Evolving classification of intensive care patients from event data

Mark Last[*a], Olga Tosas[b], Tiziano Gallo Cassarino[c], Zisis Kozlakidis[c], and Jonathan Edgeworth[b]

Address: [a]Department of Information Systems Engineering, Ben-Gurion University of the Negev, Marcus Family Campus, Rager St., Beer-Sheva 84105, Israel, [b]Department of Infectious Diseases, Guy's and St. Thomas' NHS foundation Trust, Westminster Bridge Road, London SE1 7EH, United Kingdom and [c]The Farr Institute of Health Informatics Research, University College London, 222 Euston Road, London NW1 2DA, United Kingdom

Email: Mark Last[*] - mlast@bgu.ac.il; Olga Tosas - Olga.Tosas@gstt.nhs.uk; Tiziano Gallo Cassarino - t.cassarino@ucl.ac.uk; Zisis Kozlakidis - z.kozlakidis@ucl.ac.uk; Jonathan Edgeworth - jonathan.edgeworth@gstt.nhs.uk

* Corresponding author

Abstract

Objective: This work aims at predicting the patient discharge outcome on each hospitalization day by introducing a new paradigm - evolving classification of event data streams. Most classification algorithms implicitly assume the values of all predictive features to be available at the time of making the prediction. This assumption does not necessarily hold in the evolving classification setting (such as intensive care patient monitoring), where we may be interested in classifying the monitored entities as early as possible, based on the attributes initially available to the classifier, and then keep refining our classification model at each time step (e.g., on daily basis) with the arrival of additional attributes.

Materials and Methods: An oblivious read-once decision-tree algorithm, called information network (IN), is extended to deal with evolving classification. The new algorithm, named incremental information network (IIN), restricts the order of selected features by the temporal order of feature arrival. The IIN algorithm is compared to six other evolving classification approaches on an 8-year dataset of adult patients admitted to two intensive care units (ICUs) in the United Kingdom.

Results: Retrospective study of 3,452 episodes of adult patients (≥ 16 years of age) admitted to the ICUs of Guy's and St. Thomas' hospitals in London between 2002 and 2009. Random partition (66:34) into a development (training) set n = 2,287 and validation set n = 1,165. Episode-related time steps: Day 0 – time of ICU admission, Day x – end of the x-th day at ICU. The most accurate decision-tree models, based on the area under curve (AUC): Day 0: IN (AUC = 0.652), Day 1: IIN (AUC = 0.660), Day 2: J48 decision-tree algorithm (AUC = 0.678), Days 3-7: regenerative IN (AUC = 0.717 - 0.772). Logistic regression AUC: 0.582 (Day 0) - 0.827 (Day 7).

Conclusions: Our experimental results have not identified a single optimal approach for evolving classification of ICU episodes. On Days 0 and 1, the IIN algorithm has produced the simplest and the most accurate models, which incorporate the temporal order of feature arrival. However, starting with Day 2, regenerative approaches have reached better performance in terms of predictive accuracy.

Keywords: Evolving Classification; Decision Trees; Logistic Regression; Event Data Streams; Intensive Care.

1 Introduction

Over the recent years, the nature, the scale and the speed of data collected within healthcare has changed dramatically, creating new challenges and opportunities. For example, we may be interested to utilize data mining techniques for estimating the probabilities of various discharge outcomes on each day of a given hospital episode. This can be considered as an *evolving classification* problem, where each patient is repeatedly assigned a probability distribution over the optional classes, such as A (discharged alive) vs. D (discharged dead) as more clinical data becomes available. The evolving classification problem considered in this paper is different from the well-known problem of *incremental learning from evolving data streams* [1] [2], where the model should be adapted to changing system dynamics in response to new data samples that are continuously arriving over time.

In this paper, we introduce a new paradigm for evolving classification of event data streams. We extend an oblivious read-once decision-tree algorithm, called information network (IN), to deal with evolving classification. The new algorithm, named incremental information network (IIN), restricts the order of selected features by the temporal order of feature arrival. The IIN algorithm is evaluated on the outcome prediction task in an 8-year dataset of adult patients admitted to two Intensive Care Units (ICUs) in the United Kingdom.

The rest of this paper is organized as follows. Section 2 covers related work on evolving classification algorithms and risk prediction in intensive care units. Section 3 describes the analyzed dataset and the evaluated classification algorithms. The results of the data analysis are presented in Section 4. Section 5 concludes the paper with some insights and directions for future research.

2 Related work

Most classification and regression algorithms, such as logistic regression [3], decision trees [4] [5], and support vector machines [6], are not designed for the "evolving classification" task as they consider all predictive features at the same time while ignoring the potentially temporal nature of various features and feature sets. Millan-Giraldo *et al*. [7] deal with a streaming data scenario, where one or several attributes of incoming instances arrive only after some delay. They suggest the following three straightforward strategies for an early classification of streaming data with delayed attributes: *Do-nothing* (ignore the values of delayed attributes when they become available), *Put-and-reclassify* (re-classify an instance after all attributes become available), and *Wait-and-classify* (classify an instance *only* after all attributes become available). According to the experimental evaluation of [7], *Wait-and-classify* proves to provide the most accurate results out of the above three strategies, especially when the delayed attributes are the most relevant ones. In case of hospital episodes, *Do-nothing* means classifying a given patient at a single time point (e.g., 24 hours after admission) and then ignoring all data arriving afterwards, *Put-and-reclassify* can be interpreted as repeatedly classifying a patient episode on arrival of new attributes, and *Wait-and-classify* implies that patients are classified only on the discharge day when all episode attributes become known. Of course, predicting the episode outcome on the discharge day is nearly useless in the clinical setting.

The delayed attributes scenario of [7] is related to the novel paradigm of entity stream mining introduced by Krempl et al. in [8]. This paradigm assumes monitoring a set of entities, such as hospital patients, in the course of their lifetime (e.g., during a given hospital episode). At various time points, each entity is linked to structured or unstructured records ("instances") generated by

entity-related events such as medical tests. While many different learning tasks may be defined over such an entity stream, we focus here on the evolving classification task, where an entity classification is required at multiple time steps based on partial sequences of entity-related events.

Stratification of patients into risk groups is important for comparing quality-of-care across different hospitals and units, evaluating the results of clinical trials, and other purposes [9]. Back in 1985, Knaus et al. [10] presented APACHE II, a point score system for estimating the risk of ICU death outcome from 12 physiologic measurements, age, and previous health status. Based on its worst value measured within 24 hours after ICU admission, each physiological parameter is assigned a severity weight on a scale of 0 to 4. The sum of all points is called APACHE II score. Its maximum possible value is 71 though in practice it usually does not exceed 55. The predictive capability of APACHE II was evaluated on 5,815 ICU admissions in 13 US hospitals during 1982. The data collection process was actively controlled by the authors of [10]. A statistically significant increase in the death rate was shown for each 5-point increase in the APACHE score. The predictive capability of APACHE II was evaluated using logistic regression analysis with the outcome as the dependent variable. The area under receiver operating characteristic (ROC) curve (AUC) reported in [10] for the logistic regression model based on APACHE II score is 0.863. Since the paper [10] does not specify any cross-validation procedure, the reported predictive performance may be based on the developmental (training) data only and thus it may be higher than the true (validation) accuracy. Though the paper [10] emphasizes the importance of the early patient classification at the time of ICU admission (rather than after 24 hours), no alternative models for such early prediction are proposed.

Contrary to [10], Lemeshow et al. [11] propose two mortality probability models (MPM), named MPM0 and MPM24, for use at ICU admission and 24 hours after admission, respectively. This is a relatively large study, collected data on 19,124 adult ICU patients at 139 hospitals in 12 European and North-American countries. In case of multiple ICU admissions, only data from the first ICU episode was used. The quality of the collected data was monitored by the physician coordinators at each hospital. The collected records were randomly partitioned into developmental (training) and validation samples with the training / validation ratio of 0.65 : 0.35. The variables to be used in each multiple logistic regression model were chosen based on statistical tests and clinical plausibility. The data for inducing the MPM0 model included patients who stayed in the ICU for less than 24 hours and the validation area under the ROC curve of that model reached 0.824. However, such short ICU stays were excluded from the training set of the MPM24 model, which has shown a slightly higher AUC of 0.836. Out of 13 variables included in the MPM24 model, 5 variables were available at admission and 8 additional variables were assessed at the 24-hour mark.

Both the authors of [12] and [9] indicate the need of accurate risk prediction models for patients who stay in ICU beyond 72 hours. Hence, in [12], the MPM24 model has been adapted to 48-hour and 72-hour prediction by adjusting the constant coefficient of the logistic regression equation induced for the 24-hour model while keeping the coefficients of all independent variables unchanged. This approach has resulted in a decrease in the validation AUC of the MPM48 model to 0.796 (vs. 0.836 of MPM24) and a further decrease to 0.752 for the MPM72 model. In their Discussion section, the authors of [12] try to find an explanation of this counter-intuitive result, since generally we would expect a later classification model, based on more accumulated information about the patient, to be more accurate.

Trujillano *et al.* [13] calculate the probability of hospital mortality with three decision-tree classification algorithms: CART [4], CHAID [14], and C4.5 [5]. All evaluated models are aimed at severity estimation for patients within the first 24 hours of their admission only. The authors of [13] indicate that the main benefits of decision trees include the high interpretability of the resulting decision rules along with the relative homogeneity of patient groups associated with each terminal node ("leaf") of the tree. A retrospective dataset of 2,864 patients was randomly partitioned in a 70:30 ratio, to form the development and the validation sets. On the validation set, all decision-tree models have reached in [13] a reasonable AUC level of 0.75-0.76, which was very close to the APACHE II AUC (0.77), but lower than the AUC of logistic regression (0.81).

Portela et al. [15] present the INTCare, a Pervasive Intelligent Decision Support System, which supports intensive care medical activities. The system was used for predicting Organ Failure (Cardiovascular, Coagulation, Respiratory, Hepatic, and Renal) and the Outcome (live or death) of 129 patients in a Portuguese ICU, based on the first five days of their stay. The attributes were collected from bedside monitors, lab results, drugs system, and hospital records. The predictive accuracy of an ensemble of classification models varied across targets between 43% and 83% (64% for predicting the patient outcome).

The overall conclusion is that outcome prediction models for ICU patients are mainly focused on risk assessment after 24 hours in the intensive care and, in the case of MPM_0, at the time of ICU admission. The reported predictive performance of these models is quite reasonable (AUC above 0.80). There were also several attempts to predict the outcome at later 24-hour intervals in the ICU (mainly MPM_{24} and MPM_{72}), but those models had a limited success and a general methodology of refining patient classification models during the entire ICU stay is still missing.

3 Materials and methods

3.1 Database

The dataset used in this study included 3,452 episodes of adult patients (≥ 16 years of age) admitted to the ICU of Guy's and St. Thomas' NHS Foundation Trust in London between 2002 and 2009. This is a 30-bed mixed (medical and surgical) ICU located on the St Thomas' Hospital site. The episode-related data records were extracted from the Trust clinical information systems. The dataset used in [16] for studying antibiotic guideline adherence was extracted from the same information systems and collected over the same years but it was based on different inclusion criteria. The details of the ethics committee approval can be found in [16].

Only episodes lasting between 3 days and 21 days were included in the current study. To improve the completeness of the data, we have excluded episodes with a missing diagnosis and episodes with no recorded APACHE scores within the two days from the ICU admission. Repeated ICU episodes of the same patients (about 18% of all episodes) were treated as unrelated entities disregarding the time between consecutive episodes. The ICU mortality rate for the included episodes was 20.8%. The average ICU stay duration of included episodes was 8.9 days with a median of 7 days. The average patient age at the time of admission was 61.4 years with a median of 64 years. About 63% of admitted patients were males.

The episode-related time steps were defined as follows: Day 0 – time of ICU admission, Day 1 – end of the first day at ICU, Day2 – end of the second day at ICU, etc. Due to this definition and the inclusion criteria of the dataset, the resulting number of periods and their respective data tables was 22. The outcomes (class labels) of all episodes became available on the last day of each ICU

episode, i.e., between Day 4 and Day 21, and they were recorded in a separate table *C*. Each episode was labeled as A (discharged alive) or D (discharged dead). If the outcome field was empty in an episode record of the ICU clinical information system, we have filled it by looking up the date of death for the specific patient. All patients without a recorded date of death or deceased more than 24 hours after the ICU discharge were assumed to be discharged alive. No assistance from the human experts was needed for obtaining the true labels.

The features recorded for each episode at the time of ICU admission (Day 0) are shown in Table 1. This information was extracted directly from episode records in ICU clinical information systems. A fictional *Episode Record ID* (entity ID) was created for each unique combination of *Patient ID* and *Episode Number* in the original records.

Table 1 Day 0 attributes

Ser. No.	**Attribute Name**	**Attribute Range**
1	Episode Record ID	Integer
2	Age	16 and higher
3	Sex	M/F
4	Specialty	68 various codes and labels
5	Diagnosis Label	48 various labels
6	Episode Number	1-6

Each one of the Day 1 – Day 21 data tables included several physiological parameters along with the APACHE scores calculated on that day. These records were extracted from the daily measurement records stored for each episode in a clinical information system and assigned to the appropriate data table by calculating the day number for each measurement record as the difference

between the measurement date and the ICU admission date. The *Episode Record ID* of each measurement record was identified by the unique combination of *Patient ID* and *Episode Number*. The complete list of 27 physiological parameters repeatedly measured or calculated on each of Days 1-21 is shown in Table 2. All these attributes take numeric values only. Multiple measurement records with the same combination of *Patient ID, Episode Number*, and *Date* (which were mostly duplicates) were merged into one record by taking the maximum value of each measurement. The descriptive statistics of physiological parameters is shown in the Appendix (Table 10).

Table 2 Day 1 - 21 attributes

Ser. No	Name	Ser. No	Name
1	Precipitating Factor Code (Day 1 only)	15	PaO2
2	APACHE Score	16	PaCO2
3	Acute Physio Score	17	pH
4	GCS	18	Haematocrit
5	Age Points	19	WBC
6	Chronic Health Points	20	Platelets
7	Temp	21	Sodium
8	SBP	22	Potassium

9	HR	23	Creatinine
10	Bicarb	24	Urea
11	INR	25	SGOT
12	Resp	26	Albumin
13	Urine	27	Bilirubin
14	Glucose		

In Day 2 data table, we have also included several features representing the results of microbiology samples taken during the episode, since these results usually become available on Day 2 or later. First, we have defined five binary features indicating whether at least one sample was taken from one of the following five specimen groups: skin breach, sterile site, urine, blood, and respiratory tract. For this purpose, a separate table was built relating each specimen name in the microbiology sample records to one of the above five groups. For each specimen group, we have defined a binary feature named "*Culture Found in [Specimen Group Name]*", which indicates if a positive culture was found in at least one of the samples taken from that specimen. A microbiology sample result was interpreted as a "positive culture" if the bacteria quantity was indicated as +/-, +, ++, or +++. Each microbiology sample record was related to an episode record using the *Patient*

Identifier, the *Sample Date*, and the episode dates (*Admission Date* and *Discharge Date*). For each positive microbiology sample record, we had also an indication of the bacteria type (gram-positive vs. gram-negative). Thus, we have defined another binary feature for each specimen named "*Gram-negative Found in [Specimen Group Name]*", which indicates if gram-negative bacteria (usually characterized by a stronger antibiotic resistance) were found in at least one of the relevant episode samples (taken from the corresponding specimen).

3.2 The Evolving classification paradigm

We present here a new paradigm for *evolving classification* of event data streams based on a static training dataset of labeled entity records. This is different from the data streaming scenario of [1] [2] [7], where training instances arrive over time. Each entity in the training dataset is linked to a set of entity-related events recorded at various time steps of an entity lifetime. Given the training dataset, an evolving classification algorithm should produce a set of classification models, which, ideally, satisfy the following requirements:

1. Providing an up-to-date classification model for each time step (e.g., episode day *d*) based on the predictive features available up to that step. This is similar to the *Put-and-reclassify* strategy proposed in [7].

2. Refining an existing classification model upon arrival of a new set of attributes. Such an incremental approach should improve the transparency of the classification process by minimizing the amount of changes made to the previous model rather than inducing a completely

new model at each time step. The changes may involve modification (e.g., expansion) of the model structure or just updates of the model parameters.

3. It should be an *anytime classification algorithm* [17]. This means that models induced from additional attributes that become available over time should have a non-decreasing classification performance.

4. The induced models should be *interpretable*, i.e. provide an easy to understand explanation of their prediction. Decision trees are an example of easy to interpret classification models [13].

5. The models induced from the attributes that become available after the first few time steps should be nearly as accurate as the models induced from all entity attributes (a property known as *earliness* [18]).

6. Each model prediction should be accompanied by an *uncertainty estimate* [18], which allows the user to decide whether to wait for additional data before classifying a specific episode.

In this study, we explore the evolving classification properties of three algorithms: logistic regression [19], C4.5 decision tree [5], and an oblivious read-once decision-tree algorithm (IN) [20]. In the ICU setting, where nearly the same measurements are taken every day (see Table 2 above), the *Put-and-reclassify* strategy [7] can be implemented in the following three ways: update the model built for the previous day (d -1), build a new model from all attributes available up to day d, and build a new model from the attributes that become available only on day d. We call these three approaches *incremental*, *regenerative all*, and *regenerative last*, respectively. In addition to implementing the regenerative approach with all three algorithms, we have developed an

incremental version of the IN algorithm that minimizes the difference between two successive classification models. The evaluated algorithms are briefly described in the following sections.

3.3 Logistic regression

The logistic regression (LR) models [19] estimate the posterior probabilities of target classes as linear functions of one or several numeric features. The models are called simple or multiple logistic regression when the number of predictive features is equal to one or greater than one, respectively. They are widely used in biostatistical applications. The models are usually fit by maximum likelihood. The logistic regression models use two basic assumptions: the independence of observations (in the ICU setting, this means that the outcome of one patient is independent of the outcomes of other patients) and the linearity of the relationship between the natural log of the odds ratio and the predictive features (which means that the log of the outcome odds ratio increases / decreases linearly as a function of each predictor variable). Like in multiple linear regression, variable selection techniques include forward selection, backward elimination, and stepwise. The best variable is considered the one, which maximizes the difference in likelihood between a regression equation with or without it. In this study, we have built LR models with the W-Logistic operator available on RapidMiner 5.3.15 [21] without changing any of its default settings. W-Logistic is a Weka (Waikato Environment for Knowledge Analysis) [22] class for building a multinomial logistic regression model with a ridge estimator.

3.4 C4.5 decision-tree algorithm

C4.5 is a decision tree induction algorithm introduced by Quinlan in [5] as an extension of the ID3 algorithm [23]. Both algorithms perform variable selection at the node level, which means that the sequences of attributes tested along each tree path may be quite different, except for the first attribute, which appears in the root node. C4.5 chooses the best splits of its internal nodes based on an entropy-based criterion called "information gain ratio", which aims at maximizing the "purity" of the terminal ("leaf") nodes. It can handle both continuous and discrete attributes. Missing attribute values in C4.5 training data are simply skipped in the algorithm calculations. To avoid overfitting, a constructed tree is post-pruned using a reduced error pruning technique. Due to its computational efficiency and relatively high accuracy, C4.5 is considered one of the most popular classification algorithms in machine learning. We have built C4.5 models with the W-J48 operator available on RapidMiner 5.3.15 [21]. W-J48 is a Weka [22] implementation of C4.5, written in Java.

3.5 IN algorithm

IN [20] is an oblivious read-once decision-tree algorithm. The name *oblivious* indicates the fact that, unlike most decision-tree models, all nodes at a given layer of an information network are labeled by the same predictive feature, whereas *read-once* means that a feature is tested at most once along any model path. Consequently, the order of predictive features tested by the IN model is fixed along each path, which is not necessarily true in the models induced by the C4.5 algorithm. The best feature selected for splitting the nodes of a given IN layer should maximize the statistically significant conditional mutual information with the target (classification) variable. The formulas for calculating the conditional mutual information and testing its statistical significance can be found in

[20]. As shown in [20], the IN construction algorithm tends to produce considerably smaller and thus more interpretable models than other decision-tree models of similar accuracy.

3.6 IIN algorithm

3.6.1 The algorithm outline

To deal with evolving classification of entity data streams, we have developed an incremental version of the IN algorithm, named IIN, where the order of selected features is restricted by the temporal order of feature arrival. The IIN pseudocode is presented in Algorithm 1. IIN obtains as input the maximum time step *Max_L*, which is of interest for entity classification purposes (e.g., the first week of ICU stay). It also obtains the training data tables $Data_d$ corresponding to various time steps, starting with $d = 0$, which represents the initial entity state (e.g., the state of a patient at her/his admission), the list E_d of new features arriving at time step *d*, and the table *C* of episode class labels, which were recorded at the patient discharge from ICU. Each record in every data table $Data_d$ should have a single matching record in table *C* uniquely identified by an Episode Record ID, whereas each record in table *C* may have at most one related record in each data table $Data_d$. Thus, we assume that multiple attributes recorded at the same time step may be represented by multiple features (columns) in the same data table but not by multiple records. The algorithm has two parameters: *Max_F* - the maximum number of predictive features selected at each time step (default = number of all new available features $|E_d|$) and *sign* – the minimum significance level for splitting a network node (default = 0.1%, based on [20]). The default settings of these parameters were not changed in our study. The IIN software is available upon request from the first author.

Algorithm 1: Incremental information network induction algorithm

Input: Maximum lifetime of an entity *Max_L*, a set of training data tables $Data_d$ representing entity-related events (features) E_d recorded in period d (d = 0,…, *Max_L*), a list C of class labels for each training entity, maximum number of predictive features per period *Max_F*, minimum significance level *sign* for splitting a network node.

Output: a set of classification models IN_d for predicting the entity class at the end of period d.

1: **For** d = 0 to *Max_L* **do**
2: Read the data table $Data_d$ and the list of features E_d
3: **While** the maximum number of period layers *Max_F* is not exceeded **do**
4: Find the best candidate input attribute *Best_Attr* $\in E_d$ maximizing the statistically significant conditional mutual information.
5: If the maximum conditional mutual information is greater than zero, make *Best_Attr* an input attribute and define a new layer of network nodes for period d; else **end while**.
6: End while
7: Return the set of selected attributes and the network structure IN_d for the end of period d
8: End for

A partial example of an incremental information network constructed for the first two time steps of ICU episodes is shown in Figure **1**. The first layer corresponds to Day 0, where the selected attribute is Diagnosis, whereas in the second layer, corresponding to Day 1, the APACHE score was selected by the algorithm as the best attribute. The five nodes of the first layer (Nodes 1 – 5) represent the five nominal values of the *Diagnosis* attribute. In contrast, the APACHE score is a continuous attribute, which has been discretized by the algorithm into four intervals creating four respective nodes (Nodes 6 – 9) in the second layer. Each node in this network should have two connections representing the two possible outcomes (classes): *Alive* and *Dead*. For the sake of

brevity, we have replaced these connections in Figure **1** with the probabilities of the death outcome shown at each node. One may see from Figure **1** that the least risky group of patients in the first layer (Day 0 / Diagnosis) are the patients with viral infection whose death probability is only 0.146. On the other hand, the most risky group in the second layer (Day 1 / APACHE) are the patients diagnosed with colon cancer at the admission and having APACHE score of 26 and higher at the end of the first day in ICU. Their probability of death outcome is as high as 0.417.

3.6.2 The algorithm calculations

Step 4 of Algorithm 1 requires calculating the conditional mutual information of each candidate input feature, which became available at the time step *d*, and then testing this conditional mutual information for statistical significance. The Algorithm 1 calculations corresponding to each candidate feature are identical to the calculations performed in each layer by the original IN construction algorithm [20], which ignores the temporal order of candidate features. The main calculation formulas are shown below while an interested reader is referred to [20] for complete details.

For each candidate input (predictive) attribute A_i in a layer n, the algorithm calculates the conditional mutual information of A_i and the target (classification) attribute T given n-1 input attributes $X_1, \ldots, X_{n-1}$ by the following formula [20]:

$$MI(T; A_i / X_1,..., X_{n-1}) = \sum_{z \in L_{n-1}} MI(T; A_i / z) \qquad (1)$$

Where $MI(T; A_i / z)$ is the conditional mutual information of a candidate input attribute A_i with the target attribute T given a terminal node z in the layer $n-1$ (denoted by L_{n-1}). Like in any decision

tree, each terminal (leaf) node *z* of the *k*-th layer of an information network represents a specific conjunction of values of *k* predictive attributes associated with *k* hidden layers, respectively.

For nominal predictive attributes, the conditional mutual information of a candidate input (predictive) attribute A_i and the target (classification) attribute *T* given a node *z* is calculated by the following formula [20]:

$$MI(T; A_i / z) = \sum_{t=o}^{M_T - 1} \sum_{j=0}^{M_i - 1} P(C_t; V_{ij}; z) * \log \frac{P(V_{ij}^t / z)}{P(V_{ij} / z) * P(C_t / z)} \qquad (2)$$

Where

M_T / M_i: number of distinct values ("classes") of the target attribute *T* / candidate input attribute *i*, respectively.

$P(V_{ij}/z)$: an estimated conditional probability of a value *j* of the candidate input attribute *i* given the node *z*.

$P(V_{ij}^t/z)$: an estimated conditional probability of a value *j* of the candidate input attribute *i* and a value ("class") *t* of the target attribute *T* given the node *z*.

$P(C_t/z)$: an estimated conditional probability of a value ("class") *t* of the target attribute *T* given the node *z*.

$P(C_t;V_{ij};z)$: an estimated joint probability of a value ("class") *t* of the target attribute *T*, a value *j* of the candidate input attribute *i*, and the node *z* out of all dataset records. At the root node, this probability is identical to $P(V_{ij}^t/z)$.

The statistical significance of the estimated conditional mutual information between a candidate input attribute A_i and the target attribute T given a node z is evaluated by using the following likelihood-ratio statistic [20]:

$$G^2(T; A_i / z) = 2 * (\ln 2) * E^* * MI(T; A_i / z) \quad (3)$$

Where E^* is the total number of training cases in the dataset. The null hypothesis is that the actual conditional mutual information is zero. That hypothesis is rejected if the G^2 statistic is significant at the pre-specified confidence level. Based on the empirical results with real-world datasets [20], the default significance level, leading to the most compact and accurate models, is set to 0.1%, though its *p*-value can be increased if larger models involving more predictive features are needed.

The conditional entropy of the target (classification) attribute can only be calculated with respect to input attributes taking a finite number of values. For continuous predictive attributes, the algorithm performs discretization "on-the-fly" by recursively finding a binary partition of an input attribute that minimizes the conditional entropy. The conditional mutual information of partitioning an interval S of a candidate input attribute at the threshold Th and the target attribute T given a node z is calculated by the following formula [20]:

$$MI(Th; T / S, z) = \sum_{t=o}^{M_T - 1} \sum_{y=1}^{2} P(S_y; C_t; z) * \log \frac{P(S_y; C_t / S, z)}{P(S_y / S, z) * P(C_t / S, z)} \quad (4)$$

Where

$P(S_y / S, z)$: an estimated conditional probability of a subinterval S_y, given the partitioned interval S and the node z. The number of subintervals in each partitioned interval is two.

P (C_t/*S*, *z)*: an estimated conditional probability of a value ("class") *t* of the target attribute *T* given the interval *S* and the node *z*.

P (S_y;C_t/ *S*, *z)*: an estimated joint probability of a value C_t of the target attribute *T* and a subinterval S_y given the interval *S* and the node *z*.

P (S_y; C_t; *z):* an estimated joint probability of a value C_t of the target attribute *T*, a subinterval S_y, and the node *z*.

The main steps of the recursive discretization procedure are described in [20].

At the end of each time step *d*, we have an evolving classification model, where each terminal node represents an estimated probability distribution of all possible outcomes (class labels). Thus, the network in Figure 1 has five terminal nodes (Nodes 1-5) at the end of Day 0 and four terminal nodes (Nodes 6-9) at the end of Day 1. The more different is the posteriori class distribution at Node *z* from the a priori distribution in the entire dataset, the more informative that node is. In [20], we calculate the informativeness weight of a terminal node by the following formula:

$$w_z^t = \sum_{t=0}^{M_T-1} P(C_t; z) \cdot \log \frac{P(C_t / z)}{P(C_t)} = P(z) \cdot \sum_{j=0}^{M_T-1} P(C_t / z) \cdot \log \frac{P(C_t / z)}{P(C_t)} \qquad (5)$$

Where *P*(C_t; *z)* is an estimated joint probability of the target value C_t and the node *z*; *P* (C_t / *z)* is an estimated conditional (*a posteriori*) probability of the target value C_t given the node *z*; *P*(C_t*)* is an estimated unconditional (*a priori*) probability of the target value C_t; and *P(z)* is the probability of a node *z*. As indicated in [20], the above weight represents both the simplicity and goodness-of-fit (*cross entropy*) of a given terminal node. As shown in [20], the sum of informativeness weights

across all terminal nodes in a given network is equal to the estimated mutual information between the set of input attributes and the target attribute.

4 Results and discussions

4.1 Demographic and clinical factors

In the first part of our study, we have explored the effect of demographic and clinical factors (shown in Tables 1 and 2 above) on the episode outcome. We have induced and evaluated outcome prediction models for Days 0 – 7 only, since about 50% of included patients are discharged from the ICU within that time. Since our goal is probability estimation rather than exact classification of a given episode, we have used the Area under ROC curve (AUC) as our main performance criterion, similar to the previous studies of ICU mortality prediction [10-12,23,24]. The original dataset has been randomly partitioned at the 66:34 ratio into a development (training) set with 2,287 episodes and a validation set with 1,165 episodes. The mortality rate was 21.2% in the training set and 20.0% in the testing set. The number of training and testing episodes in each retrospective data table is shown in Table 3.

Table 3 Number of training and testing episodes

Day	Training Set	Testing Set
0	2287	1165
1	1972	994
2	2287	1165
3	2222	1144
4	2182	1119
5	1923	973
6	1606	816
7	1339	680

We have also reduced to six the number of labels for each nominal feature such as the diagnosis (five most common labels + “Other” for all remaining labels). A higher number of labels has resulted in lower validation AUC values of Day 0 models. The resulting six diagnosis labels are shown in Table 4.

Table 4 Diagnosis labels

Diagnosis	Number of Episodes
Cancer of colon	579
Cancer of kidney and renal pelvis	163
Cancer of rectum and anus	287
Cancer of uterus	233
Viral infection	513
Other	1677
Total	3452

We have induced outcome prediction models for the eight time steps (Day 0 – Day 7) using the following algorithms and evolving classification approaches:

- *IIN*: the incremental version of the IN algorithm, where the order of selected features matches the temporal order of their arrival.
- *Regenerative IN (all)*: the non-incremental version of the IN algorithm, which treats all available attributes as candidate predictive features, ignoring the order of their arrival.
- *Regenerative IN (last)*: the non-incremental version of the IN algorithm, which treats as candidate predictive features only the attributes, which became available in the last time step.
- *Regenerative J48 (all)*: the J48 (C4.5) algorithm, which induces a decision tree from all available attributes. The order of features tested along each tree path does not necessarily match the order of their arrival. To maximize the C4.5 classification performance in terms of AUC, the *M* parameter (minimum number of instances per leaf) was changed from the default value of 2 to 10.
- *Regenerative J48 (last)*: the J48 (C4.5) algorithm, which induces a decision tree only from attributes, which became available in the last time step. The value of $M = 20$ has provided the best AUC performance with this approach.
- *Regenerative LR (all)*: the LR algorithm, which uses all available numeric attributes as predictive features. Nominal attributes are discarded by LR models.
- *Regenerative LR (last)*: the LR algorithm, which uses as predictive features only the numeric attributes, which became available in the last time step.

The AUC results of the seven approaches listed above are shown in Figure 2. At the time of ICU admission (Day 0), the IN algorithm produces the most accurate results with AUC = 0.652. When applied to the same data, J48 does not produce a classification tree at all, resulting in the

AUC = 0.5. The LR model based on the two numeric attributes available on Day 0 (Age and Episode Number) is less accurate than the IN model, reaching AUC of 0.582 only. Starting with Day 1, the Regenerative LR models based on all available attributes become the most accurate ones. Only on Day 7, this model is outperformed by the LR model based on the latest attributes only. An increase in the accuracy of evolving Regenerative LR models is statistically significant between Day 0 and Day 1, Day 1 and Day 2, as well as between Day 3 and Day 4. The AUC changes on all other days are not statistically significant. The significance tests were performed with Delong's method for comparing ROC curves built for the same individuals [24] using MedCalc for Windows, version 15.10.0 (MedCalc Software, Ostend, Belgium).

Out of the more interpretable, decision-tree models, the incremental IN has reached the highest AUC of 0.660 on Day 1, the first day when physiological measurements, including APACHE scores, become available. After Day 1, the incremental IN is being increasingly outperformed by other decision-tree approaches, including regenerative IN, which usually provides slightly higher AUC values than J48 models.

As shown in Table 5, the IIN algorithm has created network layers for four time steps only (Days 0, 1, 2, and 3). No features were found statistically significant for Days 4 and higher. At the time of admission (Day 0), *Diagnosis* and *Age* were selected as significant predictive features for identifying the mortality risk of each patient. The *APACHE Score*, which is considered the state-of-the-art severity measure for ICU patients, was selected on Day 1, along with *PaO2* (the partial pressure of oxygen in arterial blood). On the two other days (2 and 3), two additional physiologic parameters were found significant: *Temperature* on Day 2 and *Bicarb* (the bicarbonate of the blood plasma) on Day 3. The differences between the AUC values of IIN models induced on various days

were not found statistically significant using Delong's method for comparing ROC curves built for the same individuals [24].

Table 5 IIN construction results

Day	Layer	Attribute Name
0	0	Diagnosis Label
0	1	Age
1	2	APACHE Score
1	3	PaO2
2	4	Temp
3	5	Bicarb

Similar to other decision-tree models, an information network can be represented as a set of probability estimation rules, which incorporate both numeric and categorical features and can be easily interpreted by physicians. In case of predicting ICU outcome, these rules would identify high-risk and low-risk patient groups. Thus, Table 6 shows the prediction rules induced by the IIN algorithm for Day 0. These rules clearly identify the high and the low risk groups of ICU patients at the time of their admission. The patients with cancer of rectum and anus are under the highest death risk of 0.349, whereas the risk of patients with viral infection is nearly two times lower (0.180). Rule 5 is the most informative rule in Day 0 model, since its outcome distribution is most different from the outcome distribution in the entire dataset.

Table 6. Predictive rules - Day 0 (A – alive, D – deceased)

Rule No.		Prob (A)	Prob (D)
0	If Diagnosis Label is Cancer of colon	0.702	0.298
1	If Diagnosis Label is Cancer of kidney and renal pelvis	0.762	0.238
2	If Diagnosis Label is Cancer of rectum and anus	0.652	0.349
3	If Diagnosis Label is Cancer of uterus	0.835	0.165
4	If Diagnosis Label is Viral infection	0.820	0.180
5	If Diagnosis Label is Other and Age is greater than or equal to 16 and less than 65	0.880	0.120
6	If Diagnosis Label is Other and Age is equal to 65 and higher	0.777	0.224

Table 7 shows the new rules that were induced on Day 1 from the two additional features that were selected by the algorithm: *APACHE Score* and *PaO2*. Rule 11 represents the episodes with the highest death risk (1.000). These are patients with other diagnoses than the five most common types, age of 65 and higher, *APACHE Score* between 20 and 32, and *PaO2* between 81.75 and 84.75. Rule 5 represents episodes with the lowest death risk of 0.047, which is about 4 times lower than the mortality rate of the entire dataset. These are relatively young patients (aged 16-65) with the lowest APACHE scores (below 20).

Table 7 Additional predictive rules - Day 1 (A – alive, D – deceased)

Rule No.		Prob (A)	Prob (D)
5	If Diagnosis Label is Other and Age is greater than or equal to 16 and less than 65 and APACHE Score is greater than or equal to 1 and less	0.953	0.047

Rule No.		Prob (A)	Prob (D)
	than 20		
6	If Diagnosis Label is Other and Age is greater than or equal to 16 and less than 65 and APACHE Score is greater than or equal to 20 and less than 32	0.676	0.324
7	If Diagnosis Label is Other and Age is greater than or equal to 16 and less than 65 and APACHE Score is equal to 32 and higher	0.667	0.333
8	If Diagnosis Label is Other and Age is equal to 65 and higher and APACHE Score is greater than or equal to 1 and less than 20	0.832	0.168
9	If Diagnosis Label is Other and Age is equal to 65 and higher and APACHE Score is equal to 32 and higher	0.182	0.818
10	If Diagnosis Label is Other and Age is equal to 65 and higher and APACHE Score is greater than or equal to 20 and less than 32 and PaO2 is greater than or equal to 0 and less than 81.75	0.712	0.288
11	If Diagnosis Label is Other and Age is equal to 65 and higher and APACHE Score is greater than or equal to 20 and less than 32 and PaO2 is greater than or equal to 81.75 and less than 84.75	0.000	1.000
12	If Diagnosis Label is Other and Age is equal to 65 and higher and APACHE Score is greater than or equal to 20 and less than 32 and PaO2 is equal to 84.75 and higher	0.893	0.107

4.2 Microbiology factors

In the second part of our study, we have explored the effect of microbiology results on the episode severity estimates. Specifically, we have focused on 2,471 episodes, where at least one

microbiology sample was taken from the blood specimen, since the presence of bacteria culture in blood, especially gram-negative bacteria, is strongly associated with severe sepsis and has a significant attributable mortality. While keeping all candidate predictive features for Days 0 and 1, we have defined two new features for Day 2: *Culture Found in Blood* and *Gram-negative Found in Blood*. These features have replaced APACHE scores and all other physiologic measurements taken on Day 2. The second feature (*Gram-negative Found in Blood*) was found statistically significant by the IIN algorithm only when the number of layers per period was restricted to one (*Max_F* = 1). This feature appeared in the two rules of the Day 2 model, which are shown in Table 8. Both rules represent patients with other diagnoses than the five most common types and low APACHE scores (between 1 and 13). However, the death outcome probabilities estimated by these two rules differ by the order of magnitude. If no gram-negative bacteria is found in blood, the death risk of these patients is as low as 0.0278, whereas finding a gram-negative bacteria culture in blood increases their death risk up to 0.2632. A complete set of Day 2 predictive rules for this dataset is shown in the Appendix (Table 11).

Table 8 Predictive rules - Day 2 (blood samples only) (A – alive, D – deceased)

Rule No.		Prob (A)	Prob (D)
17	If Diagnosis Label is Other and APACHE Score is greater than or equal to 1 and less than 13 and blood gram-negative is 0	0.9722	0.0278
18	If Diagnosis Label is Other and APACHE Score is greater than or equal to 1 and less than 13 and blood gram-negative is 1	0.7368	0.2632

4.3 Discussion

In the previous section, we have explored seven different options for evolving classification of ICU episodes. The relationship of the induced models to the requirements of an evolving classification algorithm, defined by us in the Methods section, is briefly discussed below.

Providing an up-to-date classification model. Each one of the evaluated approaches provides an up-to-date classification model for every episode day *d* based on some or all of the predictive features available up to that day. This allows us to apply the *Put-and-reclassify* strategy [7] to incoming ICU episodes.

Refining an existing classification model. Regenerative approaches build a new model from scratch for each time unit rather than refining an existing classification model with a new set of attributes. For example, Table 9 shows the set of input attributes selected by the IN algorithm for each episode day using the *Regenerative IN (all)* approach. Though the most recent APACHE score is always the first attribute in the model, the list of subsequently selected attributes is subject to significant variations. To minimize the amount of changes made to the previous model, one can use IIN, an incremental version of the IN algorithm, but as shown in Figure 2, the classification performance of IIN models becomes significantly inferior to the regenerative approaches starting with Day 3.

Table 9 Attributes selected by the regenerative IN (all) approach

Day	Selected Features
0	Diagnosis Label, Age
1	Day1_APACHE Score, Day1_Urine, Day1_Platelets, Day1_Bilirubin
2	Day2_APACHE Score, Day2_Creatinine, Day2_SBP, Day2_INR, Day1_Urine, Day2_PaO2, Day1_Potassium
3	Day3_APACHE, Day3_Platelets, Day3_Urine, Day2_Temp, Day2_APACHE Score, Day2_Urine, Day1_PaO2, Day2_WBC
4	Day4_APACHE, Day4_Urine, Day4_Creatinine, Day3_Glucose, Day3_HR
5	Day5_APACHE, Day5_Urine, Day5_Platelets, Age, Day2_Haematocrit
6	Day6_APACHE Score, Day6_Platelets, Day6_Urine, Day6_Bicarb
7	Day7_APACHE Score, Day7_Platelets, Day5_AcutePhysioScore, Day3_Creatinine, Day6_SBP

Anytime classification. As shown in Figure 2, all regenerative models tend to have a non-decreasing classification performance as a function of ICU stay duration. In most cases of a slight decrease in the AUC value (e.g., in the *Regenerative LR (all)* model between Day 5 and Day 6), the decrease was not found statistically significant. However, the performance of the incremental IN algorithm experiences a steady decrease after Day 3, since no new attributes are added to the model by the algorithm (see Table 5).

Models interpretability. As indicated by [13], the interpretative advantage of decision trees over logistic regression models is only obtained with simple trees. Figure 3 shows the size of the induced decision-tree models in terms of the total number of tree nodes. The incremental IN

algorithm usually provides the smallest models of around 20 nodes, but their relative accuracy deteriorates after Day 2 (see Figure 2). Considering the models accuracy, the *Regenerative IN (last)* and the *Regenerative J48 (last)* approaches appear to provide the best trade-off between accuracy and interpretability starting with Day 4.

Models earliness. It is noteworthy that the predictive accuracy of the IN model for Day 0 (AUC = 0.652) is only slightly lower than the accuracy of the incremental IN model induced for Day 1 (24 hours after admission) and this difference is not statistically significant. Moreover, none of the decision-tree models induced from the first 24 hours data outperforms the IN model based on the admission data only! These results raise questions about the predictive value of Day 1 APACHE scores given diagnosis and age of each admitted patient. After Day 1, the incremental IN model accuracy does not increase any more, though on Day 2, there is still no other decision-tree model having a significantly higher AUC value. The difference between AUC on Day 3 and Day 7 for all decision-tree and logistic regression models does not exceed 10%, which indicates the earliness of Day 3 models.

Uncertainty estimate. Each leaf node of a decision-tree model is associated with a probability distribution of the outcome. In case of a binary classification problem, such as ICU discharge outcome, the leaves where one of the possible classes has a probability of close to one (or close to zero) may be considered more certain than the leaves where the outcome probabilities are close to their distribution in the entire population, i.e. at the root node. Similar information is available with logistic regression models, which calculate the class probabilities for each validation record.

Our experimental results have not identified a single optimal approach for evolving classification of ICU episodes. On Day 0 and Day 1, the incremental IN algorithm has produced the simplest and the most accurate models, which incorporate the temporal order of feature arrival. However, starting with Day 2, other approaches have reached better performance, especially in terms of predictive accuracy. Specifically, LR models produce the highest AUC values, but they are harder to interpret than decision-tree models of 20-30 nodes, which are obtained with IN and C4.5 algorithms. It appears to be more beneficial to induce decision-tree models from the last arriving attributes only, since these models tend to be more compact while not necessarily less accurate. The challenge of inducing both accurate and interpretable episode classification models, which are continuously refined upon arrival of new attributes, requires further research with multiple classification algorithms and ICU datasets.

Competing interests

The authors declare that they have no competing interests.

5 Conclusions

In this paper, we have presented a new paradigm for evolving classification of event data streams by a set of daily classification models, which can utilize all available features, are refined or re-trained upon arrival of new features, have anytime properties, are interpretable (in case of decision trees), become accurate as early as possible, and are accompanied by uncertainty estimates. Seven alternative approaches were shown to meet most evolving classification requirements though no "ideal" approach to evolving classification has been found. This implies that more methods for the evolving classification scenario should be developed and evaluated. The effects of various bacterial

cultures from different sites, along with their antimicrobial susceptibility data, may also be explored in the context of the timing whereby appropriate antimicrobial treatment was given to a patient.

Authors' contributions

ML designed data pre-processing, implemented and/or applied data mining tools, as well as drafted most parts of the manuscript. OT extracted and anonymized the original data as well as helped to draft the manuscript, TGC participated in data pre-processing, ZK identified the dataset, helped to define the clinical questions and helped to draft the manuscript, JE defined the clinical questions and helped to draft the manuscript.

Acknowledgments

We would like to thank the multidisciplinary team at Guy’s and St Thomas’ NHS Foundation Trust and the Farr Institute of Health Informatics Research for their assistance in data preparation. This work was supported in part by the Daniel Turnberg Fellowship of the UK Academy of Medical Sciences.

References

[1] J. Gama, Knowledge Discovery from Data Streams, Boca Raton, FL, USA: Chapman & Hall/CRC, 2010.

[2] M. Sayed-Mouchaweh and E. Lughofer, Learning in Non-Stationary Environments: Methods and Applications, New York, NY, USA: Springer, 2012.

[3] M. Collins, R. Schapire and Y. Singer, "Logistic regression, Adaboost and Bregman distances," *Machine Learning,* vol. 48, no. 1-3, pp. 253--285, 2002.

[4] L. Breiman, J. Friedman, C. Stone and R. Olshen, Classification and Regression Trees, Boca Raton: Chapman and Hall, 1993.

[5] J. R. Quinlan, C4. 5: programs for machine learning, San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1993.

[6] B. Sch\"olkopf and A. Smola, Learning with Kernels - Support Vector Machines, Regularization, Optimization and Beyond, London, England: MIT Press, 2002.

[7] M. Millán-Giraldo, J. S. Sánchez and V. J. Traver, "On-line learning from streaming data with delayed attributes: a comparison of classifiers and strategies," *Neural Computing and Applications,* vol. 20, no. 7, pp. 935-944, 2011.

[8] G. Krempl, I. Žliobaite, D. Brzeziński, E. Hüllermeier, M. Last, V. Lemaire and et al., "Open Challenges for Data Stream Mining Research," *ACM SIGKDD Explorations Newsletter,* vol. 16, no. 1, pp. 1-10, 2014.

[9] K. Strand and H. Flaatten, "Severity scoring in the ICU: a review," *Acta Anaesthesiologica Scandinavica,* vol. 52, no. 4, pp. 467-478, 2008.

[10] W. A. Knaus, E. A. Draper, D. P. Wagner and J. E. Zimmerman, "APACHE II: a severity of disease classification system," *Critical Care Medicine,* vol. 13, no. 10, pp. 818-829, 1985.

[11] S. Lemeshow, D. Teres, J. Klar, J. S. Avrunin, S. H. Gehlbach and J. Rapoport, "Mortality

Probability Models (MPM II) based on an international cohort of intensive care unit patients," *Jama,* vol. 270, no. 20, pp. 2478-2486, 1993.

[12] S. Lemeshow, J. Klar, D. Teres, J. S. Avrunin, S. H. Gehlbach, J. Rapoport and et al., "Mortality probability models for patients in the intensive care unit for 48 or 72 hours: a prospective, multicenter study," *Critical care medicine,* vol. 22, no. 9, pp. 1351-1358, 1994.

[13] J. Trujillano, M. Badia, L. Serviá, J. March and A. Rodriguez-Pozo, "Stratification of the severity of critically ill patients with classification trees," *BMC Med Res Methodol,* vol. 9, no. 1, p. 83, 2009.

[14] G. V. Kass, "An Exploratory Technique for Investigating Large Quantities of Categorical Data," *Applied Statistics,* vol. 29, no. 2, p. 119–127, 1980.

[15] P. Gago, C. Fernandes, F. Pinto and M. F. Santos, "INTCare: On-line knowledge discovery in the intensive care unit," in *International Conference on Intelligent Engineering Systems*, Barbados, 2009.

[16] J. D. Edgeworth, I. C. Ster, D. Wyncoll, M. Shankar-Hari and C. A. McKenzie, "Long-term adherence to a 5 day antibiotic course guideline for treatment of intensive care unit (ICU)-associated Gram-negative infections," *Journal of Antimicrobial Chemotherapy,* 2014.

[17] S. Zilberstein and S. Russell, "Approximate reasoning using anytime algorithms," in *Imprecise and Approximate Computation*, Boston, Springer US, 1995, pp. 43-62.

[18] M. F. Ghalwash, V. Radosavljevic and Z. Obradovic, "Utilizing temporal patterns for estimating uncertainty in interpretable early decision making," in *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining (KDD '14)*, New York, 2014.

[19] D. Hosmer and S. Lemeshow, Applied Logistic Regression, New York, NY: Wiley, 2005.

[20] M. Last and O. Maimon, "A compact and accurate model for classification," *IEEE Transactions on Knowledge and Data Engineering,* vol. 16, no. 2, pp. 203-215, 2004.

[21] M. Hofmann and R. Klinkenberg, Rapidminer: Data Mining Use Cases and Business Analytics Applications, Boca Raton: Chapman & Hall/CRC, 2013.

[22] I. H. Witten, E. Frank and M. A. Hall, Data Mining: Practical Machine Learning Tools and Techniques, 3rd ed., San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2011.

[23] J. R. Quinlan, "Induction of Decision Trees," *Machine Learning,* vol. 1, no. 1, pp. 81-106, 1986.

[24] E. DeLong, D. DeLong and D. Clarke-Pearson, "Comparing the areas under two or more correlated receiver operating characteristic curves: a nonparametric approach," *Biometrics,* vol. 44, pp. 837-845, 1988.

[25] D. A. Harrison, A. R. Brady, G. J. Parry, J. R. Carpenter and K. Rowan, "Recalibration of risk prediction models in a large multicenter cohort of admissions to adult, general critical care units in the United Kingdom," *Critical care medicine,* vol. 34, no. 5, pp. 1378-1388, 2006.

[26] M. Kayaalp, G. F. Cooper and G. Clermont, "Predicting ICU mortality: a comparison of stationary and nonstationary temporal models," in *Proceedings of the AMIA Symposium*, Bethesda, MD, USA, 2000.

Appendix

Table 10 Attributes - Descriptive Statistics

Feature Name	Min	Max	Average	Standard Deviation
Precipitating Factor Code	1	39	19.36	12.07
APACHE Score	0	53	16.16	5.87
AcutePhysioScore	0	43	11.97	5.30
GCS	0	17	4.89	6.83
AgePoints	0	6	3.56	2.12
ChronicHealthPoints	0	10	0.87	1.91
Temp	0	43	37.84	1.69
SBP	0	286	112.74	35.80
HR	0	265	94.80	31.45
Bicarb	0	258	23.82	4.65
INR	0	42	1.25	0.61
Resp	0	117	25.73	13.22
Urine	0	13002400	2361.67	72599.37
Glucose	0	1609.2	72.25	78.15
PaO2	0	3090.254	52.80	47.36
PaCO2	0	558.0459	26.83	20.60
pH	0	744	7.35	8.21
Haematocrit	0	83	29.13	4.98
WBC	0	100	13.99	7.83
Platelets	0	1388	239.16	166.06
Sodium	0	1455	141.16	12.81
Potassium	0	4135	4.32	23.10

Feature Name	Min	Max	Average	Standard Deviation
Creatinine	-100	5959	63.87	112.24
Urea	0	859.6	37.68	43.24
SGOT	0	7774	88.75	266.04
Albumin	0	400	12.83	13.22
Bilirubin	0	2137	12.99	42.43

Table 11 Microbiology Day 2 Rules

Rule No.		Prob. (A)	Prob. (D)
0	If Diagnosis Label is Cancer of kidney and renal pelvis	0.7449	0.2551
1	If Diagnosis Label is Cancer of uterus	0.8324	0.1676
2	If Diagnosis Label is Cancer of colon and APACHE Score is greater than or equal to 1 and less than 13	0.875	0.125
3	If Diagnosis Label is Cancer of colon and APACHE Score is greater than or equal to 13 and less than 21	0.7868	0.2132
4	If Diagnosis Label is Cancer of colon and APACHE Score is greater than or equal to 21 and less than 25	0.6387	0.3613
5	If Diagnosis Label is Cancer of colon and APACHE Score is equal to 25 and higher	0.5896	0.4104
6	If Diagnosis Label is Cancer of rectum and anus and APACHE Score is greater than or equal to 1 and less than 13	0.8333	0.1667
7	If Diagnosis Label is Cancer of rectum and anus and APACHE Score is greater than or equal to 13 and less than 21	0.7172	0.2828
8	If Diagnosis Label is Cancer of rectum and anus and APACHE	0.6047	0.3953

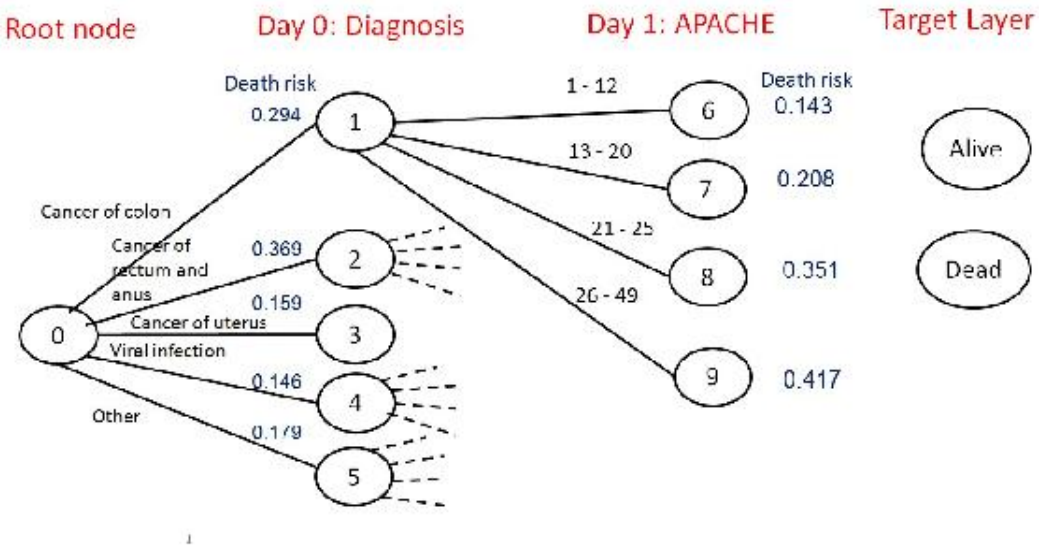
Rule No.		Prob. (A)	Prob. (D)
	Score is greater than or equal to 21 and less than 25		
9	If Diagnosis Label is Cancer of rectum and anus and APACHE Score is equal to 25 and higher	0.3585	0.6415
10	If Diagnosis Label is Viral infection and APACHE Score is greater than or equal to 1 and less than 13	0.9306	0.0694
11	If Diagnosis Label is Viral infection and APACHE Score is greater than or equal to 13 and less than 21	0.8522	0.1478
12	If Diagnosis Label is Viral infection and APACHE Score is greater than or equal to 21 and less than 25	0.7719	0.2281
13	If Diagnosis Label is Viral infection and APACHE Score is equal to 25 and higher	0.6341	0.3659
14	If Diagnosis Label is Other and APACHE Score is greater than or equal to 13 and less than 21	0.8464	0.1536
15	If Diagnosis Label is Other and APACHE Score is greater than or equal to 21 and less than 25	0.746	0.254
16	If Diagnosis Label is Other and APACHE Score is equal to 25 and higher	0.5214	0.4786
17	If Diagnosis Label is Other and APACHE Score is greater than or equal to 1 and less than 13 and blood gram-negative is 0	0.9722	0.0278
18	If Diagnosis Label is Other and APACHE Score is greater than or equal to 1 and less than 13 and blood gram-negative is 1	0.7368	0.2632

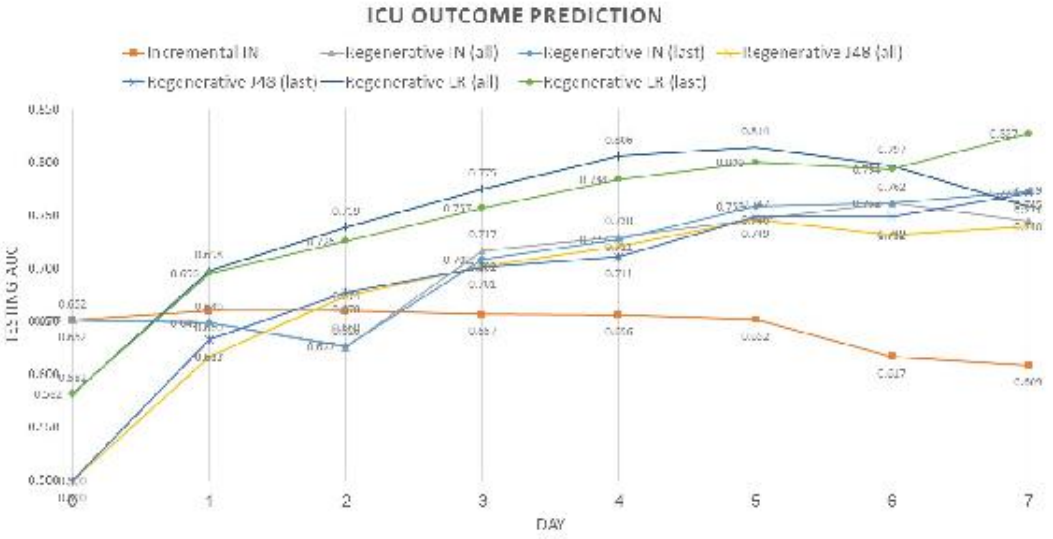
Figure legends

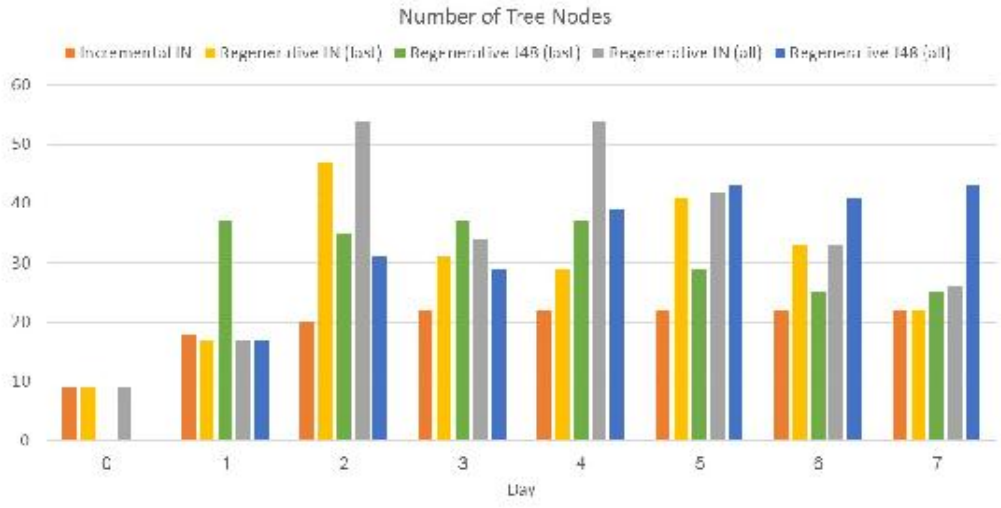
Figure **1.** Example of a temporal single-target information network

Figure 2 ICU outcome prediction results

Figure 3 Size of decision-tree models


Root node
Day 0: Diagnosis
Day 1: APACHE
Target Layer
Death risk
0.294
Cancer of colon
Cancer of rectum and anus
0.369
0.159
Cancer of uterus
Viral infection
0.146
Other
0.179
0
1
2
3
4
5
1 - 12
13 - 20
21 - 25
26 - 49
6
7
8
9
Death risk
0.143
0.208
0.351
0.417
Alive
Dead


ICU OUTCOME PREDICTION
Incremental IN
Regenerative IN (all)
Regenerative IN (last)
Regenerative J48 (all)
Regenerative J48 (last)
Regenerative LR (all)
Regenerative LR (last)
TESTING AUC
DAY
0
1
2
3
4
5
6
7


Number of Tree Nodes
Incremental IN
Regenerative IN (last)
Regenerative I48 (last)
Regenerative IN (all)
Regenerative I48 (all)
60
50
40
30
20
10
0
0
1
2
3
4
5
6
7
Day