Abstract

A prognostic Bayesian network (PBN) is new type of prognostic model that implements a dynamic, process-oriented view on prognosis. In a companion article, the rationale of the PBN is described, and a dedicated learning procedure is presented. This article presents an application hereof in the domain of cardiac surgery. A PBN is induced from clinical data of cardiac surgical patients using the proposed learning procedure; hospital mortality is used as outcome variable. The predictive performance of the PBN is evaluated on an independent test set, and results were compared to the performance of a network that was induced using a standard algorithm where candidate networks are selected using the minimal description length principle. The PBN is embedded in the prognostic system ProCarSur; a prototype of this system is presented. This application shows PBNs as a useful prognostic tool in medical processes. In addition, the article shows the added value of the PBN learning procedure.

Keywords: Prognostic model; Bayesian network; Health care process; Cardiac surgery

1. Introduction

In a companion article, we have introduced the prognostic Bayesian network (PBN) as new type of prognostic model; we presented a dedicated learning procedure to induce these networks from clinical data and described prognostic uses of PBN in clinical practice [1]. This article presents an application hereof in the clinical domain of cardiac surgery.

Cardiac surgery is a complex medical procedure that is applied to patients with severe insufficiency of the cardiac functioning. Most of cardiac surgical interventions involve coronary artery bypass grafting (CABG), repair or replacement of heart valves, aorta surgery, or a combination of these procedures. The procedures are embedded in a health care process that includes the stages of pre-assessment, operation, and recovery, and involves highly specialized clinical personnel, such as a cardiologist, cardiac surgeon, anaesthetist, and intensive care unit (ICU) physicians.

During the operation and the postoperative stay at the ICU and nursing ward, several complications may occur that extend the operation time, delay the recovery process, and may lead to permanent disabilities or death. Death is an important clinical endpoint in the care process of cardiac surgery. The patient’s prognosis for this outcome is used in decision making prior to and during the medical
procedure. In addition, the outcome is used to evaluate whether the procedures have been applied successfully. Since the mid-1980s, a large number of prognostic models have been developed for the mortality outcome, with the EuroSCORE as predominant model [2]. Most models applied logistic regression to assess preoperative risk.

We developed a PBN for this clinical domain using the PBN learning procedure. In Section 2, the patient data that are used are introduced and the data preprocessing is described. Section 3 subsequently describes the results of applying the PBN learning procedure to the data. Furthermore, we validated the resulting PBN and compared its performance to a network that we learned using the standard search and score algorithm where candidate networks are scored using the minimal description length principle [3] as implemented in the software package BayesiaLab (Section 4). To facilitate clinicians’ interaction with the network, we embedded the PBN in a prototypical prognostic system (ProCarSur); the system is presented in Section 5. We conclude the article with a discussion and conclusions in Section 6.

2. Data and data preprocessing

The study population includes 10,147 patients who underwent cardiac surgery in the Amphia Hospital, a teaching hospital in Breda, the Netherlands, between January 1998 and November 2004. The data set contains preoperative patient characteristics, details of the operative procedure, and physiological and laboratory variables measured during the first 24 h of postoperative ICU stay; all variables included in the EuroSCORE [4], SAPS II score [5], and APACHE II score [6] are in the data set. Furthermore, the data set included length of ICU stay, and binary variables that describe postoperative complications that frequently occur in cardiac surgery, and death during hospitalization; for patients who expired, the data set includes time of death.

Hospital mortality (hospmort) was used as the outcome variable of the PBN with operative mortality (ORmort) and postoperative mortality (postORMort) being subsidiary outcome variables. Among the 10,147 patients, 277 (2.74%) patients died during hospitalization: 66 patients died in the operation room and 211 patients died in the postoperative phase of the process. The data set contained missing values for the variables that describe death and time of death for 33 patients (0.33%); these patients were excluded from the data set. Furthermore, the data set contained variables that were not recorded from January 1998 but from later times, and variables with large amounts of missing values. We excluded all variables from the data set that were still not recorded in January 1999, in addition to the variables that contain more than 10% missing values in the years of recording.

Subsequently, the data set was randomly divided into a training set (n = 6778) and a test set (n = 3336); the training set was used for data preprocessing, variable selection, and PBN learning, the test set for network validation. In the training set, 189 patients expired during hospitalization: 42 patients died in the operation room and 147 patients died in the postoperative phase of care. In the test set, 88 patients expired during hospitalization: 24 patients died in the operation room and 64 patients died in the postoperative phase of care.

The following steps were performed to preprocess the training data. First, we discretized all continuous variables in five equally sized categories using the quintile values of their distribution to prevent for overfitting in PBN learning. Second, we imputed all missing values with the majority class value for the included discrete/binary variables, and the middle category (i.e., .4 and .6 percentiles of the empirical distribution) for discretized continuous variables. No values were imputed for the year 1998 for variables that were recorded since 1999. We excluded all 997 patients who underwent surgery in 1998 from the training set in all analyses in which these variables were involved. Furthermore, imputation was only performed for patients that were at risk during the phase in which the variables were measured. So, no values were imputed for the postoperative variables of the 42 patients in the training set who died during the intervention. These patients were excluded from the training set in all analyses for the postoperative variables.

In this case study of the PBN learning procedure, we used the training set to select a limited set of variables that represent the different stages of care from the available data. From each stage, variables were selected with a high predictive value with respect the final outcome variable (hospmort); the predictive value was quantified in terms of the 10-fold cross validation information gain (ΔI) on the training set. Variables that represent a prognostic score, such as the EuroSCORE, were excluded, because our objective was to model the mutual relationships of the underlying variables with process and outcome variables. The resulting set of variables was subsequently inspected by the clinical experts involved (PR, EdJ, and BdM). They recommended inclusion of the preoperative variables bmi and diabetes. Physiological and laboratory data of the first 24 h ICU stay were available in the form of summary values as used in the SAPS II score, i.e., maximal and minimal values. The creatinine value is generally measured for a low number of times during ICU stay. The maximal and minimal creatinine value for a 24 h period are therefore strongly related or even similar. For this reason, we only included the variable creatmax in the network. Table 1 shows the final set of 22 selected preoperative and process variables, the percentage of missing values and the information gain with respect to hospital mortality in the training set; the five complication variables have been recorded since January 1999.

The test set was preprocessed by discretizing all included continuous variables using the same thresholds as were used on the training set. We performed no imputation in the test set, as Bayesian networks allow making predictions
on incomplete data. Patients with missing values in the postoperative complication variables were excluded from the test set during network validation for the complication variables, because the predicted probability for the variables could not be evaluated for these patients; 28 patients had missing values for the variable ICUlos24h, and, respectively, 27, 31, 45, 32, and 31 patients for the variables neurcomp, pulmcomp, cardcomp, mof, and infect, in addition to all patients of the year 1998 (488 patients).

The PBN learning procedure assumes the predictor variables and the subsidiary outcome variables to be ordered in a number of temporal ‘strata’ defined by the time and order in which the variables are observed; this was done to ensure that the directions of arcs in the network are consistent with the flow of time. The stages of preassessment, intervention, and recovery roughly define an ordering of the selected variables, but when considering the time and order of observation, a larger set of strata can be defined. The strata are shown in Table 2. The five complication variables are in the highest stratum in addition to the subsidiary outcome variable postORmort (postoperative mortality), and the variable age is in the lowest stratum. Variables are in the same temporal stratum when their values are determined within a relatively short period and not always in the same order. We used the nine strata in PBN learning.

### 3. PBN learning from local models

We induced a PBN using the dedicated learning procedure that is presented in the companion article. In the procedure, the network is composed of a collection of local supervised learning models that are recursively learned from the data. The procedure optimizes performance of the network’s primary task, outcome prediction, and handles the fact that patients may die during earlier parts of the process, and ‘drop out’ of the process.
3.1. Class probability trees

In the application of the learning procedure to the cardiac surgical data, we used the method of class probability trees from the tree building methodology Classification and Regression Trees (CART) of Breiman et al. [7] for local model building. Compared to classification trees, class probability trees estimate the (conditional) probability distribution on the outcome classes for a given case, instead of predicting the most probable outcome class. So, the terminal nodes of a class probability tree contain probability distributions. When building class probability trees, the data set is recursively partitioned into subsets by selecting features that contribute most to identifying homogeneous subsets (in terms of the Gini index [8]) with respect to the features that contribute most to identifying homogeneous subsets of the Gini index [8] with respect to the outcome. The feature subset selection is thus incorporated in the tree building algorithm.

All class probability tree models were developed using the S-PLUS library Rpart [9], which is an implementation of CART [7]. The optimal tree size was determined by minimizing the 10-fold cross validation error.

3.2. PBN learning

3.2.1. Step I

We started network learning with a graph, consisting 22 nodes that represent the predictor variables and three nodes to represent the (subsidiary) outcome variables. The graph contains two arcs to represent the sub-outcomes ORmort and postORmort as parent nodes of the global outcome hospmort. In the first iteration of the procedure, a class probability tree was developed for the sub-outcome variable postoperative mortality (postORmort); the 22 predictor variables were used as potential predictive features. The variables mof (multiple organ failure), ecctime (duration of the extracorporeal circulation), and infect (infection) were selected as predictors in the tree model. Therefore, three arcs were drawn in the graph from the selected variables to the outcome variable. It is valuable to note that all 22 variables were earlier found to have predictive value for death during hospital stay as shown by their information gain $\Delta I$ for this outcome in Table 1. However, when combining them in a multivariate tree analysis, only multiple organ failure, duration of the extracorporeal circulation, and infection appear as predictors in the tree model.

The class probability tree for postoperative mortality is shown in Fig. 1. This tree model shows that the risk of postoperative mortality is high for patients with occurrence of multiple organ failure, especially for patients with a relatively short duration of the extracorporeal circulation (probability of 0.807). The occurrence of multiple organ failure and the related high risk of mortality in this latter patient group are not explained by a complicated operative course (long duration of the extracorporeal circulation), but is probably caused during the recovery process itself. For the patient with multiple organ failure and a complicated operative course, a lower risk of mortality is found for those with an infection compared to those without an infection (probability of 0.296 and 0.681, respectively). This finding suggests that complications that are less favorable for patient survival than an infection occurred in this latter patient group. The left part of the tree model shows that patients without multiple organ failure (97.9% of the patient population) have a low risk of postoperative death (probability of 0.010).

3.2.2. Step II

The selected features for the sub-outcome postoperative mortality were subsequently enqueued in a priority queue with the information gain $\Delta I$ for hospital mortality as priority value (Table 1). So, after the first iteration, the queue had the following content: $Q = \{(\text{mof}, 0.0466), (\text{ecctime}, 0.0154), (\text{infect}, 0.0104)\}$. Therefore, in the second iteration of the learning procedure, the variable mof was dequeued from the priority queue. For this variable, all other variables, with exception of the outcome hospital mortality, were used as potential predictive features during tree induction. The variables neurocomp (neurological complication), infect (infection), pulmcomp (pulmonary complication), fiO2 (fraction inspired oxygen), meanbpmax (maximal mean blood pressure), meanbpmin (minimal mean blood pressure), ecctime (extracorporeal circulation time), temp (temperature), and pulmhyp (pulmonary hypertension) were selected as predictors in the tree model for this variable. From each of them an arc was added to the graph to the variable mof. The graph structure that was created thus far is shown in Fig. 2. The selected variables for mof were subsequently enqueued in the priority queue except for the variables infect and ecctime, as these variables were already enqueued in the initial step. Subsequently, a class probability tree was developed for the variable ecctime (extracorporeal circulation time) with all variables from preceding strata as potential predictive features.
In the following iterations, a feature subset was selected for each predictor variable that appeared in the priority queue. The data of all patients who survived the operation were used in the iterations for the sub-outcome postORmort and all postoperative variables ($n = 6736$); in the iterations for the operative and preoperative variables, we used the data of all patients ($n = 6778$).

3.2.3. Step III

The next step in the procedure was to assess the set of parent nodes of the subsidiary outcome ORmort, and to build the associated local model. This outcome variable represents death during surgery which is the reason for dropout from the care process. Only 42 (0.62%) patients from 6778 in the training set expired during surgery. This extreme unbalance in classes rendered it impossible to build a tree model (other than the trivial ‘single node’ tree).

Various methods to cope with class imbalance have been described in the literature [10]. Here, we applied a simple, ad hoc solution that is based on the problem at hand. In our training set, another 147 patients (2.17%) died after surgery, at the ICU or at the nursing ward. It occurs frequently that these patients have a troublesome operation and die during the next day. For this reason, we decided to “borrow strength” from the sub-outcome postORmort in the analysis. So, we used data from all deaths in the training set to induce a tree for the sub-outcome ORmort, including those who died post operatively. The estimated probabilities in the resulting tree model were subsequently rescaled and then checked for their validity to predict operative death.

In the following iterations, a feature subset was selected for each predictor variable that appeared in the priority queue. The data of all patients who survived the operation were used in the iterations for the sub-outcome postORmort and all postoperative variables ($n = 6736$); in the iterations for the operative and preoperative variables, we used the data of all patients ($n = 6778$).

3.2.3. Step III

The next step in the procedure was to assess the set of parent nodes of the subsidiary outcome ORmort, and to build the associated local model. This outcome variable represents death during surgery which is the reason for dropout from the care process. Only 42 (0.62%) patients from 6778 in the training set expired during surgery. This extreme unbalance in classes rendered it impossible to build a tree model (other than the trivial ‘single node’ tree).

Various methods to cope with class imbalance have been described in the literature [10]. Here, we applied a simple, ad hoc solution that is based on the problem at hand. In our training set, another 147 patients (2.17%) died after surgery, at the ICU or at the nursing ward. It occurs frequently that these patients have a troublesome operation and die during the next day. For this reason, we decided to “borrow strength” from the sub-outcome postORmort in the analysis. So, we used data from all deaths in the training set to induce a tree for the sub-outcome ORmort, including those who died post operatively. The estimated probabilities in the resulting tree model were subsequently rescaled and then checked for their validity to predict operative death.

In the following iterations, a feature subset was selected for each predictor variable that appeared in the priority queue. The data of all patients who survived the operation were used in the iterations for the sub-outcome postORmort and all postoperative variables ($n = 6736$); in the iterations for the operative and preoperative variables, we used the data of all patients ($n = 6778$).

3.2.4. Step IV

The variable cardcomp (cardiac complication) was not selected in any feature subset and therefore did not have any incoming or outcome arcs after all previous steps were carried out. A possible explanation is that the variable is statistically independent of all other variables in the network. From the univariate analysis, however, correlation with the outcome variable was known ($\Delta I = 0.009$, Table 1). Another explanation is that the variable is conditionally independent of the other complication variables and postoperative mortality variables given variables that are in lower strata. Using the procedure, the latter variables are then selected for the complication and mortality variables, and their feature subset is

![Figure 2](image)

**Figure 2.** The graph structure after feature selection and model building for the subsidiary outcome postoperative mortality and the variable multiple organ failure.

![Figure 3](image)

**Figure 3.** The class probability tree model for the subsidiary outcome ‘operative mortality’. Each leaf node is labeled with the rescaled estimated probability and, between brackets, the observed frequency and the number of observations in the corresponding subgroup of the training set. The threshold of 50 min represents the threshold between the fourth and fifth quintile of the discretized variable eccacctime.
subsequently selected from lower strata; the variable cardcomp thus remains unselected.

To discover the dependencies, we concluded the procedure with developing a class probability tree for this variable; all variables were used as potential predictive features with exception of postORmort (postoperative mortality). The variables ICUlos24h (ICU length of stay longer than 24 h), ecctime (duration of the extracorporeal circulation), emerg (emergency), and surtype (surgery type) were selected. We subsequently added arcs to the graph to represent that these variables form the parent nodes of the variable cardcomp; this variable has no child nodes.

3.2.5. Step V

In the resulting graph, the deterministic relationships between operative mortality ORmort and the postoperative (sub-outcome) variables that describe the irrelevancy of the postoperative variables in case of operative death were still lacking. To complete the network, we added these relationships in this final step by drawing arcs and extending the corresponding local conditional probability models of the variables. Fig. 4 shows the structure of the resulting PBN.

4. Network validation

The predictive performance of the PBN was validated in terms of its ability to make unbiased estimates of outcome probabilities (calibration) and to separate positive and negative outcomes (discrimination). The validation procedure included the performance assessment of the networks for the (sub-)outcomes that describe mortality during the process, the variable ICUlos24h (ICU length of stay longer than 24 h), and the five variables that represent postoperative complications; the postoperative (sub-outcome) variables were evaluated only on data of the 3312 patients who survived the operation. Validation was performed at two prediction times: (i) during the preoperative stage, and (ii) at ICU admission. For the outcome variables operative and hospital mortality (ORmort and hospmort), performance was only validated at the first prediction time. Prediction of the first outcome at ICU admission is not meaningful, while prediction of the latter outcome at ICU admission is, by definition, equal to prediction of postoperative mortality (postORmort).

We validated the calibration of network distributions by comparing the expected and observed probabilities of the variables in five equal-sized groups, obtained by ordering the observations in the test set by the expected probability. The differences in these probabilities were statistically tested using the $\chi^2$ distribution with four degrees of freedom. Furthermore, we quantified the discriminative ability of the PBN in terms of the area under the ROC curve (AUC) [11]. The predicted probabilities of the PBN were obtained using the Netica software (Norsys...
Software Corp.\(^1\); all further analyses were performed in S-PLUS (Insightful Corp. Version 6.2 for Windows, Seattle, WA). Table 3 lists the validation results of the PBN for each selected variable and both prediction times, in the third and fourth column, respectively. The table shows a good calibration for the variables ICUlos24h and cardcomp. The network was found to be poorly calibrated for the mortality variables; the expected probabilities for these variables are only in a small range, close to their marginal probabilities. Among the examined variables, the mortality variables and the variable mof had best discrimination.

An important objective of the validation was to verify the effectiveness of our dedicated PBN learning procedure. For this purpose, we induced a network from the training set using a standard algorithm for Bayesian network learning with the software package BayesiaLab,\(^2\) and compared the predictive performance of the networks.

BayesiaLab implements a search and score algorithm where candidate networks are selected using the minimal description length (MDL) principle [3], and the candidate space is traversed with tabu search [12]. As in our own learning procedure, we used the temporal ordering on network variables from Table 2 to constrain the network topology. BayesiaLab assumes the variables in the training set to be relevant for all patients and cannot deal with values that are missing due to patient dropout. Therefore, we imputed the category label ‘I’ in the postoperative variables for patients who died during surgery, denoting irrelevancy of these variables for these patients.

The resulting MDL network is shown in Fig. 5. The network is sparsely connected with 30 arcs compared to 103 arcs in the PBN that was learned from local tree models. The arcs between the postoperative variables partly represent the deterministic relations that exist between operative mortality and the postoperative variables, i.e., their category label ‘I’. Furthermore, only four arcs represent probabilistic relationships between preoperative and operative variables and postoperative variables.

We quantified the calibration and discrimination of this network on the test set using the same statistics as were used for PBN validation. The estimated probabilities of the MDL network were obtained in BayesiaLab; again, all further analyses were performed in S-PLUS. Table 3 lists the results for each selected variable and the two prediction times for the MDL network, in column 5 and 6, respectively. These results show a good calibration for the mortality variable ORmort and the variable cardcomp, and best discrimination for the mortality variables and the variable mof. For the variables ICUlos24h and cardcomp, the same results were found for both prediction times. Because no relationships among these variables and the operative variables were modeled in the MDL, the estimated probabilities did not change when operative data was used for prediction.

When comparing the discrimination statistics of the PBN and the MDL network, higher AUC values were found for the PBN for all variable at both prediction times. We statistically tested the differences in AUC values between the networks using the method of Delong et al.

---

**Table 3**

<table>
<thead>
<tr>
<th>Predicted variable</th>
<th>Prediction time</th>
<th>PBN</th>
<th>MDL network</th>
<th>PBN vs MDL</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Calibration</td>
<td>Discrimination</td>
<td>Calibration</td>
</tr>
<tr>
<td></td>
<td></td>
<td>$\chi^2$</td>
<td>df$^a$</td>
<td>p-value</td>
</tr>
<tr>
<td>nospmort</td>
<td>Preoperative</td>
<td>40.44</td>
<td>4</td>
<td>&lt;0.001</td>
</tr>
<tr>
<td>ORmort$^e$</td>
<td>Preoperative</td>
<td>9.60</td>
<td>2</td>
<td>0.008</td>
</tr>
<tr>
<td>postORmort</td>
<td>Preoperative</td>
<td>32.60</td>
<td>4</td>
<td>&lt;0.001</td>
</tr>
<tr>
<td>ICUlos24h</td>
<td>Preoperative</td>
<td>31.79</td>
<td>4</td>
<td>&lt;0.001</td>
</tr>
<tr>
<td>At ICU admission</td>
<td></td>
<td>5.04</td>
<td>4</td>
<td>0.283</td>
</tr>
<tr>
<td>neurcomp</td>
<td>Preoperative</td>
<td>23.73</td>
<td>4</td>
<td>&lt;0.001</td>
</tr>
<tr>
<td>At ICU admission</td>
<td></td>
<td>21.00</td>
<td>4</td>
<td>&lt;0.001</td>
</tr>
<tr>
<td>pulmcomp</td>
<td>Preoperative</td>
<td>32.22</td>
<td>4</td>
<td>&lt;0.001</td>
</tr>
<tr>
<td>At ICU admission</td>
<td></td>
<td>43.18</td>
<td>4</td>
<td>&lt;0.001</td>
</tr>
<tr>
<td>cardcomp</td>
<td>Preoperative</td>
<td>8.82</td>
<td>4</td>
<td>0.066</td>
</tr>
<tr>
<td>At ICU admission</td>
<td></td>
<td>8.08</td>
<td>4</td>
<td>0.089</td>
</tr>
<tr>
<td>mof</td>
<td>Preoperative</td>
<td>25.61</td>
<td>4</td>
<td>&lt;0.001</td>
</tr>
<tr>
<td>At ICU admission</td>
<td></td>
<td>21.76</td>
<td>4</td>
<td>&lt;0.001</td>
</tr>
<tr>
<td>infect</td>
<td>Preoperative</td>
<td>20.60</td>
<td>4</td>
<td>&lt;0.001</td>
</tr>
<tr>
<td>At ICU admission</td>
<td></td>
<td>24.27</td>
<td>4</td>
<td>&lt;0.001</td>
</tr>
</tbody>
</table>

---

3. df, degrees of freedom.
4. CI, confidence interval.
5. $^a$ The network distribution for operative mortality (ORmort) of the PBN and the MDL network was compared in three and two groups, respectively, due to the low number of different expected probabilities that were assigned to the test cases for this variable.
With performing 16 statistical tests to examine the calibration of the PBN, 16 tests to examine the calibration of the MDL network, and 16 tests to compare the discrimination of both networks, the validation and comparison of the network involves a problem of multiple testing. We therefore used the Bonferroni adjustment for multiple testing, and considered test results to be statistically significant when a \( p \)-value of less than 0.001 was observed.

The results of testing the differences in AUC values are listed in the rightmost column of Table 3. The superiority of the PBN in discriminative ability is found to be statistically significant for the variables cardcomp and mof at ICU admission, and for ICUlos24h at both prediction times. When inspecting the calibration statistics of the PBN and the MDL network, both networks turned out to be poorly calibrated for the majority of variables (low \( p \)-values). Although the calibration statistic of the PBN for the mortality variable ORmort does not prove poor calibration for this variable (\( p \)-value 0.008), the corresponding \( \chi^2 \) value is relatively high compared to the \( \chi^2 \) value of the MDL network for this variable. This suggests that the PBN was overfitted by additionally including the variable eccacctime as parent variable of ORmort. Fig. 6 visualizes the calibration results of both networks as listed in Table 3 for preoperative prediction of two variables and prediction at ICU admission of two variables. Note that the axes of the graphs cover different and limited parts of the interval [0, 1].

The calibration results show that the predicted probabilities of the PBN are underdispersed, especially for the mortality outcomes: the variation in predicted probabilities is smaller than it should be. There are different explanations for this finding. First, it could be caused by the PBN learning procedure. This appears not to be the case as similar results were found for the MDL network. A second possible explanation is that underdispersion is related to sparseness of the available observations. Postoperative predictions can use, by definition, observations on a larger set of variables than preoperative observations, and perhaps therefore the predictions are more dispersed. However, when we only use observations on the three parent variables of the variable postORMort without instantiating any other variable in the network, then the predictions are equally dispersed as when all predictors (preoperative, surgical, and postoperative) are instantiated. This follows from the graphical representation of conditional independence. So, sparseness of observations is also not the explanation per se.

A third possibility is that the validation on an independent set shows that the model is ‘underfit’. In this case, underdispersion of predicted probabilities should not occur on the training set. This possibility requires further scrutiny. And fourth, the underdispersion may be a result of statistical inference through chained probability estimates. It is then directly related to the Bayesian network methodology. When this is true, preoperative predictions of mortality must be less dispersed than postoperative predictions, as they are computed through longer chains of unobserved variables in the network.

To investigate the third and fourth explanations, we performed a closer evaluation of the calibration of PBN and MDL networks for the mortality outcome postORMort. We applied both networks on the training set at four prediction times in the care process: (1) during the preoperative stage, (2) at ICU admission, (3) after 24 h ICU stay, and (4) when all predictor data are known.

The left-hand graph in Fig. 7 shows the calibration performance of the PBN on the training set for postORMort at the different prediction times. By definition, the PBN is perfectly calibrated on the training set when data of the parents variables of this sub-outcome are available for prediction. In that case, it is actually just the local tree model shown in Fig. 1 that is applied to the data; the expected probability in each leaf node of this tree is calculated as
the observed probability in the corresponding patient group in the training set. The figure clearly illustrates a regression of the estimated probabilities to the marginal probability of the outcome as the prediction time is earlier in the process and thus inference is performed through a longer chain of unobserved variables, and does not support the explanation that the network is underfit. The right-hand graph shows similar results for the MDL network, suggesting that the underdispersion of predicted probabilities is directly related to the Bayesian network methodology.

5. The ProCarSur system

In the companion article mentioned before [1], we described six prognostic use cases of PBNs. To support the use of PBNs in clinical practice, we proposed these networks to be embedded in a three-tiered architecture. In the architecture, a PBN is supplemented with a task layer that holds a number of procedures to perform the prognostic use cases of PBNs, and a presentation layer. The task layer translates the user’s clinical information needs to probabilistic inference queries for the network, and the
presentation layer presents the aggregated results of the inferences to the user.

We developed a prototype implementation of a task layer and user interface; together with the cardiac surgical PBN, they make up the ProCarSur system. The task layer was written in Java, and the Netica Java-API was used to access the PBN; the user interface was developed in HTML. Fig. 8 shows a screen shot of the output screen of the system. The screen consists of three panes. The left pane shows the system’s menu. The right upper pane shows the patient profiles as entered by the user, and the right lower pane shows the results of probabilistic inference.

The figure shows the system’s output for a patient case of a 62-year-old non-diabetic patient who has undergone an elective (i.e., non-emergency) coronary artery bypass grafting (CABG) operation; this patient had pulmonary hypertension and a preoperative serum creatinine value of 80 μmol/l. These data were available for prognostic assessment in the preoperative stage of the process; the results hereof for the variable ‘ICU length of stay longer than 24 h’ (ICUlos24h) are visible in the right-hand diagram of the lower right pane. The operation of this patient took relatively long, resulting in a duration of the extracorporeal circulation (ectime) of 197 min and a duration of the extracorporeal circulation while not aortic cross-clamping (eccacctime) of 99 min. This information was used to update the prognosis after the operation. The results are shown in the left-hand diagram in the right lower pane. The prolonged operation time indicates surgical complications and therefore the risk of an ICU stay longer than 24 h has increased from 26% to 55%, an increase with a factor of 2.1. The actual ICU stay of the patient was four days. Finally, the patient was discharged from hospital after five days of recovery at the nursing ward.

6. Discussion and conclusions

In the companion article [1], we proposed PBNs as prognostic tools that implement a dynamic, process-oriented view on prognosis: they explicate the scenarios that lead to different clinical outcomes, and can be used to update predictions when new information becomes available. This article presents an application of PBNs to the domain of cardiac surgery. In this application, PBNs are shown as a useful methodology for prognostic modeling of medical care processes. During demonstrations of ProCarSur to a large number of intended users, such as cardiac surgeons, intensive care physicians, and management staff from different medical centers, the system was received as a valuable tool to support their task of prognosis during patient care and to obtain insight into critical factors in the care process, as well as a useful instrument in the evaluation of care. This case study also shows the added value of the dedicated learning procedure to induce PBNs from clinical data.

From the literature on prognostic modeling in cardiac surgery, the prognostic problems in this domain have been proved to be difficult; this is also the main reason for the demand for prognostic systems to support this task in clinical practice. For instance, an online version of the European system for Cardiac Operative Risk Evaluation (EuroSCORE) [4] is available as the EuroSCORE Interactive Calculator. This system however only allows for prediction of the risk of death prior to the intervention.

Simchen et al. [14] have developed a more general model to predict mortality following cardiac surgery, consisting of three logistic regression submodels for preoperative, oper-

---

3 http://www.euroscore.org/calc.html
ative, and postoperative factors. In the second and third submodel, the predicted risk of the previous submodel is used as a covariate. So, the model allows for updating the predicted preoperative risk of death twice during the process, using operative and postoperative data, respectively. The main difference with the PBN is that these submodels are based on separate regression analyses for the three prediction times. If one would wish to extend the model to additional prediction times or additional outcome variables, the number of separate analyses and submodels would quickly increase. The PBN in contrast is a single, integrated model with the same functionality.

In the application of the PBN learning procedure in the case study, we were confronted with the problem of sparse data for the subsidiary outcome ORmort (operative mortality): no local predictive model could be built for this variable. To overcome this problem, we temporarily borrowed strength from the sub-outcome postORmort in the analysis, and subsequently rescaled the estimated probabilities. This strategy turned out to be valid for this outcome variable. The inclusion of subsidiary outcomes to represent the phenomenon of patient dropout in the network involves this problem of sparse data for the sub-outcomes: by definition, the number of events for each sub-outcome is less than for the final outcome. The extension of the PBN learning procedure with a general strategy to handle this problem is part of future work.

We used the tree induction method in the PBN learning procedure for the transparency of resulting models: the local tree models that composed the PBN were suitable to be discussed with clinical experts. A disadvantage of tree induction methods, however, is their instability: small changes in the data may result in very different tree models [15]. The use of this method in the network learning procedure therefore increases the variance in the structure of the resulting networks. An important cause of the instability is that in tree induction methods, the selection of features is incorporated in the modeling procedure. In the learning procedure, however, also separate methods for feature subset selection and local model building can be used. In addition, more powerful supervised learning methods than tree induction can be used for local model development, such as ensemble learners [15] and artificial neural networks [16].

The PBN for cardiac surgery was developed as a case study of the PBN learning algorithm as proposed in the companion article. To be clinically relevant and trustworthy, several adjustments of the PBN are probably needed. For practical reasons, we included a limited set of discrete variables in the network learning process and used data from a single medical center. We hope to conduct a more rigorous analysis of this prediction problem using a more extensive set of variables and a multi-center data set in the future. In addition, the missing values in the data set were imputed with the majority class value, instead of applying a more advanced method for imputation. Furthermore, no special attention was given to the relatively high amount of missing values that were present in the preoperative variables of emergent patients. This may have biased the PBN learning process resulting in an underestimation of the relatively worse prognosis of emergency patients. Taking account of this type of non-randomly missingness in the data is an important issue for future work.

The capability of the PBN to discriminate between survivors and non-survivors is comparable to existing models in cardiac surgery that have been developed using logistic regression analysis. The developers of the EuroSCORE reported an AUC value of 0.759 on an independent test set [4]. Simchen et al. reported an AUC value of 0.788 for preoperative prediction of the risk of death, and this value increased to 0.853 when operative variables were included in the model [14]. An increase in performance when using operative data for prediction such as reported by Simchen was not observed for the PBN. In their study, however, a more extensive set of operative variables was used, including an important predictive feature that describes the use of an intra-aortic balloon pump. Moreover, the AUC values in that study were obtained on the training set, and are therefore optimistically biased. With respect to calibration, Nashef et al. reported good calibration results for the EuroSCORE on a test set (χ²: 7.5, 10 df, p-value: 0.68) [4]; Simchen et al. did not report on the calibration of their models.

We found that the predicted mortality distributions of the PBN are underdispersed when predictions are made in early stages of the peri-operative process; the same problem was observed for most other outcome variables, but not for ICU length of stay and cardiac complications. We conjecture that this is a general problem of Bayesian networks, related to statistical inference through chains of stochastic variables. Because each of these variables adds to the uncertainty in the prediction, we observe a regression to the mean when predictions are made through longer chains. A similar phenomenon occurs in forecasting with autoregressive models and Markov models, where long-term predictions tend to move towards the grand mean of the predicted variable [17]. This is a topic that needs further attention before PBNs can be deployed in practice. A potential solution may be found in estimating the dispersion factor using logistic regression [18].

The calibration problem will affect the PBN’s reliability in various tasks, especially those where precise probability estimates are important. An example is the use of probabilistic predictions for risk adjustment [19]. When, however, predictions are merely used to stratify risk (e.g., into low, intermediate, and high risk), calibration is less important than discrimination. Similarly, for the risk factor analysis, one of the use cases that is described in the companion article [1], precise probabilities may be less important as this analysis is aimed at a qualitative result (i.e., identifying relevant variables). Similar considerations hold for the prognostic scenario analysis and what-if scenario analysis: the main, qualitative results will not be affected by poorly calibrated outcome distributions, but the associated numbers should be regarded with caution.
The ProCarSur system currently has a prototype status and has not been evaluated in routine medical care. We have therefore no evidence that the system is suitable for use by clinical staff and that all defined use cases of PBNs are useful during patient care. Clinical evaluation of the usability of the ProCarSur system is therefore an issue for future research, in addition to development and evaluation of such prognostic systems in other clinical domains.

Acknowledgments

The authors thank Bas Groot, Maarten Stenvers, and Maurits Visser, students Medical Informatics, for developing an early prototype of the ProCarSur system, and Winston Tjon Sjoe Sjoe, Antoon Prins, and Floris Wiesman for further developing this system. Furthermore, they would like to thank Lionel Jouffe of BayesiaLab for the support during network development using this software package. Niels Peek receives a grant from the Netherlands Organization of Scientific Research (NWO) under project No. 634.000.020.

References