the data is incomplete and imbalanced we develop the Chow–Liu and tree-augmented naive Bayesian to deal with that data in better conditions, and compare the quality of these algorithms with others.

Key words: Machine learning, data mining, classification, incomplete data, imbalanced data, Bayesian networks, acute myocardial infarction

1. Introduction

An enormous amount of data is being generated every day. Analyzing big datasets is impossible without the help of automated procedures. Machine learning [1] provides these procedures. The most commonly used form of machine learning is supervised classification [2]. Its goal is to learn a mapping from the descriptive features of an object to the set of possible classes, given a set of features-class pairs.

Probabilities play a central role in modern machine learning [3]. Probabilistic graphical models (PGMs) [4] have emerged as a general framework for describing and applying probabilistic models. A PGM allows us to efficiently encode a joint distribution over some random variables by making assumptions of conditional independence.

A Bayesian network classifier (BNC) [5] is a Bayesian network applied to the classification task. BNCs have many strengths, including good interpretability, the possibility of including prior knowledge about a domain, and competitive predictive performance. They have been successfully applied in practice, e.g., [6–8].

Acute myocardial infarction (AMI) is commonly known as heart attack. A heart attack occurs when an artery leading to the heart becomes completely blocked and the heart does not get enough blood or oxygen. Without oxygen, cells in that area of the heart die. AMI is responsible for more than half of deaths in most countries worldwide. Its treatment has a significant socioeconomic impact.

*Correspondence: issam.salman@fjfi.cvut.cz

One of the main objectives of our research is to design, analyze, and verify a predictive model of hospital mortality based on clinical data about patients. A model that predicts mortality well can be used, for example, for the evaluation of medical care in different hospitals. Evaluation based merely on mortality would not be fair for hospitals where complicated cases are often dealt with. It seems better to measure the quality of health care using the difference between predicted and observed mortality.

A related work was published by Krumholz et al. in [9], where the authors analyzed the mortality data in USA hospitals using the logistic regression model. In another work [10], the authors designed and verified a predictive model of hospital mortality in ST elevation myocardial infarction (STEMI). In another work [11], the authors analyzed the medical records of patients suffering myocardial infarction from a third world country, Syria, and a developed country, the Czech Republic, and presented an idea of how to deal with incomplete and imbalanced data for tree-augmented naive Bayesian (TAN).

2. Data

Our dataset contains data from 787 patients from 2 different countries (603 patients from The Czech Republic and 184 from Syria) characterized by 24 variables. The attributes are listed in Table 1. Most records contain missing values, i.e. for most patients only some attribute values are available, and some attributes are not available for Syrian patients, i.e. the data is incomplete. The thirty-day mortality is recorded for all patients; 89% of the patients survived, i.e. the data is imbalanced.

In The Czech Republic, the results of blood tests are reported in millimoles per liter of blood. In Syria some of the measurements are reported in milligrams per liter and some in millimoles per liter. We standardized all measurements to the millimoles per liter scale.

3. Machine learning methods

Since the explanatory variables may combine their influence and the influence of a variable may be mediated by another variable, it is worth studying the relations of variables altogether. We will do it in two steps: (1) since the mortality prediction is of our primary interest, we will compare how different classifiers are able to predict mortality, (2) to get an overall picture of the relations between all variables, we will learn some Bayesian network models from the collected data, (3) to handle incomplete and imbalanced data, we will provide an idea of how to develop the Chow–Liu [12] and TAN algorithms [5] to be able to process this data.

We will work with different versions of data which vary depending on how we treat variables that have more than two states: (1) real valued ordinal variables, (2) discrete valued variables (with five states at most), and (3) binary variables. We will discuss the values' transformation in more detail in the next sections.

3.1. Ordinal attributes

In our data, we have several categorical variables (sometimes also called nominal variables). These are variables that have two or more categories. For example, sex is a categorical variable having two categories (male and female). However, for some machine learning methods we need ordinal attributes which are attributes whose values have an ordering of values that is natural for the quantification of their impact on the class. This is satisfied by all attributes that can take only two values even if they are nominal, e.g. by sex (0 for male, 1 for female), mortality (0 for survived, 1 for died). In our data it seems that the ordinality can be assumed for most real valued attributes, but note that the fact that there might also exist laboratory tests whose values deviate

Table 1. Attributes

Attribute	Code	Type	Value range in data	Country
Age	AGE	Real	[23,94]	SYR, CZ
Height	HT	Real	[145,205]	CZ
Weight	WT	Real	[35,150]	CZ
Body mass index	BMI	Real	[16.65,48.98]	CZ
Sex	SEX	Nominal	{male, female}	SYR, CZ
Nationality	NAT	Nominal	{Czech, Syrian}	SYR, CZ
STEMI location	STEMI	Nominal	{inferior, anterior, lateral}	SYR, CZ
Hospital	Hospital	Nominal	{CZ, SYR1, SYR2}	SYR, CZ
Kalium	K	Real	[2.25,7.07]	CZ
Urea	UR	Real	[1.6,61]	SYR, CZ
Kreatinin	KREA	Real	[17,525]	SYR, CZ
Uric acid	KM	Real	[97,935]	SYR, CZ
Albumin	ALB	Real	[16,60]	SYR, CZ
HDL cholesterol	HDLC	Real	[0.38,2.92]	SYR, CZ
Cholesterol	CH	Real	[1.8,9.9]	SYR, CZ
Triacylglycerol	TAG	Real	[0.31,11.9]	SYR, CZ
LDL cholesterol	LDLC	Real	[0.261,7.79]	SYR, CZ
Glucose	GLU	Real	[2.77,25.7]	SYR, CZ
C-reactive protein	CRP	Real	[0.3,359]	SYR, CZ
Cystatin C	CYSC	Real	[0.2,5.22]	SYR, CZ
N-terminal prohormone of brain natriuretic peptide	NTBNP	Real	[22.2,35000]	CZ
Troponin	TRPT	Real	[0,25]	CZ
Glomerular filtration rate (based on MDRD)	GFMD	Real	[0.13,7.31]	CZ
Glomerular filtration rate (based on Cystatin C)	GFCD	Real	[0.09,7.17]	CZ

from a normal range in both directions (i.e. both lower and higher values) may increase the mortality. We will refer to the ordinal data as D.ORD.

3.2. Discrete attributes

Discrete variable is a variable that can take values from a finite set. Some classification methods require discrete variables. To get a statistically reliable estimates of model parameters it is advisable to keep the number of values as low as possible while still being able to express the significant relations. We performed discretization of all real-valued attributes. It is not easy to find the optimum number and the values of split points in discretization. Fortunately, there exists the Czech National Code Book that classifies numeric laboratory results, with respect to age and sex, into nine groups $1, 2, \ldots, 9$. The group 5 corresponds to standard values in the standard population. We further reduced the number of states to 5 by joining some groups together. We will refer to data in this form as D.DISCR.

3.3. Binary attributes

Binary data are data whose variables can take on only two possible states, traditionally termed 0 and 1 in accordance with the binary numeral system and Boolean algebra. In our case, all laboratory tests are encoded using two binary attributes. The first attribute takes a value of 0 for the standard values of the test and a value of 1 if the values are decreased. The second attribute takes a value of 0 for the standard values of the test and value of 1 if the values are increased. The age, height, and weight attributes are removed. From the demographic group of attributes only sex and body mass index (BMI) were kept with BMI being encoded using two binary attributes BMI high and BMI low where the BMI greater than the mean takes a value of 1, otherwise it takes a value of 0. We will refer to data in this form as D.BIN.

3.4. Attribute selection

Before learning a model, we preprocess the data. Usually, one of the most useful parts of preprocessing is the attribute selection, where irrelevant attributes are removed. Attribute selection is a process by which we automatically search for the best subset of attributes in our dataset. The notion of "best" is relative to the problem we are trying to solve, but typically means the highest accuracy. Three key benefits of performing attribute selection on our data are:

- It reduces overfitting. Less redundant data means lower possibility of making decisions based on a noise.
- It improves accuracy. Less misleading data means that modeling accuracy improves.
- It reduces training time. Less data means that algorithms train faster.

The CfsSubsetEval method of Weka [13] selects the subsets of attributes that are highly correlated with the class while having low intercorrelation. We searched the space of all subsets by a greedy best first search with backtracking. Data D after the application of this attribute selection method will be suffixed as D.AS.

3.5. Tested classifiers

For tests, we used a large subset of classifiers implemented in Weka. Classifiers that performed best in the preliminary tests qualified for the final tests. In the final tests we compared the following classifiers:

- Decision tree C4.5 [14].
- Logistic regression [15].
- Naive Bayes (NB) classifier [16] assumes that the value of a particular explanatory variable (attribute) is independent of the value of any other attribute given the class variable.
- NB-tree generates a decision tree with naive Bayes classifiers at the leaves [17]
- Bayesian network (BN) classifiers (1) learned by K2 algorithm [18]—referred to as BN.K2 and (2) Tree augmented naive bayes classifier referred to as BN.TAN [5].

All BN algorithms implemented in Weka assume that all variables are discrete finite variables. We will use NA in the results of these classification methods.

We use the leave-one-out cross-validation as the model evaluation method. It means that N separate times, the classifier is trained on all the data except for one point and a prediction is made for that point. After that, the average error is computed and used to evaluate the model.

3.6. Prediction quality

For each data record classified by a classifier there are possible classification results. Either the classifier got a positive example labeled as positive (in our data the positive example is the patient not survived) or it made a mistake and marked it as negative. Conversely, a negative example may have been mislabeled as a positive one, or correctly marked as negative. This defines the following metrics:

- True positives (TP): number of positive examples, labeled as such.
- False positives (FP): number of negative examples, labeled as positive.
- True negatives (TN): number of negative examples, labeled as such.
- False negatives (FN): number of positive examples, labeled as negative.

We used the following measures of the prediction quality:

- Accuracy measures how often the classifier makes the correct prediction. It is the ratio between the number of correct predictions and the total number of predictions.

$$ACC = \frac{TP + TN}{TP + TN + FP + FN}$$

- Recall is also known as sensitivity. It is the fraction of positive instances that are correctly classified as positive (rate of true positives).

$$REC = \frac{TP}{TP + FN}$$

- Precision is the fraction of true positives over the number of all reported positives.

$$PRE = \frac{TP}{TP + FP}$$

- F-measure is the harmonic mean of the precision and the recall

$$F = 2 \cdot \frac{PRE \cdot REC}{PRE + REC}$$

- Specificity is the fraction of true negatives over the number of all negatives.

$$SPE = \frac{TN}{FP + TN}$$

- Area under the ROC curve (AUC). The ROC curve shows how the classifier can sacrifice the true positive rate (recall or sensitivity) for the false positive rate (1-specificity) by plotting the TP rate to the FP rate. In other words, it shows you how many correct positive classifications can be gained as you allow for more and more false positives. As an example, in Figure 1 we report the ROC curve for the naive Bayes classifier with the ordinal attributes. Its area under the curve is 0.782.

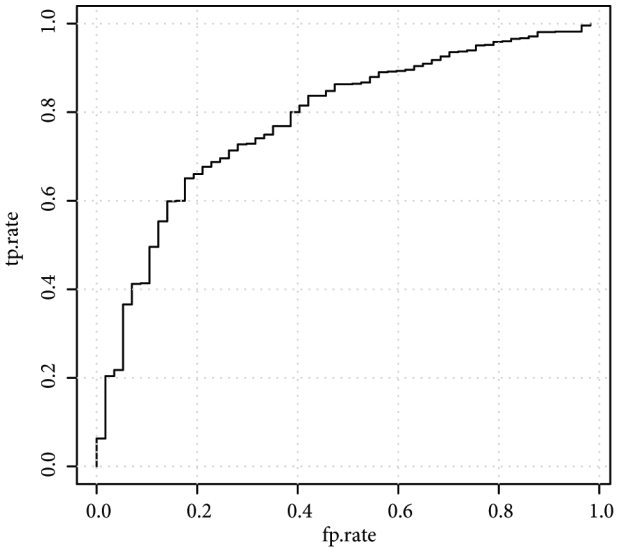

Figure 1. ROC for the naive Bayes classifier with ordinal attributes.

3.7. Results of experiments

In Table 2, we compare the results of different classifiers on different versions of data. The C4.5 classifier with D.DISCR has the highest accuracy of 0.942, its recall and precision are also among the best achieved. However, its area under the ROC curve is very low, only 0.371, which suggests that this classifier cannot be satisfactorily tuned if we want to sacrifice precision to recall or vice versa.

The contribution of attribute selection method (CfsSubsetEval method of Weka) to the performance of models was pretty good where the accuracy was improved in general except C4.5 with D.ORD, and LOG.REG with D.ORD and D.BIN. Moreover, the AUC and F-measure were improved in most of the models. Moreover, Precision, recall, and F-measure values of almost all methods are very low because of imbalanced data where we predict patients who will not survive.

In Figure 2, we present the tree structure of the C4.5 learned from the discrete data. It has achieved the highest accuracy from all tested classifiers. Its structure is surprisingly simple. If the patient is Czech then it is predicted to survive if the patient is Syrian then the LDL cholesterol value should be checked. If it is below 4.78 then the patient is predicted to survive, otherwise, if LDL cholesterol value is between 4.78 and 6.28 then it depends on the Syrian hospital in which he/she is treated. If he/she is treated in the public hospital (SYR1) then he/she dies; if he/she is treated in the private one (SYR2) then he/she survives. If his/her LDL cholesterol values are higher than 6.28 then he/she dies (no matter what Syrian hospital he/she is treated in). The simplicity of the C4.5 classifier is in line with the general recommendation that in order to avoid the overfitting of training data the models should be as simple as possible. This is probably the best we can learn from data but most probably it oversimplifies the reality. More data would be needed.

The highest AUC was achieved by naive Bayes classifier with the ordinal attributes. The highest value of F-measure was achieved by BN.K2 with discrete attributes selected by the method CfsSubsetEval method of Weka [13]. The learned BN model is actually also a naive Bayes model, see Figure 3. We can conclude that there is no single winner—a classifier that would be the best in terms of all considered criteria. Moreover, the classifiers differ in what variables they consider to be important for AMI mortality prediction. We believe

Table 2. Results of experiments.

Classifier	Criteria	D.ORD	D.ORD.AS	D.DISCR	D.DISCR.AS	D.BIN	D.BIN.AS
Naive Bayes	ACC	0.855	0.925	0.860	0.914	0.875	0.911
	AUC	0.782	0.722	0.744	0.781	0.695	0.717
	Recall	0.439	0.158	0.351	0.368	0.246	0.140
	Precision	0.234	0.450	0.215	0.396	0.203	0.276
	F-measure	0.305	0.234	0.267	0.382	0.222	0.186
C4.5	ACC	0.935	0.933	0.942	0.921	0.926	0.927
	AUC	0.527	0.621	0.371	0.627	0.528	0.273
	Recall	0.263	0.105	0.246	0.123	0.070	0.035
	Precision	0.625	0.750	0.875	0.368	0.444	0.333
	F-measure	0.370	0.185	0.384	0.184	0.121	0.063
LOG.REG	ACC	0.930	0.925	0.907	0.919	0.926	0.919
	AUC	0.746	0.755	0.622	0.746	0.675	0.746
	Recall	0.140	0.018	0.193	0.140	0.070	0.140
	Precision	0.571	0.250	0.289	0.364	0.364	0.364
	F-measure	0.225	0.033	0.232	0.203	0.118	0.203
NB-tree	ACC	0.932	0.936	0.914	0.920	0.913	0.920
	AUC	0.658	0.480	0.701	0.726	0.701	0.726
	Recall	0.211	0.228	0.228	0.088	0.070	0.088
	Precision	0.600	0.684	0.310	0.313	0.211	0.313
	F-measure	0.312	0.342	0.263	0.137	0.105	0.137
BN.K2	ACC	NA	NA	0.886	0.918	0.900	0.926
	AUC	NA	NA	0.750	0.775	0.687	0.671
	Recall	NA	NA	0.316	0.368	0.193	0.105
	Precision	NA	NA	0.265	0.429	0.256	0.462
	F-measure	NA	NA	0.288	0.396	0.220	0.171
BN.TAN	ACC	NA	NA	0.908	0.925	0.904	0.927
	AUC	NA	NA	0.721	0.768	0.653	0.642
	Recall	NA	NA	0.193	0.228	0.088	0.053
	Precision	NA	NA	0.297	0,464	0.179	0.333
	F-measure	NA	NA	0.234	0.306	0.118	0.091

```
Hospital <= 0: 0 (603.0/36.0)
Hospital > 0
|   LDLC <= 4.78: 0 (157.86/6.0)
|   4.78 < LDLC <= 6.28
|   |   Hospital <= 1: 1 (12.95/2.95)
|   |   Hospital > 1: 0 (9.0/1.0)
|   LDLC > 6.28: 1 (4.18/0.18)
```

Figure 2. Decision tree C4.5 learned from D.DISCR has the highest accuracy 0.943 of all tested models.

that it is worth learning diverse classifiers since it may help medical specialists to get a deeper insight into the modeled problem.

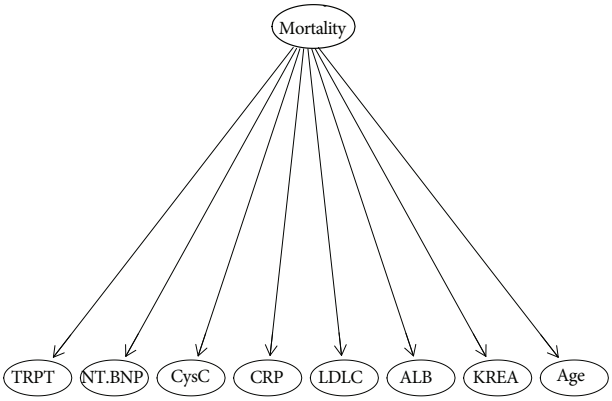

Figure 3. BN learned by BN.K2

4. Dealing with incomplete and imbalanced data

As we can see from Section 2, our dataset contains incomplete and imbalanced data. In [11] we presented an idea to develop TAN [5] to handle incomplete and imbalanced data (Algoritms 1 and 2), where the conditional mutual information (CMI) is defined as:

$$I(X,Y|Z) = \sum_{\mathbf{x,y,z}} f(\mathbf{x,y,z}) \log \frac{f(\mathbf{z})f(\mathbf{x,y,z})}{f(\mathbf{x,z})f(\mathbf{y,z})},$$

where the sum is only over $\mathbf{x,y,z}$ such that $f(\mathbf{x,z}) > 0$ and $f(\mathbf{y,z}) > 0$.

Algorithm 1 TAN For Incomplete Data

Read $D = \{\mathbf{u}_1, \ldots, \mathbf{u}_N\}, \mathbf{u}_m = (a_1, \ldots, a_n, c), m \in \{1, \ldots, N\}$
procedure CMI($A_i, A_j, C\}$) ▷ // Conditional Mutual Information
    $\overline{D} = \{\overline{\mathbf{u}}_1, \ldots, \overline{\mathbf{u}}_N\}, \overline{\mathbf{u}}_m = (a_i, a_j, c), m \in \{1, \ldots, N\}$, such that $\mathbf{u}_m = (a_1, \ldots, a_n, c) \in D$
    Foreach $\overline{\mathbf{u}}_m \in \overline{D}$
        If$(a_i == NA | a_j == NA)$
            Delete $\overline{\mathbf{u}}_m$ from $\overline{D}$
    endfor
    Compute $I_p = I(A_i, A_j|C)$ from $\overline{D}$
    return I_p
Endprocedure
Compute $I_p = I(A_i, A_j|C))$ between each pair of attributes, $i \neq j$, using the Procedure CMI.
Build a complete undirected graph in which the vertices are the attributes $A_1, A_2, \ldots, A_n$. Annotate the weight of an edge connecting A_i to A_j by $I_p = I(A_i, A_j|C))$.
Build a maximum weighted spanning tree.
Transform the resulting undirected tree to a directed one by choosing a root variable and setting the direction of all edges to be outward from it.
Construct a TAN model by adding a vertex labeled by C and adding edges from C to all other nodes in the graph.

Algorithm 2 Procedure for WeightMatrix computation with incomplete and imbalance data

var
  M The number of samples for the majority class
  N The number of samples for the minority class
  D_T All instances of the majority class, $D_T \subset D$
  D_F All instances of the minority class, $D_F \subset D$
integer division $L = M/N$
Divide D_T to L parts, $D_{T_k}, k \in \{1, \ldots, L\}$
Foreach D_{T_k}
  $D_k = D_{T_k} \cup D_F$
EndForeach
Compute WeightMatrix $\mathbb{I}_{p_k}$ foreach D_k
$\hat{\mathbb{I}}_p =$ the average of $\mathbb{I}_{p_k}, k \in 1, \ldots, L$ ▷ // $\hat{\mathbb{I}}_p$ is the final WeightMatrix

In a similar way, we can create a procedure that enables the Chow–Liu algorithm to deal with incomplete data, where a normal Chow–Liu algorithm [12] just deals with complete data. The procedure is shown in Algorithm 3, where the mutual information (MI) is defined as:

$$I(X,Y) = \sum_{\mathbf{x},\mathbf{y}} f(\mathbf{x},\mathbf{y}) \log \frac{f(\mathbf{x},\mathbf{y})}{f(\mathbf{x}) f(\mathbf{y})},$$

where the sum is only over $\mathbf{x},\mathbf{y}$ such that $f(\mathbf{x}) > 0$ and $f(\mathbf{y}) > 0$.

Algorithm 3 Procedure Chow–Liu for incomplete data

Read $D = \{\mathbf{u}_1, \ldots, \mathbf{u}_N\}, \mathbf{u}_m = (a_1, \ldots, a_n), m \in \{1, \ldots, N\}$
procedure MI(A_i, A_j) ▷ // Mutual Information
  $\overline{D} = \{\overline{\mathbf{u}}_1, \ldots, \overline{\mathbf{u}}_N\}, \overline{\mathbf{u}}_m = (a_i, a_j), m \in \{1, \ldots, N\}, a_i, a_j \in \mathbf{u}_m$, such that $\mathbf{u}_m = (a_1, \ldots, a_n) \in D$
  Foreach $\overline{\mathbf{u}}_m \in \overline{D}$
    If($a_i == NA | a_j == NA$)
      Delete $\overline{\mathbf{u}}_m$ from $\overline{D}$
  endfor
  Compute $I_p = I(X,Y)$ from $\overline{D}$
  return I_p
Endprocedure
Compute $I_p = I(A_i, A_j)$ between each pair of attributes, $i \neq j$, using the Procedure MI.
Build a complete undirected graph in which the vertices are the attributes $A_1, A_2, \ldots, A_n$. Annotate the weight of an edge connecting A_i to A_j by $I_p = I(A_i, A_j)$.
Build a maximum weighted spanning tree.
Transform the resulting undirected tree to a directed one by choosing a root variable and setting the direction of all edges to be outward from it.

The idea behind Algorithms 1 and 3 is that we think if we use more data then the estimates of mutual information and conditional mutual information are more reliable.

4.1. Results

We will refer to TAN and Chow–Liu which deal with incomplete and imbalanced data as TANI and CLI. We used 10-fold cross-validation to compare how the results change. The results are summarized in Table 3. We

compare the results of our methods with those of TAN in bnclassify[1] we will refer to it as (TB), Chow–Liu [12] (we will refer to it as CL), EM algorithm [19] for Chow–Liu using Hugin[2] (we will refer to it as EMCL), normal TAN [5], and [20] (this algorithm deals with TAN based on the EM principle, where they have proposed an adaptation of the learning process of tree augmented naive Bayes classifier from incomplete data, where any variable can have missing values in the dataset) (we will refer to it as FL), and SMOTE algorithm [21] for TAN (we will refer to it as ST), on two versions of dataset (binary and discrete attributes). For measures of the prediction quality, we use log-likelihood (LL) and AUC. Moreover, we use the 10-fold cross-validation as the model evaluation method. Algorithm TANI with D.BIN has achieved the highest AUC (ROC = 0.953) and the highest LL with −2744.4279. The results of Algorithm 1 is better than those of the normal TAN algorithm in both datasets D.DISCR and D.Bin. However, ST has achieved the second highest LL with D.DISCR (LL= −6043.0785) but the AUC is (ROC = 0.802), also its ROC is better than the ROC(s) of Algorithm 1 with D.DISCR and Algorithm 3 with both datasets. We can conclude that the TANI is a single winner with D.Bin.

Table 3. BN results.

		D.DISCR	D.Bin
TB	AUC	0.804	0.448
	LL	–7340.9414	–7497.461
FL	AUC	0.77081	0.871
	LL	–11319.6	–6368.38
ST	AUC	0.802	0.818
	LL	–6043.0785	–7168.8239
CL	AUC	0.723	0.69
	LL	–12763.4	–6396.2673
EMCL	AUC	0.6917	0.71
	LL	–11508.2	–8869.63
BN.TAN	AUC	0.62	0.67
	LL	–11321.406	–6368.453
Algo1	AUC	0.77	0.93
	LL	–19914.4937	–2819.3032
Algo3	AUC	0.75	0.73
	LL	–6145.0196	–2755.7778
TANI	AUC	0.82	0.953
	LL	–9393.4688	–2744.4279
CLI	AUC	0.476	0.8956
	LL	–6317.81655	–2953.3373

[1] Comments on bnclassify package runtimes (2015). R[online]. Website: https://cran.r-project.org/web/packages/bnclassify [accessed 10 May 2018]

[2] Hugin Expert A/S (2010). Wibsite: http://www.hugin.com [accessed 20 May 2018]

5. Quality of classifiers tested on artificial data

The data we have is not big enough to have a very good result. Where TAN [5] is a reliable model and has been tested on many datasets, we decided to use the model BN.TAN [5]; its results are presented in Table 2 to generate a sequence of datasets with those sizes (3000, 5000, 7000, and 10,000) and 10% missing completely at random, with 26 attributes including the class in two different types of probability (basic probability distribution and binary distribution) to test the Algorithms (Algo 1, TANI, and FL [20]). See Figures 4 and 5. We can see that our Algorithm 1 is better than the others, and TANI does not seem good with the big binary datasets.

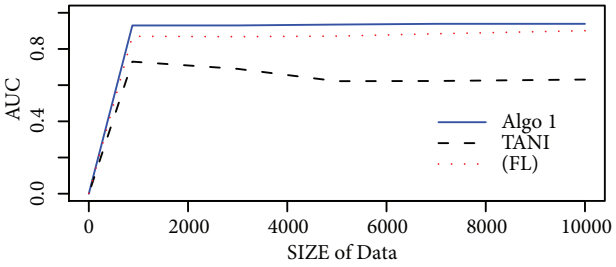

Figure 4. AUC quality of classifiers (D.Bin).

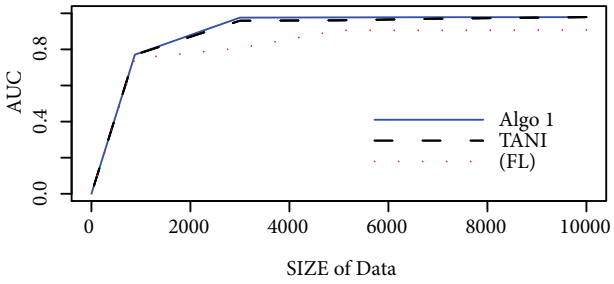

Figure 5. AUC quality of classifiers (D.DISCR).

6. Conclusion

We used medical data on patients with AIM to compare the results of (a) classification models and (b) Bayesian networks modeling the relations found in data. Although the conclusions might seem to be specific only for the data used here, we also report general observations.

In principle, the BN learning algorithms are able to discover the mediated correlation, since they test not only pairwise independence but also the conditional independence given values of other variables.

Bayesian networks are a tool of choice for reasoning in uncertainty, with incomplete data. However, often, Bayesian network structural learning only deals with complete data. We have proposed here an adaptation of the learning process of the Chow–Liu and TAN from incomplete and imbalanced datasets. These methods have been successfully tested on our dataset. We have seen that the TANI algorithm is a single winner with D.Bin.

Acknowledgments

This work was supported by the Czech Science Foundation through projects 16-12010S, and the student grant CTU SGS16/253/OHK3/3T/14.

References

[1] Murphy KP. Machine Learning: A Probabilistic Perspective. Canada: The MIT Press, 2012.

[2] Duda R, Hart P, Stork DG. Pattern Classification. USA: John Wiley and Sons, 2001.

[3] Hastie T, Tibshirani R, Friedman J. The Elements of Statistical Learning. USA: Springer, 2009.

[4] Koller D, Friedman N. Probabilistic Graphical Models: Principles and Techniques. Canada: MIT Press, 2009.

[5] Friedman N, Geiger D, Goldszmidt M. Bayesian network classifiers, Machine Learning 1997; 29(2): 131-163.

[6] Onisko A, Druzdzel M, Wasyluk H. A Bayesian network model for diagnosis of liver disorders. In: Proceedings of the Eleventh Conference on Biocybernetics and Biomedical Engineering; Warsaw, Poland; 1999. pp. 842-846.

[7] Blanco R, Inza I, Naga PL. Feature selection in Bayesian classifiers for the prognosis of survival of cirrhotic patients treated with TIPS. Journal of Biomedical Informatics 2005; 38 (05): 507-543.

[8] Heckerman D, Horvitz E, Nathwani B. Toward normative expert systems: Part I. The Pathfinder project. Methods of Information in Medicine 1992; 31: 90-105.

[9] Krumholz HM, Normand SLT, Galusha DH, Mattera JA, Rich AS et al. Risk-Adjustment Models for AMI and HF 30-Day Mortality, Methodology, USA: Harvard Medical School, Department of Health Care Policy, 2007.

[10] Vomlel J, Krua a k H, Tůma P, Přeček J, Hutyra M. Machine learning methods for mortality prediction in patients with ST elevation myocardial infarction. In: Proceedings of The Nineth Workshop on Uncertainty Processing WUPES'12; Czech Republic; 2012. pp. 204-213.

[11] Salman I, Vomlel J. A machine learning method for incomplete and imbalanced medical data. In: Proceedings of the 20th Czech-Japan Seminar on Data Analysis and Decision Making Under Uncertainty; Pardubice, Czech Republic; 2017. pp. 188-195.

[12] Chow C, Liu C. Approximating discrete probability distributions with dependence trees. IEEE Transactions on Information Theory 1968; 14: 462-467.

[13] Hall M, Frank E, Holmes G, Pfahringer B, Reutemann P et al. The WEKA data mining software: An update, ACM Sigkdd Explorations 2009; 11 (1): 10-18.

[14] Quinlan R, Kaufmann M. C4.5: Programs for machine learning, Machine Learning 1993; 29(2): 131-163.

[15] Cessie Sle, Houwelingen JC. Ridge estimators in logistic regression. Applied Statistics 1992; 41(1): 191-201.

[16] Duda RO, Hart PE. Pattern classification and scene analysis. Wiley-Interscience 1973; 30(1): 106-110.

[17] Kohavi R. Scaling up the accuracy of Naive-Bayes classifiers: A decision-tree hybrid. In: Proceedings of Second International Conference on Knowledge Discovery and Data Mining; Portland, Oregon, USA; 1996. pp. 202-207.

[18] Cooper GF. A Bayesian method for the induction of probabilistic networks from data, Machine Learning 1992; 9(4): 309-347.

[19] Cohen Ira, Cozman FG, Sebe N, Marcelo C, Huang TS. Semi-supervised learning of classifiers: Theory, algorithms and their application to human-computer interaction. IEEE-Transactions on Pattern Analysis and Machine Intelligence 2004; 26(12): 1553-1568.

[20] Francois OCH, Leray P. Learning the tree augmented Naive Bayes classifier from incomplete datasets. In: Third European Workshop on Probabilistic Graphical Models (PGM); Prague, Czech Republic; 2006. pp. 91-98.

[21] Chawla N, Bowyer K, Hall L, Kegelmeyer W. Synthetic minority over-sampling technique. Journal of Artificial Intelligence Research 2002; 11(16): 321-357.