Artificial Intelligence in Medicine 29 (2003) 5–23

Artificial Intelligence in Medicine

www.elsevier.com/locate/artmed

Integrating classification trees with local logistic regression in Intensive Care prognosis

Ameen Abu-Hanna*, Nicolette de Keizer

Department of Medical Informatics, AMC-University of Amsterdam, Meibergdreef 15,
1105 AZ Amsterdam, The Netherlands

Received 17 January 2002; received in revised form 4 December 2002; accepted 17 March 2003

Abstract

Health care effectiveness and efficiency are under constant scrutiny especially when treatment is quite costly as in the Intensive Care (IC). Currently there are various international quality of care programs for the evaluation of IC. At the heart of such quality of care programs lie prognostic models whose prediction of patient mortality can be used as a norm to which actual mortality is compared. The current generation of prognostic models in IC are statistical parametric models based on logistic regression. Given a description of a patient at admission, these models predict the probability of his or her survival. Typically, this patient description relies on an aggregate variable, called a score, that quantifies the severity of illness of the patient. The use of a parametric model and an aggregate score form adequate means to develop models when data is relatively scarce but it introduces the risk of bias.

This paper motivates and suggests a method for studying and improving the performance behavior of current state-of-the-art IC prognostic models. Our method is based on machine learning and statistical ideas and relies on exploiting information that *underlies* a score variable. In particular, this underlying information is used to construct a classification tree whose nodes denote patient *sub-populations*. For these sub-populations, *local* models, most notably logistic regression ones, are developed using only the total score variable. We compare the performance of this hybrid model to that of a traditional global logistic regression model. We show that the hybrid model not only provides more insight into the data but also has a better performance. We pay special attention to the precision aspect of model performance and argue why precision is more important than discrimination ability.
© 2003 Elsevier Science B.V. All rights reserved.

Keywords: Prognostic models; Classification trees; Logistic regression; Local regression; Intensive Care; Quality of care; Accuracy; Precision

* Corresponding author. Tel.: +31-20-5665959; fax: +31-20-6919840.
E-mail address: a.abu-hanna@amc.uva.nl (A. Abu-Hanna).

doi:10.1016/S0933-3657(03)00047-2

1. Introduction

Health care is expensive and evidence for its effectiveness and efficiency is continuously sought. Many national and international programs have been set up to assess health care quality. For example, our department is responsible for various registries, among which are those for the National Intensive Care Evaluation (NICE), cardiac-interventions, and the renal registry of the European Renal Association. These registries include information about patients and outcomes of care such as mortality and co-morbidity. The registries form vehicles for analytic tools aimed at the evaluation of the quality of care provided.

Prognostic models for outcome prediction form indispensable ingredients among these evaluative tools [1,19]. Model predictions of an event, such as death, are used as a norm to which actual outcomes are compared. Discrepancies between the model's predictions, e.g. probability of death, and observed outcomes, e.g. the proportion of patients who did not survive, are then assessed by experts. The assessment is used in deciding whether actions to improve care delivery should be taken.

In Intensive Care (IC), various prognostic models, such as the SAPS-II [17]—version II of the Simplified Acute Physiology Score—have been developed to estimate the probability of in-hospital mortality. In-hospital mortality includes deaths in the hospital during, or after, stay in an Intensive Care Unit (ICU). Like many other prognostic models in medicine, these are statistical models that can be characterized by their use of a small set of covariates, at the heart of which is a *score* variable reflecting the patient's overall severity of illness, and by their reliance on the probabilistic logistic model.

In machine learning too, various techniques for building prognostic models have been suggested, many of which can be characterized by their use of an extended set of covariates and by their non-parametric and often symbolic nature [1,19]. Classification trees, for example, are extensively used for classification tasks. Other non-parametric, but numeric, techniques have been suggested in exploratory statistics such as local regression. These techniques can be used to visualize smoothed data in order to get insight into the structure of a function. Compared to the current parametric models used in the IC, classification trees provide the advantages of symbolic representation and, together with local regression, can also provide for better fit of the data. They, however, may introduce overfitting of the data if care is not taken. It is hence natural to seek a method to combine the advantages of the various techniques in one hybrid model.

This paper motivates and describes a hybrid method for studying and improving the performance behavior of current logistic regression models. The idea behind the method is the identification of ‘‘interesting’’ patient sub-groups for which specialized local models are then fit. In this paper, the patient sub-populations are devised according to a classification tree that attempts to discriminate between patients that live and those that die. The tree uses variables, such as temperature and heart rate, that *underlie* the aggregate severity of illness *score*. The local models are logistic regression models that use the score variable only. Our assumption is that the attributes underlying the score provide an added value that can be exploited by the tree. This added value is expressed in terms of getting better insight in the data and in terms of improved prognostic performance.

To facilitate comparison with ‘‘traditional models’’, these traditional models are exemplified by a new logistic regression model developed on our NICE data. This model is based on the SAPS-II severity of illness score. The model will be referred to as N-SAPS, where the N stands for NICE.

In measuring prognostic performance of a model in a quality of care program, we are interested in the model's *precision*: how close are the predicted probabilities to the true probabilities? This means that the popular error rate and area under the Receiver Operator Characteristic (ROC) measures are less suitable. This is because they essentially measure how much a model can discriminate between classes without taking precision into consideration. In this paper, we pay special attention to aspects of model evaluation and use measures that are indicative of the precision of a probabilistic model.

The paper is organized as follows. Intensive Care and prognostic models are introduced in Sections 2 and 3. Then in Section 4 global aspects of model evaluation are provided. Section 5 describes the IC dataset and introduces the method of combining classification trees with local models trained on sub-populations. In Section 6 the results of applying this method to the IC data are reported. Section 7 concludes this paper with a discussion, reference to related work, and outlook.

2. Intensive Care and quality assessment

Intensive Care can be defined as *a service for patients with potentially recoverable conditions who can benefit from more detailed observation and invasive treatment than can safely be provided in general wards or high dependency areas* [3]. In The Netherlands there are more than 100 IC Units (ICUs) varying from one room four-bed ICUs to large departments with up to 60 beds in university and teaching hospitals. Approximately 100,000 patients are admitted in Dutch ICUs every year.

In the Intensive Care, monitoring and treatment of organ failure of critically ill patients are not only costly as consequence of the price tags of the advanced equipment used, they also require a large number of skilled personnel to maintain and use these technical facilities 24 h a day, 7 days a week. In addition, differences in the view on the best delivery of care, professional ambitions, budgetary constraints and insurance regulations now have prompted physicians and managers to assess the quality of IC treatment. In this paper, we restrict our quality measure to in-hospital mortality which is a measure of the *effectiveness*, rather than efficiency, of an ICU. The motivation of this choice is the fact that death is a sensitive and objective outcome measure and it has regrettably a relatively high frequency, making it a suitable indicator of the effectiveness of quality of care.

Since it is unethical to evaluate the effectiveness of ICU treatment in a randomized trial, information from (inter)national databases is used as a foundation of objective appraisal of the quality of the care process. This is performed by comparing outcome data such as mortality, morbidity and length of stay, with the *predicted* outcome values [14,24]. The predictions should take into account the characteristics of the patient population admitted to an ICU. This is called case-mix adjustment and is usually quantified by a score of the severity of illness of the patient at his or her admission. Existing IC case-mix adjustment models are mainly concerned with predicting mortality. Prognostic models that make these case-mix adjusted predictions lie hence at the heart of quality assessment. The use of prognostic models is described below and their form and development is dealt with in the next section.

There are a number of different logistic regression models used in quality of care programs such as the NICE program. NICE is a foundation established in 1996 by an initiative of a professional group of intensivists to gain insight and to improve the effectiveness and efficiency of Dutch ICUs. Many different ICUs in The Netherlands are currently participating in NICE. The following is a sketch of the procedure followed by NICE in supporting decisions about quality of care within the ICUs. The procedure highlights the role of prognostic models.

1. Each participating ICU provides its care-related data. For each patient, more than 100 items are collected including patient description and various outcomes such as length of stay and mortality.
2. After data quality validation procedures have taken place, the information from all participants is accumulated and stored in the NICE registry.
3. Based on a large amount of data from all ICUs that have been registered in the past, a prognostic model to predict mortality was constructed as described in the next section.
4. The prognostic model is validated on a large test set that includes data from all participating ICUs. Given a new patient population, a prognostic model may now be considered as predicting the mortality in this specific population that would be expected in an "average" national ICU.
5. The model is used to predict probability outcomes for new patients from each ICU. For *each* ICU the model's predictions are *lumped into probability groups* and compared with the actual *proportion* of mortality in that probability group.
6. Representatives of the participating units meet periodically and discuss the reasons behind discrepancies between their "performance" and the "national average" predictions for their patients.
7. Based on the outcomes of the last step, decisions are made for future improvement of care.

3. Prognostic models

Various methods have been suggested for the representation of prognostic models ranging from quantitative and probabilistic approaches to symbolic and qualitative ones. These models are built either by hand, learned from data, or by a combination of both. In the early approaches to building knowledge-based systems, symbolic prognostic models have been built by hand. For example, consider the Ventilator Manager (VM) system [9]. VM interprets online quantitative data in the Intensive Care Unit to advise physicians on the management of post-surgical patients needing a mechanical ventilator to help them breathe. VM uses, among others, rules to predict the future values of clinical parameters. By comparing them with actual values of the parameters, these predictions are used to evaluate the accuracy of the system.

Instead of the laborious work of building the model by hand, statistical and machine learning approaches try to fit, or learn, a model based on the available data. In medicine, classical statistical approaches for fitting a model are typically parametric and probabilistic in nature. They usually tackle a regression task (the prediction of a numeric value). In contrast, machine learning approaches [21] are typically non-parametric in nature and often tackle a classification task (the prediction of the class to which an instance belongs).

When considering the suitability of a prognostic approach in a quality of care assessment program, one is to note that the notion of a *group* is more central than the notion of an *individual*. In particular, one typically is interested in the comparison of some outcome *summary* among patient groups rather than in the classification of individual patients to a particular class. This emphasizes the importance of probabilistic statements about groups as will be demonstrated in Section 4.

The method described in this paper uses different modeling approaches for prognosis. The central two modeling approaches are based on *logistic regression* and the symbolic non-parametric method of *classification trees*. These approaches are described below.

3.1. Logistic regression models and current IC prognostic models

The major prognostic systems in IC are the Acute Physiology and Chronic Health System (APACHE) [15] and the SAPS [17]. These systems are based on logistic regression.

A logistic regression model is a parametric model that specifies the probability of a dichotomous variable $Y(\{0,1\})$ to have the value 1 given the values of the covariates of the model. $Y = 1$ indicates the occurrence of an event such as death. The logistic model has the following form:

$$p(Y=1|\boldsymbol{x}) = \frac{\mathrm{e}^{g(\boldsymbol{x})}}{1+\mathrm{e}^{g(\boldsymbol{x})}} \tag{1}$$

where $\boldsymbol{x} = (x_1, \ldots, x_m)$ is the covariate vector. For m variables (also called predictors), the *logit function* $g(\boldsymbol{x})$ has the following form:

$$g(\boldsymbol{x}) = \beta_0 + \sum_{i=1}^{m} \beta_i x_i \tag{2}$$

where β_i $(i = 1, \ldots, m)$ denotes the coefficients of the m predictors. Fitting a logistic regression model implies finding estimates of the $\beta_0, \ldots, \beta_m$ that maximize the likelihood of the model (that is, the probability of the data given the model). Note that the conditional probability in Eq. (1) is in effect the conditional expectation $E(Y = 1|\boldsymbol{x})$ because Y is binary.

The monotonicity of logistic regression in each covariate means that it is not a good idea to represent a raw physiological attribute, such as temperature of the patient, directly as a covariate. To illustrate, consider that a similar increase, e.g. from 35 to 37 °C should have an opposite effect on the probability of the adverse event than an increase from, say, 37 to 39 °C. In the first case, the patient's temperature moves from an abnormal to a normal state, whereas in the second case the opposite occurs. Because of this, the APACHE and SAPS use aggregate *scores* for the quantification of the severity of illness. A score is then used as covariate instead of the individual attributes. A higher score corresponds to greater deviation from the healthy status and hence a higher probability of death. Different elements contribute to this total additive score such as physiological variables, e.g. heart rate, and white blood cell count; demographic variables, e.g. age; and covariates concerning earlier chronic history. For example, the total SAPS-II score ranges from 0 to 163 points and is related to a probability of hospital mortality based on the following logit model:

$$g(\text{score}) = -7.7631 + 0.0737\,\text{score} + 0.9971\,\ln(\text{score} + 1) \tag{3}$$

Besides the monotonicity assumption underlying the logistic function it has additivity and linearity assumptions that will be discussed in Section 5.2. It is, however, clear that the functional form of this model is fairly pre-specified and one is only required to find the appropriate β_i parameters that fit the data. When the assumptions underlying logistic regression hold, this approach does not require much data and provides models that are relatively immune to overfitting. Our prior ‘‘knowledge’’ of the parametric form allows one to search faster in a restricted space of possible models and compensates for idiosyncrasies in the data. If the assumptions do not hold, however, one introduces bias (in the statistical sense). This bias manifests itself in (the lack of) the goodness of fit (precision) of the model.

3.2. Classification trees

Classification trees are widely called decision trees [23] in the machine learning literature. We will stick to the term classification trees (CTs) in order to avoid confusion with the term decision trees in Decision Analysis which bears a different meaning. CTs are very popular in machine learning applications and are also used as prognostic models in medicine [2]. Similar techniques for the construction of CTs have been independently developed in statistics [5]. CTs are non-parametric models with very high expressive power. Non-parametric models can in principle better fit the data but they introduce the risk of overfitting the data (see [25]). CTs are attractive because they provide a symbolic representation that lends itself to easy interpretation by humans (that is not to say, however, that they are necessarily intuitive). The representation can also be extended or easily modified when a tree is translated into convenient If–Then rules.

The idea in CT learning is to use variables to partition the dataset into *homogeneous groups* with respect to the same class (e.g. ‘‘survivors’’ and ‘‘non-survivors’’). The CT has a tree structure in which an internal node denotes a variable, the branches of a node denote value (or value ranges) of the corresponding variable, and a leaf denotes a (dominant) class. The CT construction is achieved by recursively partitioning sets beginning with the whole dataset (e.g. of patients). Each partitioning of a set is based on a corresponding value-partitioning of some variable. For example, $\{\leq 45, >45\}$ is a possible value-partitioning of the *age* variable that implies partitioning a given patient set in two (age) subsets accordingly. In each of the recursive iterations, the aim is to find the variable, along with its value-partitioning, that can result in subsets that are maximally homogeneous (pure) in their class value. A popular impurity measure of classes in a given set is the entropy of the set with respect to the classes; for a two-class problem the entropy of a set S is defined as $-p_1 \ln(p_1) - p_2 \ln(p_2)$, where p_i denotes the probability of the ith class in S. The search for variables that maximally discriminate between the classes, by minimizing entropy, aims at finding the minimal set of variables that would lead to a satisfactory separation between the classes. In this paper, the classification tree will be used to identify subsets of patients and will not be used to directly predict outcomes.

4. Model evaluation aspects

Consider two different probabilistic prognostic models, M1 and M2. Suppose that M1 and M2 are given a specific score, say score_0, as the covariate value and are used to return a mortality probability. Suppose that M1 and M2 provide two different estimates, say 0.55 and 0.85, respectively. If a classification of survival is sought for *individual* patients with score_0, both models will predict non-survival for each of them (because the probability assigned for the event—non-survival—is greater than the probability assigned for survival). Assuming there are more non-survivors than survivors among the patients with score_0, these two models are equally effective in discriminating between the two survival states and hence will inflict the same total error rate (the Bayes error rate) in this group.

There is, however, a substantial difference between the probability estimates of the two models which cannot be ignored if one is interested in the *probability* of an event within a *group* of patients. To judge the performance of an ICU, one compares the *estimated probability* of non-survival with the *observed mortality rate* of a specific group. If 70% of patients with score_0 did not survive then according to M1 the ICU is performing very poorly but according to M2 it is performing much better than expected for that group of patients. Although classification models and error rates still form the traditional focus of much work in machine learning, in this work we are interested in a *precise* model instead. A precise model provides honest estimates of the *true probability* of an event rather than merely a discriminating model with the ability to assign the highest probability to the actual event or class.

The concepts of discrimination and precision can be put into perspective by considering the following definitions (based on [10]). For brevity we will only consider the two-class case: 0 and 1. We use the following notations: $f(j|x_i)$ is the (unknown) true probability that *instance* x_i belongs to class j. Note that x_i here is an instance such as a patient, so for example, x_1 and x_2 can correspond to two different patients with possibly the same score values. $\hat{f}(j|x_i)$ denotes the predicted probability that instance x_i belongs to class j. We use c_i to denote the true class of instance x_i.

Accuracy is a measure of the effectiveness of the model to assign an instance to its actual class. An estimate of accuracy error is based on some summary measure of $|c_i - \hat{f}(1|x_i)|$. Classification error rate is a special case of an accuracy error measure obtained by imposing a threshold on $|c_i - \hat{f}(1|x_i)|$ and taking its average. Other measures include the logarithmic score, described below, which also consists of an element of precision.

Precision is a measure of the closeness between the true and estimated probabilities. An estimate of precision error can be based on a summary measure of $|f(j|x_i) - \hat{f}(j|x_i)|$. As $f(j|x_i)$ is unknown, one can obtain an estimate of it from a *test set*. When the response variable is binary this requires some grouping principle to lump instances in the test set and take, e.g. their average. We add a distinction between two flavors of precision based on the grouping principle of instances. In *y-precision*, the *estimated* probability is divided into *probability regions* and compared to the true (observed) proportion of the event in each region. In *x-precision*, one world group instances only according to the *similarity in their "x" values*, regardless of the estimated probabilities.

Evaluation of logistic regression models such as the APACHE-II and SAPS-II models usually rely on the Hosmer–Lemeshow statistics [12] where it is referred to as calibration (see [20] for a more general discussion). In our terminology, these are essentially y-precision measures with *non-overlapping* probability regions. A major disadvantage of the Hosmer–Lemeshow statistics is that they have been shown to be quite sensitive to the cut-off points used to form the estimated probability regions [13]. We will need to select measures indicative of precision without the drawbacks of the Hosmer–Lemeshow statistics.

4.1. Selected performance measures

Measures of errors, often called scoring rules, can be *proper* or *improper*. A proper score is a score (measure of error, not to be confused with the severity of illness score) that receives its *minimal* value only when the model provides the *true* probabilities. This means that a model that is imprecise (for example, by exaggerating the probability of a dominant class) will be penalized.

Error rate is a very popular accuracy measure used in machine learning research. It also forms the basis for the Receiver Operating Characteristic (ROC) curve [11], whose area is a summary measure of error rate taken at all possible thresholds. Error rate, however, is an improper scoring rule. Although it may provide a good idea of how well a model is able to separate between classes, error rate can make the model *look* better, or worse, than it really is from the precision point of view. It is hence inappropriate to use error rate as our performance measure.

In this work, we measure accuracy based on the logarithmic score which is a proper scoring rule [8,10]. In other words, models that overpredict or underpredict mortality are penalized. This means that this measure of accuracy is indicative of precision too. Although we do not get a direct measure of precision, the *comparison* of accuracies of two models on the same test set will amount to comparing their precision in this case (see Section 7 for other direct and indirect ways to compare precision). The logarithmic score LS_i for instance i is given by:

$$\mathrm{LS}_i = -\ln p_i(Y_i = y_i|x_i) \tag{4}$$

where y_i is the true class (observed in the test set). A total penalty on the whole test set is the sum of the logarithmic scores on all N test cases: $\sum_{i=1}^{N} \mathrm{LS}_i$.

We will obtain this cumulative measure on patient groups sharing various "x" characteristics (e.g. admission type and physiological variables) as will be demonstrated below. Hence, our measure exhibits x-precision characteristics and does not require to group patients by thresholding their estimated probabilities. Therefore, our measure does not suffer from the drawbacks of the Hosmer–Lemeshow statistics.

5. Data and method

5.1. The Intensive Care dataset

Data used in this analysis originates from the Dutch NICE registry. This registry includes, among others, all variables needed to calculate the SAPS-II score and includes IC-mortality data as well as in-hospital mortality data. We used NICE data from January 1998 till April 2000 and applied the SAPS-II exclusion criteria: patients younger than 18 years of age; length

Table 1
Characteristics of important attributes in the dataset

Variable name	Description (value)	Mean ± S.D.	Normal range/value	Frequency (%)
syst.min	Minimal systolic blood pressure	92.0 ± 32.4	100–199	
urine.24	Urine production in first 24 h	2.6 ± 2.3	>1	
heartr.min	Minimum heart rate	71.2 ± 23.0	70–119	
bicarb.min	Minimum bicarbonate	22.4 ± 5.2	≥20	
bicarb.max	Maximum bicarbonate	25 ± 4.5	≥20	
gcs.low	Lowest Glasgow Coma Scale	13.8 ± 3.2	15	
wbc.max	Maximum white blood cell count	12.1 ± 11.4	1–19.9	
adm.type	Admission type			
	Medical admission (1)			45.3
	Unscheduled surgical (2)			17.7
	Scheduled surgical (3)			37.0

of ICU stay under 8 h; patients with acute myocardial infarction, or burns, or patients in the postoperative period after coronary artery by-pass surgery. For patients with multiple ICU admissions during one hospital stay, only data from the first admission were used. The data included in this study consisted of 7803 consecutive admissions to ICUs of eight different hospitals. Important attributes are described in Table 1. All values were obtained within the first 24 h of IC admission. The mortality within the ICU is 15.5% and the in-hospital mortality, the dependent (response) variable, amounts for 22.5%.

Fig. 1 shows the design of the experiments. The data are divided into two disjoint sets: a training set to build the tree, called *tree training set* ($n = 2585$), and a model training/test dataset ($n = 5218$) that is used in a 5 × 2 cross-validation (5 × 2 CV) manner to train and test, the global and local, prognostic models. The 5 × 2 CV design means that the dataset is used in five experiments. In each experiment the model train/test dataset is split into two randomly selected subsets S_1 and S_2 of equal size. The experiment has two folds: in the first fold a model is trained on S_1 (the global model will use all cases in $S1$ and the local models, described below, will be trained on the tree-induced groups of $S1$) and tested on S_2. In the second fold of an experiment, a model is trained of S_2 and tested on S_1. In total there are 10 folds. This design is meant to minimize the dependency among training sets, and also among test sets. It also provides a sufficiently large test set to work with, which is important in our case as the training data and the test data are to be split into (patient) sub-groups.

5.2. Hybrid method

In this section, we motivate the use of a method aimed at a better understanding of the IC data and the improvement of models fitted to it. First let us review the assumptions underlying logistic regression and consider the case that there is only one covariate, the (severity of illness) score:

$$p(Y = 1|\text{score}) = \frac{e^{\beta_0+\beta_1\text{score}}}{1 + e^{\beta_0+\beta_1\text{score}}}$$

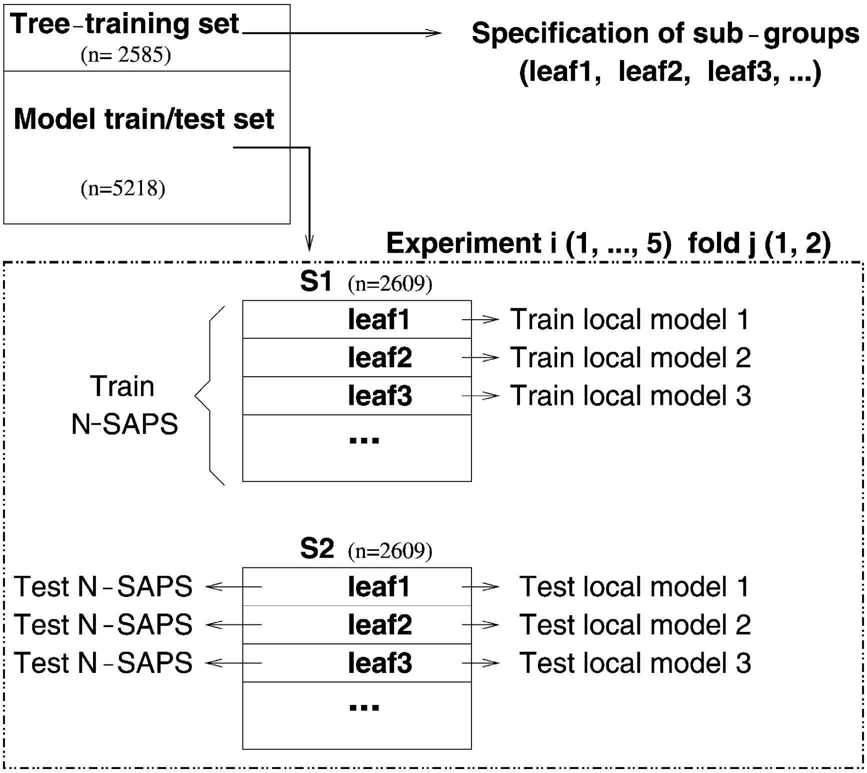

Fig. 1. The design of experiments showing the use of training and test data. The figure shows the set-up in one fold (out of two folds) in each of the five experiments.

The assumptions can be decomposed into independence, distributional, additivity and linearity assumption. In our data the patients are independent. Moreover, the distributional assumption is not problematic as it only implies a binomial distribution of the residuals. Additivity does not pose problems either as we have only one covariate, the score, and there is no need to seek interactions with other covariates. Linearity, however, assumes that $g(\text{score})$ is linear in score (see Eq. (2)). Note, in particular, that a change of u units in score leads to a change of $\beta_1 u$ in $g(\boldsymbol{x})$ independently of the current value of score. This might not be always appropriate. Our approach, presented below, to coping with this assumption is to introduce local logistic regression (LLR) models where linearity is only assumed within some identified *sub-groups* of patients.

A perhaps more interesting issue in current models that rely heavily on a score variable is whether there is valuable information within the underlying attributes that are disregarded once the aggregate score is computed. For example, a patient with unscheduled surgery (8 sub-points) and normal heart rate (0 sub-points) will score as a patient with a medical admission (6 sub-points) with a heart rate between 40 and 69 (2 sub-points), assuming for simplicity that they score the same for the other attributes. As far as the total score is concerned these differences are masked. This means in effect that changes in the total score, and hence in $g(\boldsymbol{x})$, are insensitive to patient characteristics: a unit change in the score will affect patients with different admission types, ages, etc. in the same way. Our approach to investigate the role of the underlying attributes is to create the sub-groups based on patients that share some of their underlying attribute values. The total score, however, will be the sole covariate in the models fitted once the groups are identified. In summary, we seek sub-groups to mitigate the linearity assumption, and we request that the sub-groups rely on the variables underlying the score variable. The question that remains is how to create these patient sub-groups.

The idea behind partitioning the patients in groups is based on the following observations. A logistic model basically "distributes" the total number of events as probability estimates on the whole population. Suppose that the outcome variable is 0 or 1 if the patient lives or dies, respectively, and that there are n patients in some *training* set. Then fitting the logistic model results in $\sum_{i=1}^{n} y_i = \sum_{i=1}^{n} \hat{f}(Y_i = 1|x_i)$ on the training set. We believe a challenge for these models resides at patient sub-populations with very different proportions of the event. We do not want to construct these groups based on the outcomes but rather based on the characteristics of the patients. A classification tree, which minimizes entropy with respect to the event—by discriminating between the two classes—is hence suitable for creating such groups. The tree will use the "raw" values underlying attributes of the severity of illness score. Its leaves correspond to the patient sub-populations of interest. The way the method is to be used can be summarized as follows: first classify each patient into a node using only the "raw" variable values underlying the score. No scores are used in this step. In the second step, use *only* the total score of the patient to get a probability estimate. In this step, N-SAPS would use the fitted model on the whole training set to provide this estimate. In the case of the local models, the mortality estimate for each patient will be performed by the local model corresponding to the patient's node.

It is to be noted that the entropy-based tree is not the only way to construct the sub-groups. In fact even if one insists on a tree structure, the entropy-based approach to building the tree is not specifically geared toward optimizing the performance of the local models. One could, for example, formulate a node splitting criterion that maximizes the performance of the local models on the sub-nodes. Another approach is to use an unsupervised algorithm to cluster the patients into sub-groups without using information about mortality. However, employing a tree structure, along with using entropy as the splitting criterion, yields the following features. First, it provides a symbolic representation that clinicians can interpret. Second, because the tree tries to discriminate between survivors and non-survivors, the groups can be interpreted in terms of *risk*, which is a natural way for clinicians to conceptualize the groups. Third, because of the use of entropy, the probability distributions of mortality in sub-nodes are likely to be very different, which means they form a challenging test for a global model—the common practice in IC prognosis. Note that the tree is able to discriminate between different patients with the same score as they may end up in different nodes. This is illustrated in Section 6.2 and explains why these patients may receive different mortality probabilities by the local models.

There are various scenarios for experimenting with these ideas whose results are reported in Section 6. In the scenarios below we used the experimental design sketched in Fig. 1. In this design the tree training set is used to grow a classification tree. This is done only once and the tree training data is then immediately discarded. Next, in each of the 10 folds (of the five experiments), half of the randomly selected train/test dataset is used as a model training set. The other half is used as a model test set in that fold. Note that the classification tree implies a partitioning of the data in the model train/test sets.

5.2.1. Qualitative insight into the mortality function in sub-groups

In this scenario, the test set (of the fold at hand) is partitioned according to the tree. Then a smoothing method for visualizing the probability of mortality as function of the total SAPS-II score is applied to each sub-group, as described shortly. This provides insight into the behavior of the probability function. This insight can also be used to qualitatively inspect how well a prognostic model fits the test data.

5.2.2. Global logistic regression, N-SAPS (NICE-SAPS)

In this scenario one obtains the N-SAPS model trained on the whole model training set (for the same fold). Then one simply examines its performance on each of the subsets induced by the classification tree on the test set (for that fold).

5.2.3. Local logistic regression

In this scenario one develops a logistic model, again using only the total SAPS-II score covariate, but now on each tree-induced *subset* of the training set of the corresponding fold. One then inspects the performance of these models on their respective test subsets. In this scenario the added value of training on sub-groups is assessed by comparing the performance of the local models with that of the global model on the test subsets.

For implementing these scenarios we used the statistical package S-Plus (S-Plus 6 for Linux, Insightful Corp.).

6. Results

This section describes the results of using our method for studying the probability function within interesting IC patient sub-populations. It shows where weaknesses of the N-SAPS model are encountered. This also suggests how to build an improved hybrid prognostic model for IC outcome.

In supporting understanding the results we will resort to two additional methods. The first is Cleveland's *Lowess* [7]. Lowess is a non-parametric "local weighted polynomial regression smoother" that is used to visualize the structure of data in a scatter plot. We will use Lowess to get a visual idea of the mortality function in patient sub-populations. Given an x-value (score in our case), Lowess calculates a y-value based on smoothing the y values (0's and 1's in our case) of its neighbors using a local parametric model. Neighbors are weighted according to their distance from the given x-value: closer points receive higher weight. One could interpret the smoothed value of each point as the *probability* of mortality of patients with that score.

The second additional method is the related K-Nearest Neighbors method (K-NN). We use the average value of the mortality attribute of the five nearest neighbors of a patient to get an estimate of the patient's probability of death. Due to its popularity in machine learning and statistics we use the 5-NN estimates solely to provide a *frame of reference* in order to put the results of our local and global logistic models in perspective. The 5-NN estimates themselves are far worse than the logistic models.

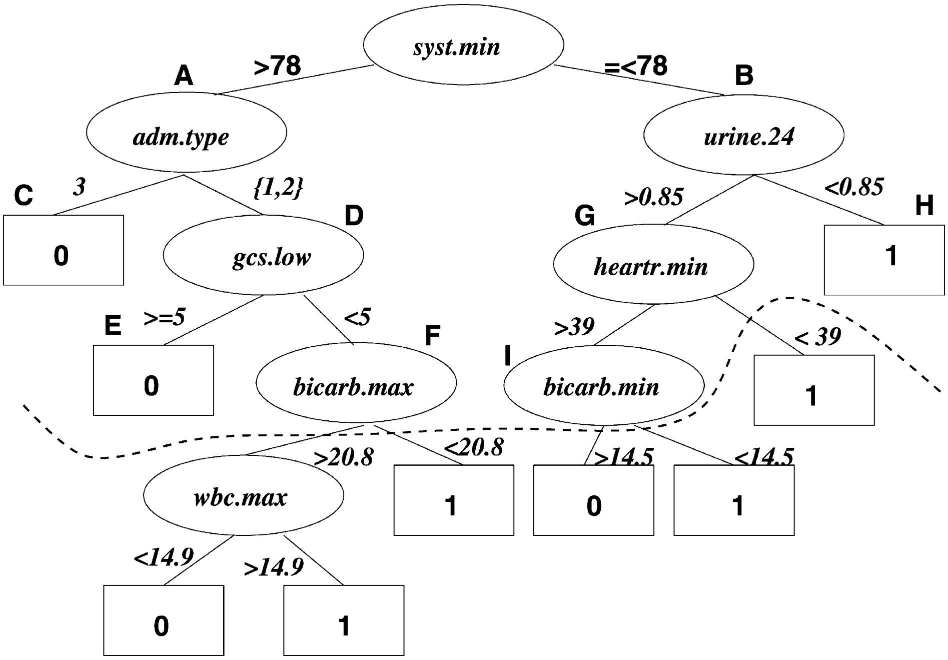

Fig. 2. The IC classification tree based on physiological attributes and admission type (see Table 1 for attribute descriptions). The leaves denote the majority class. The nodes above the dashed line correspond to groups with sufficient number of patients and will be formally inspected.

6.1. The classification tree

We developed a binary classification tree on the tree training set (which is completely disjoint from the model train/test sets used to build and test the prognostic models). The restriction to a binary tree is mainly aimed at combating fragmentation. Moreover, all attributes are either continuous or ordinal (admission type can be clinically viewed this way in the sense that higher values of admission type indicate more serious conditions). Information gain based on entropy was used as the criterion for the selection of attributes in the tree. The tree is shown in Fig. 2. Note that patients with the same score could end up in different nodes in this tree as most scores can be obtained by different combinations of, e.g. physiological variables. Each node above the dashed line in the figure corresponds to a sufficiently large group of patients—one with an expected number of at least 80 in each fold of the test sets. The nodes C, E, F, H, and I will be considered leaves and are inspected below.

6.2. Insight

In order to view the probability functions in the different nodes on the training set we smooth the raw mortality data by Lowess. The Lowess smoothing made it clear that there are differences between the functions at different nodes. The functional form at each node turned out to be qualitatively consistent in random samples of the dataset, as long as there

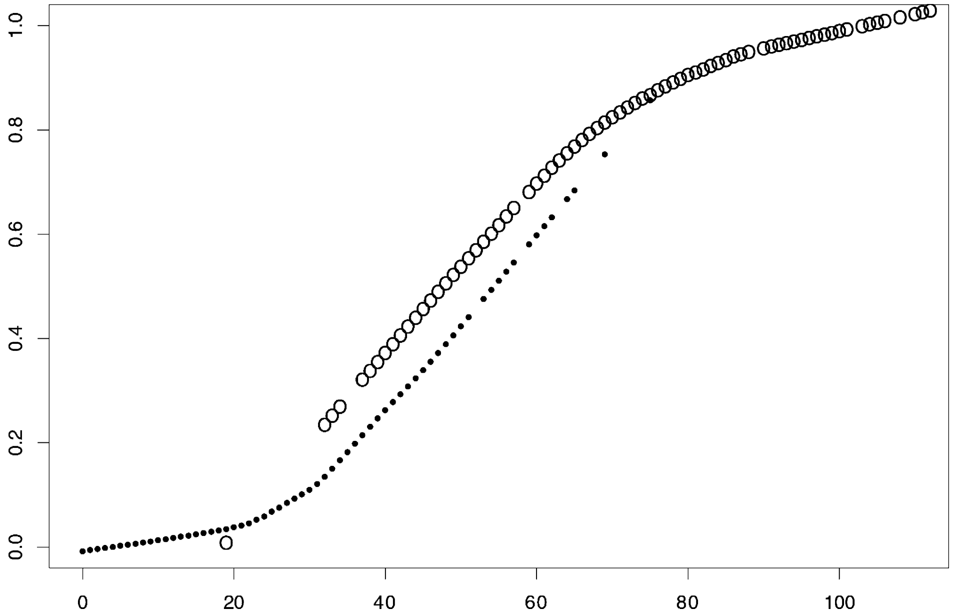

Fig. 3. Smoothed mortality as function of SAPS-II score at node C (small circles) and node H (big circles).

were sufficient instances in each node. As an illustration, consider Fig. 3 corresponding to nodes C and H. These are characteristic figures for a random fold. As can be seen, the functions have a great deal of overlap in their SAPS-II scores but are different (and in other nodes there were qualitative differences in the shape of the function than from the one shown here). One may argue that the two nodes refer to groups corresponding to different dominant outcomes (0 for C and 1 for H) and perhaps it is unfair to compare them. However, instances with *similar* scores are assigned into these groups based only on their attribute values, and the mortality probability is calculated based only on the total score. This means that the tree is exploiting information which has not been explicitly used in the score and hence is masked from N-SAPS. This proves the added value of the tree. One way to use this insight is simply to induce the tree partition within the IC data at hand and inspect the conformance of some ICU to this (qualitative) behavior. One may also try to detect drift by obtaining data from the same ICU but taken at different times.

6.3. Evaluation tool

6.3.1. The development of N-SAPS

In our 5 × 2 CV scheme there are 10 folds in total. An N-SAPS model is developed on the whole training set for each fold. An N-SAPS model uses only the SAPS-II score as covariate. For simplicity, we did not use the logarithm of the score that appears in Eq. (3) as the second covariate. The following is an example of an N-SAPS model taken from the first fold:

$$g(\text{score}) = -4.2548 + 0.0737\,\text{score}$$

6.3.2. The development of local models

For each of the 10 folds, the training set of that fold is partitioned according to the classification tree. LLR models are then fit to the nodes C, E, F, H, and I above the dashes line in Fig. 2. To formally inspect the model performances, we obtain our performance measures for the local prognostic models on the test sets of that fold for the same nodes. The performance of N-SAPS is also inspected for that fold on these nodes.

The results of the experiments are shown in Table 2 where the columns correspond to the nodes considered. The second row provides the average number of instances in the 10 folds, along with the standard deviation. The following two rows show the performance results in terms of the logarithmic score for the global logistic model (N-SAPS) and the LLR models. To provide solely as a *frame of reference* we also provide the logarithmic score performance of the 5-NN model. The estimates of this model are obtained by seeking the five patient scores closest to the patient at hand and calculating the proportion of their mortality. When several patients have the same scores, all of them are included. The 5-NN model has been outperformed in every node in any fold by each of N-SAPS and LLR.

The results in Table 2 show that for each of the nodes C, E, H, and I, the performance of the local models has been clearly better: for the nodes C, E, and H the local models have outperformed the N-SAPS model in 8 out of the 10 folds. The local models have also outperformed N-SAPS in 7 out of the 10 folds for the node I. Only for the node F the local and global models tie: in five of the folds N-SAPS performed better while in the other five folds the local models outperformed N-SAPS. Note that F has the least number of patients which may explain the difficulty of local models to come up with better performance.

When looking at *each fold*, the performance of the local models on all the nodes C, E, F, H, and I for that fold, a clearer view emerges. In 8 out of the 10 folds the local models (combined for that fold) beat the N-SAPS model. In two folds the models tie. If we assign 0.5 a point for a tie we conclude that the local models win in 9 out of 10 folds. Assuming independence between folds, this amounts to a difference in performance with a statistical significance of 0.019 and hence $P < 0.05$. In theory the independence assumption is of course not completely valid due to overlap among the training sets and among the test sets. However, the 2 × 5 CV design, and the relatively big size of the database should maintain the robustness of the P value.

Table 2
Results of the 10 folds including: average number of instances in the test set; average of the logarithmic score; and number of wins and losses between N-SAPS and the local logistic regression (LLR) model

Node	C	E	F	H	I	Combined
av. #instances	843 (14)	1095 (29)	87 (7)	129 (4)	416 (22)	2572 (5)
av. score LLR	178 (15)	487 (16)	53 (6)	52 (5)	235 (8)	1006 (24)
av. score N-SAPS	182 (12)	489 (18)	53 (6)	53 (3)	236 (8)	1012 (25)
av. score 5-NN	195 (19)	529 (25)	61 (10)	66 (8)	275 (11)	1126 (22)
#winsLLR vs. N-SAPS	8	8	5	8	7	8 wins, 2 ties
#winsN-SAPS vs. LLR	2	2	5	2	3	0 wins, 2 ties

Numbers in brackets indicate the standard deviation from the mean.

When inspecting the results where N-SAPS could be improved, by considering the average error for a patient in a group, the following sub-populations stand out:

- [Node C] Patients with a normal to high systolic blood pressure who are admitted after scheduled surgery; these patients are relatively healthy.
- [Node H] Patients with low blood pressure and low urine production reflecting possible organ failures such as heart and renal failure. These patients are seriously ill.

This result is in line with other studies in which the goodness of fit of current (global) logistic regression models is often found lacking at the extremes: patients which are relatively healthy and patients which are seriously ill. The difference is, however, that in our analysis one can characterize these groups in terms of their attribute values. In current approaches, one establishes that the model does not calibrate for very low and very high scores but the different patient populations cannot be further discerned.

To get a better appreciation of how the models behave, consider Fig. 4. It illustrates the model's behavior at node C on an arbitrary fold. N-SAPS's predictions on the test set are shown to be consistently far from the "true" probabilities—those obtained by Lowess smoothing on the observed outcomes of the test set. About 95% of the patients in this relatively healthy group have a score of less than 35. A further examination has shown that the N-SAPS model overpredicts the mortality of this group by 26 deaths compared to 8 overpredictions by the LLR model. To complete the illustration, also predictions based on a non-parametric local model are shown (using the triangle symbol). These are obtained by Lowess—"learned" from the training set—with the test set scores. We note that although

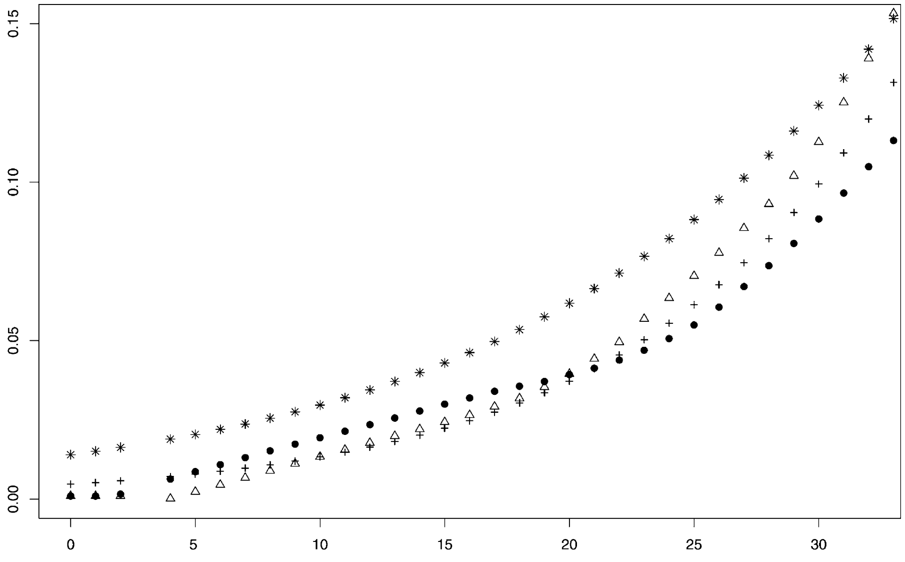

Fig. 4. Mortality probability as function of SAPS-II score: (•) "true" probability based on Lowess smoothing on test set; (∗) N-SAPS predictions; (+) local logistic regression predictions; (△) predictions based on Lowess training (see text).

the non-parametric estimates are slightly worse than the parametric estimates on this node, they still perform better than N-SAPS.

7. Discussion and conclusions

The results reported in the last section are encouraging. The precision of the hybrid method is statistically significantly better than N-SAPS—the representative model of current IC prognosis. Because we have only used the logarithmic score to compare precision between models, one may speculate whether the results would change if other measures would be used. We resorted to three other approaches. In the first approach, we used the Brier score which is also a proper scoring rule. In the second approach, we used a statistical test of precision based on a logarithmic summary statistic obtained from the test set and from the estimated probabilities (see [10]). In the third approach, we used a direct way to measure precision: we considered Lowess smoothing results on a test set as the ‘‘true’’ probabilities and compared these with the model predictions. In all approaches the results have been essentially the same: the hybrid method was superior to N-SAPS.

It was also interesting to inspect how non-parametric local regression models would compare to the logistic regression models. As could be seen from Table 2, the 5-NN model was clearly inferior to the parametric models. However, when we conducted experiments in which we used Lowess on the training set as basis for predictions on the test set, the results were mixed. In general, the local parametric models have been superior to this Lowess-model. In most cases, however, this Lowess-model has been superior to N-SAPS. This probably indicates that the Lowess-model is benefiting from grouping the patients but that the number of patients in the leaves is still too small in order to beat the local logistic regression models.

From this work, one can postulate that using the hybrid method of a classification tree with local prognostic parametric models provides better insight into the data and better prognostic performance. One may conclude that the attributes underlying severity of illness scores can indeed contribute to better models. It is interesting to note that in our search for the the granularity of covariates we started from the severity of illness score, which turned out to be too coarse, while in other approaches, such as [4], one seeks a granularity by searching for aggregations starting from (too) low level features.

The idea of comparing logistic regression with classification trees and also its combination with other models are in themselves not new. Moreover, work concerning the general theme of the interplay between machine learning and statistics [22] has been reported in various publications. In [18], a comparison between classification trees and logistic regression on a medical database is reported. In [26], the correspondence between classification trees and the logit model are explained. In [16], Naive Bayes Classifiers are fit on some of the tree nodes to boost classification. More recently, a related idea to ours has been described in [6] for learning ‘‘Treed Models’’ based on Bayesian methods where, among others, logistic regression models have been proposed. The contribution of our work lies in providing a motivated synthesis of modeling and evaluation concepts and their application in state-of-the art prognosis in Intensive Care. The idea is to tap information from the score variable and hence it is applicable to many other medical domains. The method is tailored to the specific constraints of decision support in quality of care programs with special emphasis on precision. As argued in this paper, in quality of care programs, precision plays a more important role than discrimination. This has influenced our hybrid design and its evaluation. This emphasis on precision would, for example, not allow for models such as Naive Bayes Classifiers which are known to typically have good and robust classification error but are often quite imprecise.

We have listed the advantages of our approach of using an entropy-based tree to devise the patient sub-groups. However, in further work it is interesting to look at other ways to form sub-groups including approaches that are unsupervised, or tree-based supervised approaches that are geared toward maximizing the performance of the local models instead of solely minimizing entropy. Another topic for further study is to empirically investigate when to stop growing the tree and how to opportunistically combine different model types for providing the best prognosis including local non-parametric models.

From the management of care point of view it is hoped that the hybrid method will eventually enhance decision support concerning quality of care. Further work includes putting our method to the test in the decision support process concerning the improvement of quality of care. An important sequel to the work presented in this paper, from both the methodological as well as the clinical point of view, is the inclusion of other outcome measures than mortality, such as length of stay in the ICU. Also the inclusion of organ failure scores registered daily for each patient is likely to improve case-mix adjustment and also allows for comparison of *temporal patterns* of these scores between different ICUs. These additional data are more complex to handle but they do provide more sensitive measures of quality of care.

One might conclude that our results provide proof of concept that there is room for a variety of modeling concepts for studying and enhancing the current generation of prognostic models. We feel that the inspection of interesting patient sub-populations is an important enrichment to the traditional logistic regression models.

Acknowledgements

We would like to thank the board of the National Intensive Care Evaluation (NICE) foundation for its support and feedback. The board consists of: G.J. Scheffer, R.J. Bosman, E. de Jonge, J.C.A. Joore, H.H.M. Korsten, J.G. van der Hoeven, P.H.J. van der Voort. Furthermore, we are grateful to all NICE participants for collecting the data. Niels Peek, Linda Peelen, and Barry Nannings provided comments on earlier drafts of this manuscript.

References

[1] Abu-Hanna A, Lucas PJF. Prognostic models in medicine AI and statistical approaches. In: Abu-Hanna A, Lucas PJF, editors. Special issue of Methods Inf Med, vol. 40. Stuttgart: Schattauer; 2001. p. 1–5.

[2] Aitchison TC, Sirel JM, Watt DC, MacKie RM. Prognostic trees to aid prognosis in patients with cutaneous malignant melanoma. BMJ 1995;311:1536–9.

[3] Bennett D, Bion J. ABC of intensive care. Organisation of intensive care. BMJ 1999;318:1468–70.

[4] Bohanec M, Zupan B, Rajkovic V. Applications of qualitative multi-attribute decision models in health care. Int J Med Inf 2000;58–59:191–205.

[5] Breiman L, Friedman JH, Olsen RA, Stone CJ. Classification and regression trees. Belmont: Wadsworth; 1984.

[6] Chipman H, George E, McCulloch R. Bayesian treed models. Machine Learn 2002;48:299–320.

[7] Cleveland WS. Robust locally weighted regression and smoothing scatterplots. J Am Stat Assoc 1979;74:829–36.

[8] Cowell RG, Dawid APh, Lauritzen SL, Spiegelhalter DJ. Probabilistic networks and expert systems. New York: Springer; 1999.

[9] Fagan LM, Shortliffe EH, Buchanan BG. Computer-based medical decision making: from MYCIN to VM. In: Clancey WJ, Shortliffe EH, editors. Readings in medical artificial intelligence: the first decade. Massachusetts: Addison-Wesley; 1984. p. 241–55.

[10] Hand DJ. Construction and assessment of classification rules. Chichester: Wiley; 1997. p. 105.

[11] Hanley JA. Receiver operating characteristic (ROC) methodology: the state of the art. Crit Rev Diagn Imaging 1989;29:307–35.

[12] Hosmer DW, Lemeshow S. Applied logistic regression. New York: Wiley; 1989.

[13] Hosmer DW, Hosmer T, Le Cessie S, Lemeshow S. A comparison of goodness-of-fit tests for the logistic regression model. Stat Med 1997;16:965–80.

[14] de Keizer NF. An infrastructure for quality assessment in intensive care; prognostic models and terminological systems. Ph.D. Thesis. Amsterdam: University of Amsterdam; 2000.

[15] Knaus W, Draper E, Wagner D, Zimmerman J. APACHE II: a severity of disease classification system. Crit Care Med 1985;13:818–29.

[16] Kohavi R. Scaling up the accuracy of naive-bayes classifiers: a decision-tree hybrid. In: Proceedings of the Second International Conference on Knowledge Discovery and Data Mining. Monlo Park: AAAI Press; 1996. p. 202–7.

[17] Le Gall J, Lemeshow S, Saulnier F. A new simplified acute physiology score (SAPS-II) based on a European/North American multicenter study. JAMA 1993;270:2957–63.

[18] Long WJ. A comparison of logistic regression to decision-tree induction in a medical domain. Comput Biol Res (1993) 74–97.

[19] Lucas PJF, Abu-Hanna A. Prognostic methods in medicine. In: Lucas PJF, Abu-Hanna A, editors. Artif Intell Med (1999) 15(2):105–9 [special issue].

[20] Miller ME, Hui SL. Validation techniques for logistic regression models. Stat Med 1991;10:1213–26.

[21] Mitchell T. Machine learning. New York: McGraw-Hill; 1997.

[22] Nakhaeizadeh G, Taylor CC, editors. Machine learning and statistics, the interface. New York: Wiley; 1997.

[23] Quinlan JR. C4.5: programs for machine learning. San Mateo (CA): Morgan Kaufman; 1993.

[24] Rowan K, Kerr J, Major E, McPherson K, Short A, Vessey M. Intensive care society's APACHE II study in Britain and Ireland-II. BMJ 1993;307:977–81.

[25] Schwarzer G, Vach W, Schumacher M. On misuses of artificial neural networks for prognostic and diagnostic classification in oncology. Stat Med 2000;19:541–61.

[26] White AP, Liu WZ. Statistical properties of tree-based approaches to classification. In: Nakhaeizadeh G, Taylor CC, editors. Machine learning and statistics, the interface. New York: Wiley; 1997.