Localized customized mortality prediction modeling for patients with acute kidney injury admitted to the intensive care unit

By

Leo Anthony G. Celi

B.S. Basic Medical Science
University of the Philippines, 1985
M.D.
University of the Philippines, 1990

SUBMITTED TO THE DIVISION OF HEALTH SCIENCES AND TECHNOLOGY IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF

MASTER OF SCIENCE IN BIOMEDICAL INFORMATICS
AT THE
MASSACHUSETTS INSTITUTE OF TECHNOLOGY

MAY 2009

Signature of Author: ____
Division of Health Sciences and Technology
May 15, 2009

Certified by: ____
Peter Szolovits Ph.D.
Professor of Computer Science and Engineering
& Health Sciences and Technology
Thesis Supervisor

Accepted by:____
Ram Sasisekharan, Ph.D.
Professor of Computer Science and Engineering
& Health Sciences and Technology
Chairman, Committee for Graduate Students

LOCAL CUSTOMIZED MORTALITY PREDICTION MODELING FOR PATIENTS WITH ACUTE KIDNEY INJURY ADMITTED TO THE INTENSIVE CARE UNIT

By

LEO ANTHONY CELI, M.D.

Submitted to the Division of Health Sciences and Technology
on May 18, 2009 in partial fulfillment of the requirements for the
Degree of Master of Science in Biomedical Informatics

ABSTRACT

Introduction. Models for mortality prediction are traditionally developed from prospective multi-center observational studies involving a heterogeneous group of patients to optimize external validity. We hypothesize that local customized modeling using retrospective data from a homogeneous subset of patients will provide a more accurate prediction than this standard approach. We tested this hypothesis on patients admitted to the ICU with acute kidney injury (AKI), and evaluated variables from the first 72 hours of admission. *Methods*. The Multi-parameter Intelligent Monitoring for Intensive Care II (MIMIC II) is a database of patients admitted to the Beth Israel Deaconess Medical Center ICU. Using the MIMIC II database, we identified patients who developed acute kidney injury and who survived at least 72 hours in the ICU. 118 variables were extracted from each patient. Second and third level customization of the Simplified Organ Failure Score (SAPS) was performed using logistic regression analysis and the best fitted models were compared in terms of Area under the Receiver Operating Characteristic Curve (AUC) and Hosmer-Lemeshow Goodness-of-Fit test (HL). The patient cohort was divided into a training and test data with a 70:30 split. Ten-fold cross-validation was performed on the training set for every combination of variables that were evaluated. The best fitted model from the cross-validation was then evaluated using the test set, and the AUC and the HL p value on the test set were reported. *Results*. A total of 1400 patients were included in the study. Of these, 970 survived and 430 died in the hospital (30.7% mortality). We observed progressive improvement in the performance of SAPS on this subset of patients (AUC=0.6419, HL p=0) with second level (AUC=0.6639, HL p=0.2056), and third level (AUC=0.7419, HL p=0.6738) customization. The best fitted model incorporated variables from the first 3 days of ICU admission. The variables that were most predictive of hospital mortality in the multivariate analysis are the maximum blood urea nitrogen and the minimum systolic blood pressure from the third day. *Conclusion*. A logistic regression model built using local data for patients with AKI performed better than SAPS, the current standard mortality prediction scoring system.

Thesis Supervisor: Peter Szolovits, Ph,D.

Title: Professor of Computer Science and Engineering & Health Sciences and Technology

Introduction

There are numerous severity scoring systems that are available in the intensive care unit (ICU), including Acute Physiology and Chronic Health Evaluation (APACHE), Simplified Acute Physiology (SAPS), and Multiple Organ Dysfunction Score (MODS). Although initially designed for mortality prediction, they universally lack clinically acceptable accuracy at an individual patient level [1]. These systems perform relatively well in predicting how many patients will die in an ICU when the individual patient risks are calculated and averaged for that ICU. Thus, these severity scoring systems are currently used primarily for case-mix determination and benchmarking purposes. However, although the prognostic accuracy of the scoring system for an entire ICU population is good, its prognostic accuracy at different levels of risk, or its calibration, is poor. This suggests that within an ICU, there are groups of patients whose risk of death is over-estimated that are "balanced" by the groups of patients whose risk of death is under-estimated. It is also important to note that the performance of these predictive models is always better on whole ICU populations than on specific subsets of ICU patients. For example, they have significantly underestimated the risk of death among patients who develop acute kidney injury (AKI) [2, 3].

Another feature of these severity scoring systems is their variable accuracy among ICU populations in different regions of the world [1]. Not surprisingly, the system performs well on ICU population that is similar to the group of patients whose data were used to build the predictive model. For example, SAPS II, which was developed using data from mostly European ICUs[4], performs well in France but not in the US (AUC = 0.67, Hosmer-Lemeshow p value = 0.05) [5] . But even among ICUs in Europe, the performance of the SAPS for case-mix determination wanes over time. This is referred to as model fade, and is the reason why newer versions of scoring systems are released and replace older versions. APACHE was developed from a North American database using logistic regression on patient variables obtained during the first 24 hours in the ICU [6]. APACHE I was built using 34 patient variables while APACHE IV utilized 142 variables [7]. MODS [8]was also built using North American database. Like SAPS, the performance of APACHE and MODS in different geographic regions varies and their calibration is poor even among ICU groups where they perform well [1]. Even when much bigger patient cohorts from more regions of the world are used, there has been no consistent improvement in calibration looking at studies evaluating the performance of the different scoring systems in the last 10 years [3].

ICU patients who develop AKI are one subset of patients where severity scoring systems have consistently performed poorly. AKI develops in approximately 6% of critically ill patients; two-thirds of these patients require renal support therapy [2]. The largest worldwide multi-center prospective study found that the observed mortality among these patients was substantially greater than predicted by SAPS II (60.3% vs. 45.6%, $p < 0.001$). In another UK-wide study of ICU patients who develop AKI, the APACHE II score under-predicted the number of deaths [3]. In this study, the null hypothesis of perfect calibration was strongly rejected ($p < 0.001$) by both the Hosmer-Lemeshow and Cox's calibration regression.

Several methods have been proposed to improve the performance of existing severity scoring systems. Customization is a simple procedure that adapts a model to specific patient populations [9]. In second level customization, a multivariate regression is run on the same variables included in the severity score on the new database. The new coefficients generated by the model are then used to calculate the "new" severity score. In third level customization, additional variables are evaluated either in addition to the severity score or the original variables included in the severity score. These methods of customization have been successfully used in improving accuracy and calibration of existing severity scoring systems on patients from countries not represented in the original database, as well as for specific subgroup of ICU patients [10].

Another approach that has been used to improve mortality prediction is to calculate severity scores over a period of time [11, 12]. Severity scores on admission to the ICU ignore the many factors that can influence patient outcomes during the course of an ICU stay. These factors include, but are not limited to, the quality of care – the accuracy and timeliness of diagnosis and provision of appropriate treatment – and the patient's inherent ability to heal as reflected by his response to therapy. Being able to evaluate changes in patient status over time thus represents an improvement on severity scores on admission.

Daily calculation of Sequential Organ Failure Assessment (SOFA) and Logistic Organ Dysfunction (LOD) scores has been evaluated in research studies [13]. These studies have demonstrated that models based on temporal patterns outperformed those based on physiologic variables during the first 24 hours of ICU admission. However, these models remain very poorly calibrated, preventing their adoption in clinical practice to guide management decisions for individual patients. Poor calibration of these models has been attributed to the observation that the influence and contribution of the variables on mortality change over time [14].

At present, ICU clinicians focus on and evaluate a subset of physiologic variables and their evolution over time when deciding whether to carry on with an individual patient or whether to recommend switching to comfort measures only. The selection of which variables are important is based on clinical intuition and experience, and likely varies from patient to patient and from one clinician to another.

Over the last few decades, the associated mortality of patients with AKI has remained largely unchanged (even after adjustment for age and severity of illness) despite advances in ICU care, including renal and other organ support therapy [15, 16]. Dialysis and/or filtration in the ICU is not only costly but also consumes a significant fraction of nursing time diverted from tasks that may be more beneficial in terms of patient outcomes. To date, no AKI-specific severity of illness scoring method has exhibited excellent predictive power for mortality [3]. Such a system might assist clinicians in predicting which patients would benefit from renal support therapy.

Rather than developing predictive models with good discrimination and calibration among general ICU populations in different regions of the world, we build a case for "local", customized models for homogeneous patient subsets built on patients from one's own ICU database.

The specific questions we addressed are as follows:

1. To determine whether customization using local institutional data on patients with AKI will perform better compared to SAPS in predicting mortality

2. To assess whether variables over the first 72 hours in the ICU can predict mortality better than variables obtained during the first 24 hours of admission among this subset of patients

3. To evaluate whether the addition of selected interaction terms into the best fitted logistic regression model would improve the accuracy

4. To compare the logistic regression models using automated and heuristically-driven variable selection

5. To evaluate whether filtering using principal component analysis can improve the accuracy of mortality prediction models built on patients with AKI

6. To compare different machine learning algorithms in predicting mortality among this subset of ICU patients

Methods

The Laboratory of Computational Physiology at Massachusetts Institute of Technology (MIT) developed and maintains the Multi-parameter Intelligent Monitoring for Intensive Care (MIMIC II) database, a high resolution database of ICU patients admitted to the Beth Israel Deaconess Medical Center (BIDMC) since 2003 that has been de-identified by removal of all Protected Health Information [17]. An Institutional Review Board (IRB) approval was obtained from both MIT and BIDMC for the development, maintenance and public use of a de-identified ICU database. The MIMIC II database currently consists of data from more than 18,000 patients that has been de-identified and formatted to facilitate data-mining. The 3 sources of data are waveform data collected from the bedside monitors, hospital information systems and other third party clinical information systems.

Using the MIMIC II database, we identified the patients who had an ICD-9 diagnosis of acute renal failure (584.9) and who survived at least 72 hours in the ICU. We verified whether the patients developed acute kidney injury at around the time of ICU admission by looking at the serum creatinine and urine output during the first 72 hours in the ICU. Patients whose serum creatinine determinations were less than 1.0 mg/dl and who had an average urine output of 0.5 ml/kg/hr during the first three days of their ICU stay were excluded from the cohort, as they are unlikely to have sustained acute kidney injury at around the time of ICU admission.

<u>Variable Selection</u>

The outcome of interest is survival to hospital discharge. The covariates that were evaluated included demographic factors (age and sex), SAPS, Glasgow Coma Score (GCS), and physiologic variables measured during the first three days in the ICU. We obtained the minimum, maximum, standard deviation and average value of the majority of the physiologic variables. For variables where a low value does not have clinical significance during critical illness (e.g. serum creatinine, serum bilirubin, blood urea nitrogen), only the maximum values were extracted from the database. We only included the worst Glasgow Coma Score (GCS) on presentation to the ICU because as soon as a patient is intubated, GCS becomes clinically irrelevant as a patient is typically given medications in order to tolerate the endotracheal tube. At this time, a low GCS is no longer a reflection of central nervous system dysfunction. Finally, some variables were excluded (minimum temperature, minimum respiratory rate) because a significant fraction of the data was deemed inaccurate after manual inspection. This will be explained further in the discussion section of this paper.

Inclusion of the minimum, maximum and average values, and the standard deviation, which are not independent, may represent redundant variables. However, noise reduction and consequently better class separation may be obtained by adding variables that are presumably redundant [18]. Only perfectly correlated variables are truly redundant in the sense that no additional information is gained by adding them. The complete list of variables that were obtained and evaluated is found in Table 1.

SAPS	AGE	SEX
MIN_GCS_1ST_DAY		
OUTPUT_1ST_DAY OUTPUT_2ND_DAY OUTPUT_3RD_DAY	MAX_CREAT_1ST_DAY MAX_CREAT_2ND_DAY MAX_CREAT_3RD_DAY	MAX_BUN_1ST_DAY MAX_BUN_2ND_DAY MAX_BUN_3RD_DAY
MAX_BILI_1ST_DAY MAX_BILI_2ND_DAY MAX_BILI_3RD_DAY	MAX_TEMP_1ST_DAY MAX_TEMP_2ND_DAY MAX_TEMP_3RD_DAY	MAX_RESP_1ST_DAY MAX_RESP_2ND_DAY MAX_RESP_3RD_DAY
MIN_HR_1ST_DAY MAX_HR_1ST_DAY STDDEV_HR_1ST_DAY AVG_HR_1ST_DAY	MIN_HR_2ND_DAY MAX_HR_2ND_DAY STDDEV_HR_2ND_DAY AVG_HR_2ND_DAY	MIN_HR_3RD_DAY MAX_HR_3RD_DAY STDDEV_HR_3RD_DAY AVG_HR_3RD_DAY
MIN_SYSBP_1ST_DAY MAX_SYSBP_1ST_DAY STDDEV_SYSBP_1ST_DAY AVG_SYSBP_1ST_DAY	MIN_SYSBP_2ND_DAY MAX_SYSBP_2ND__DAY STDDEV_SYSBP_2ND__DAY AVG_SYSBP_2ND__DAY	MIN_SYSBP_3RD_DAY MAX_SYSBP_3R D__DAY STDDEV_SYSBP_3R D__DAY AVG_SYSBP_3R D__DAY
MIN_SODIUM_1ST_DAY MAX_SODIUM_1ST_DAY STDDEV_SODIUM_1ST_DAY AVG_SODIUM_1ST_DAY	MIN_SODIUM_2ND_DAY MAX_SODIUM_2ND _DAY STDDEV_SODIUM_2ND _DAY AVG_SODIUM_2ND _DAY	MIN_SODIUM_3RD_DAY MAX_SODIUM_3RD _DAY STDDEV_SODIUM_3RD_DAY AVG_SODIUM_3RD _DAY
MIN_POTASSIUM_1ST_DAY MAX_POTASSIUM_1ST_DAY STDDEV_POTASSIUM_1ST_DAY AVG_POTASSIUM_1ST_DAY	MIN_POTASSIUM_2ND_DAY MAX_POTASSIUM_2ND _DAY STDDEV_POTASSIUM_2ND _DAY AVG_POTASSIUM_2ND _DAY	MIN_POTASSIUM_3RD_DAY MAX_POTASSIUM_3RD _DAY STDDEV_POTASSIUM_3RD _DAY AVG_POTASSIUM_3RD _DAY
MIN_GLUCOSE_1ST_DAY MAX_ GLUCOSE _1ST_DAY STDDEV_ GLUCOSE _1ST_DAY AVG_ GLUCOSE _1ST_DAY	MIN_ GLUCOSE _2ND_DAY MAX_ GLUCOSE _2ND _DAY STDDEV_ GLUCOSE _2ND _DAY AVG_ GLUCOSE _2ND _DAY	MIN_ GLUCOSE _3RD_DAY MAX_ GLUCOSE _3RD _DAY STDDEV_ GLUCOSE _3RD _DAY AVG_ GLUCOSE _3RD _DAY
MIN_ BICARBONATE _1ST_DAY MAX_ BICARBONATE _1ST_DAY STDDEV_ BICARBONATE _1ST_DAY AVG_ BICARBONATE _1ST_DAY	MIN_ BICARBONATE _2ND_DAY MAX_ BICARBONATE _2ND _DAY STDDEV_ BICARBONATE _2ND _DAY AVG_ BICARBONATE _2ND _DAY	MIN_ BICARBONATE _3RD_DAY MAX_ BICARBONATE _3RD _DAY STDDEV_ BICARBONATE _3RD _DAY AVG_ BICARBONATE _3RD _DAY
MIN_ WBC _1ST_DAY MAX_ WBC _1ST_DAY STDDEV_ WBC _1ST_DAY AVG_ WBC _1ST_DAY	MIN_ WBC _2ND_DAY MAX_ WBC _2ND _DAY STDDEV_ WBC _2ND _DAY AVG_ WBC _2ND _DAY	MIN_ WBC _3RD_DAY MAX_ WBC _3RD _DAY STDDEV_ WBC _3RD _DAY AVG_ WBC _3RD _DAY
MIN_ HEMATOCRIT _1ST_DAY MAX_ HEMATOCRIT _1ST_DAY STDDEV_ HEMATOCRIT _1ST_DAY AVG_ HEMATOCRIT _1ST_DAY	MIN_ HEMATOCRIT _2ND_DAY MAX_ HEMATOCRIT _2ND _DAY STDDEV_ HEMATOCRIT _2ND _DAY AVG_ HEMATOCRIT _2ND _DAY	MIN_ HEMATOCRIT _3RD_DAY MAX_ HEMATOCRIT _3RD _DAY STDDEV_ HEMATOCRIT _3RD _DAY AVG_ HEMATOCRIT _3RD _DAY

Table 1. List of Features

A univariate logistic regression was performed on each of the variables to determine whether they are correlated with hospital mortality.

We employed two automated feature selection algorithms as well as domain knowledge for variable selection. The algorithms were correlation-based feature subset selection (CFS) and consistency subset evaluation using the best first search method (greedy hill-climbing with backtracking). Correlation-based feature subset selection assesses the predictive ability of each attribute individually and the degree of redundancy among them, preferring sets of attributes that are highly correlated with the outcome but have low inter-correlation [19]. Consistency subset evaluation assesses attribute sets by the degree of consistency in class values when the training instances are projected onto the set. The consistency of any subset of attributes can never improve on that of the full set, so that this evaluator seeks the smallest subset whose consistency is the same as that of the full attribute set.

We also evaluated various combinations of the variables based on domain expertise. Variables with a p value greater than 0.05 in the univariate or in a multivariate analysis were still considered for inclusion or were retained in a model, as they can still contribute to the performance of a model with good discrimination and calibration.

Pre-Processing

Instead of replacing the missing values with the mean for variables with Gaussian distribution or the median for all other variables, we applied a set of rules derived from clinical experience. The rules are as follows:

1. If the values for a certain variable are missing for all three days, they are replaced by the middle value of the normal range of that variable. If a variable is not measured in the ICU, e.g. serum bilirubin, there is a good likelihood that there is no concern that this may be abnormal.

2. A missing value on the second ICU day was replaced with the average of the first and third day values. A missing value on the first or third ICU day was replaced with the second day value.

3. If values on two of the first three days in the ICU are missing, they are both replaced with the value that is present.

Statistical Analysis

The patient cohort was divided into a training and test data with a 70:30 split. The test set was not used to build any of the models. Ten-fold cross-validation was performed on the training set ten times for every combination of variables that were evaluated. Two sets of AUC and Hosmer-Lemeshow (HL) p value were obtained for each model. The first is the average of the ten values obtained from each of the cross-validation run. The second was obtained by evaluating the model that performed the best on the training set on the test data, in order to eliminate a learning bias. Only the second set of AUC and HL p value is reported.

The SAPS of each patient was converted to predicted mortality using the following formula: Predicted Death Rate = $e^{(Logit)}/(1+e^{(Logit)})$, where Logit = -7.7631+0.0737*SAPS+0.9971*ln(SAPS+1) [8]. The predicted death rate was compared against the true outcome, and an Area under the Receiver Operating Characteristic Curve (AUC) was calculated. This AUC is used as the benchmark to compare the AUC's of the customized models.

Using the R software (R version 2.7.2, The R Foundation for Statistical Computing, Auckland, New Zealand), second level customization was performed by building a multivariate regression model using the physiologic variable components of SAPS. Ten-fold cross-validation was performed on the training set and the best-fitted model was evaluated on the test set.

Third level customization was performed by evaluating various combinations of variables from the first three days in the ICU to predict mortality among these patients with AKI. We built logistic regression model using the variables selected by the correlation-based feature subset evaluator and the consistency subset evaluator. We did the same for the different combination of variables selected based on domain expertise. We then compared the performance of the best fitted models on the test data.

A number of interaction terms, selected based on clinical knowledge, were evaluated to see whether they improved the performance of the best multivariate logistic regression models. These included:

1. Blood pressure and heart rate – The effect of the heart rate on mortality might differ at different blood pressure levels. Bradycardia is deleterious among patients who are hypotensive, but may be protective for those who are normotensive by reducing the oxygen requirement of the heart.

2. Age and serum creatinine – The effect of serum creatinine on mortality might differ at different age groups.

3. BUN and serum creatinine – Protein catabolism is more marked among patients who are sicker. This results in an increased nitrogen load to the kidneys for excretion. The consequence of protein catabolism, as reflected by the blood urea nitrogen, might differ among patients depending on their kidney function.

4. Sex and serum creatinine - The effect of serum creatinine on mortality might differ between men and women.

5. Glucose standard deviation and GCS – The influence of glucose variability, a result of impairment in neurohormonal mechanisms, on clinical outcome might vary according to the level of CNS dysfunction, as measured by the GCS.

To compare different machine learning algorithms in predicting hospital mortality, Weka version 3.5.7 (University of Waikato, Hamilton, New Zealand) was used. Weka uses the Quasi-Newton method to optimize outcome prediction.

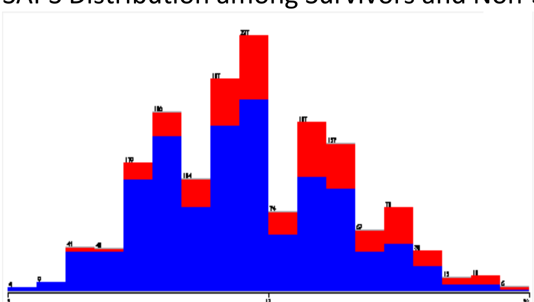
<u>Filtering</u>

Principal component analysis was performed on the training set to filter these data using the first four largest eigenvectors, accounting for 99.75% of the variability. We performed filtering by collapsing the Eigenvectors down and re-projecting the data back into the original space. Logistic regression analysis was then performed on the filtered data using ten-fold cross-validation to determine whether the accuracy of prediction can be improved by using this noise reduction approach.

Results

There were a total of 1400 patients with an ICD-9 diagnosis of acute renal failure who survived at least 72 hours in the ICU. Of these, 970 survived and 430 died in the hospital (30.7% mortality). These were divided into a training set (979 patients) and test set (421 patients). The difference between the mortality rate of the training set (31.9%) and test set (28.2%) is not statistically significant ($p > 0.05$). The difference in the distributions of the variables in the training and test sets is also not statistically significant (data not shown). The distributions of selected variables in the entire patient cohort among survivors and non-survivors are shown below.

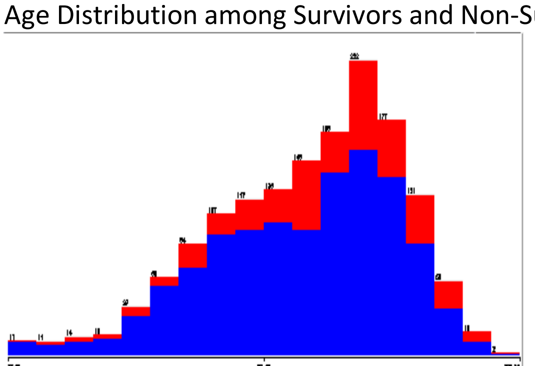
A. SAPS Distribution among Survivors and Non-Survivors



Red:
Mean (Survivors) = 15.6
Standard Deviation (Survivors) = 5.2

Blue:
Mean (Non-Survivors) = 18.3
Standard Deviation (Non-Survivors) = 5.2

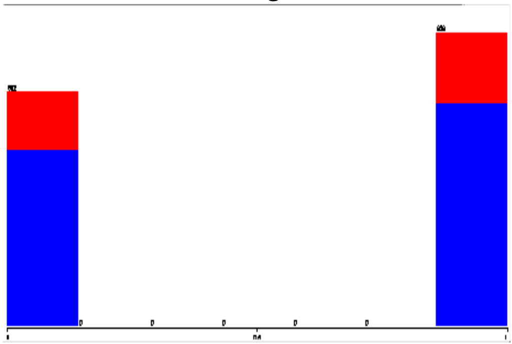
B. Age Distribution among Survivors and Non-Survivors



Red:
Mean (Survivors) = 68.0
Standard Deviation (Survivors) = 16.0

Blue:
Mean (Non-Survivors) = 71.9
Standard Deviation (Non-Survivors) = 15.6

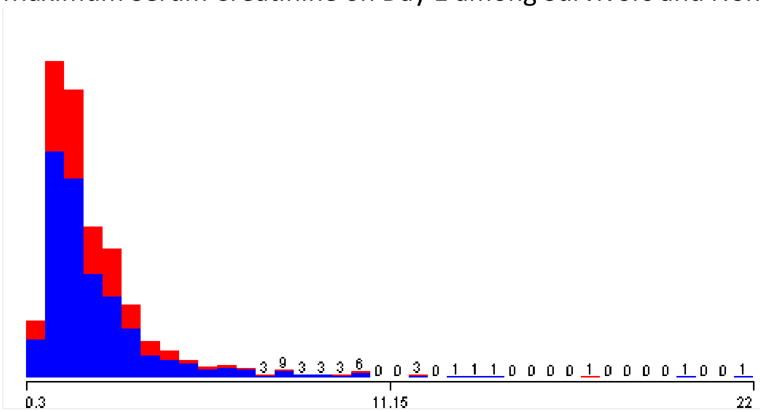
C. Sex Distribution among Survivors and Non-Survivors



Survivors:
561 Males (57.8%)
409 Females (42.2%)

Non-Survivors:
244 Males (56.6%)
187 Females (43.4%)

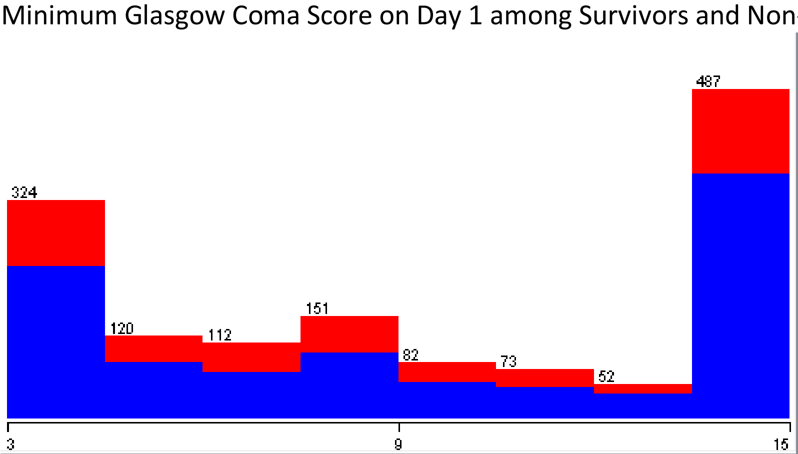
D. Maximum Serum Creatinine on Day 1 among Survivors and Non-Survivors



Red:
Mean (Survivors) = 2.41 mg/dl
Standard Deviation (Survivors) = 1.96 mg/dl

Blue:
Mean (Non-Survivors) = 2.29 mg/dl
Stand. Deviation (Non-Survivors) = 1.57 mg/dl

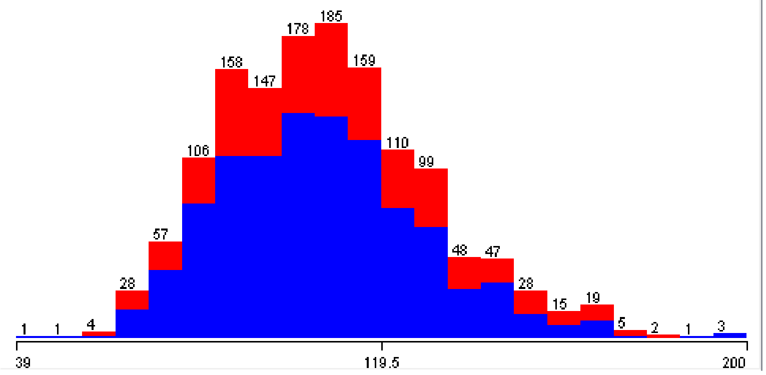
E. Minimum Glasgow Coma Score on Day 1 among Survivors and Non-Survivors



Red:
Mean (Survivors) = 9.6
Standard Deviation (Survivors) = 4.7

Blue:
Mean (Non-Survivors) = 9.1
Standard Deviation (Non-Survivors) = 4.4

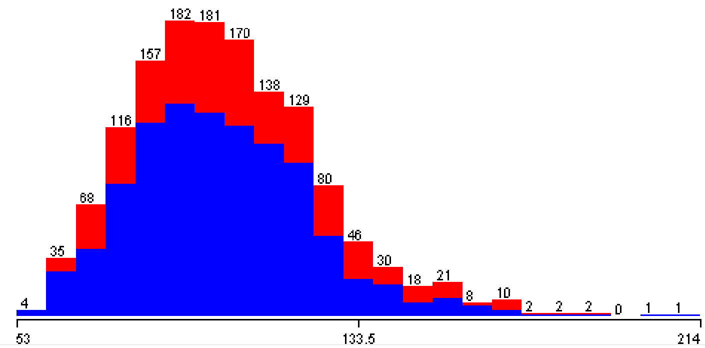
F. Maximum Heart Rate on Day 1 among Survivors and Non-Survivors



Red:
Mean (Survivors) = 106.2
Standard Deviation (Survivors) = 22.1

Blue:
Mean (Non-Survivors) = 109.7
Standard Deviation (Non-Survivors) = 25.3

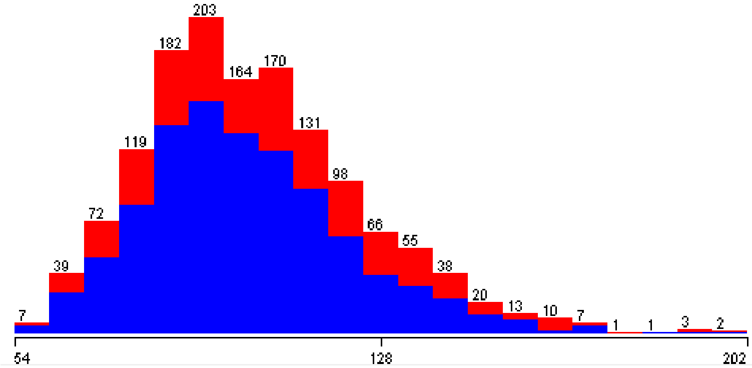
G. Maximum Heart Rate on Day 2 among Survivors and Non-Survivors



Red:
Mean (Survivors) = 101.9
Standard Deviation (Survivors) = 21.2

Blue:
Mean (Non-Survivors) = 105.7
Standard Deviation (Non-Survivors) = 23.4

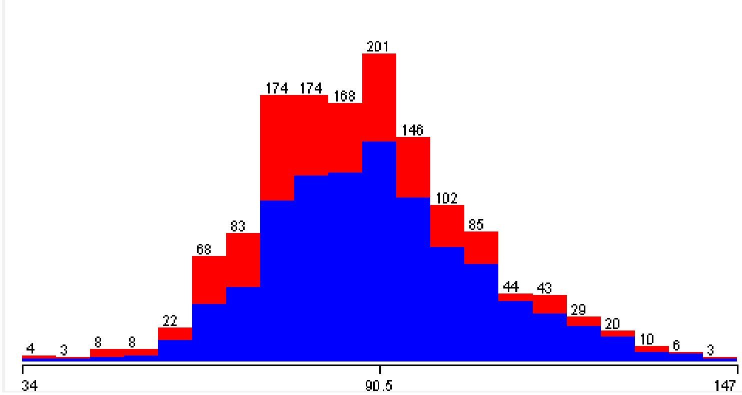
H. Maximum Heart Rate on Day 3 among Survivors and Non-Survivors



Red:
Mean (Survivors) = 101.2
Standard Deviation (Survivors) = 20.8

Blue:
Mean (Non-Survivors) = 105.8
Standard Deviation (Non-Survivors) = 24.4

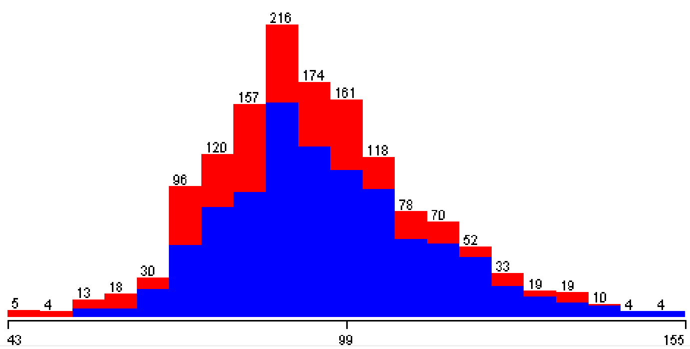
I. Minimum Systolic Blood Pressure on Day 1 among Survivors and Non-Survivors



Red:
Mean (Survivors) = 90.0 mmHg
Standard Deviation (Survivors) = 16.7 mmHg

Blue:
Mean (Non-Survivors) = 84.9 mmHg
Stand. Deviation (Non-Survivors) = 17.0 mmHg

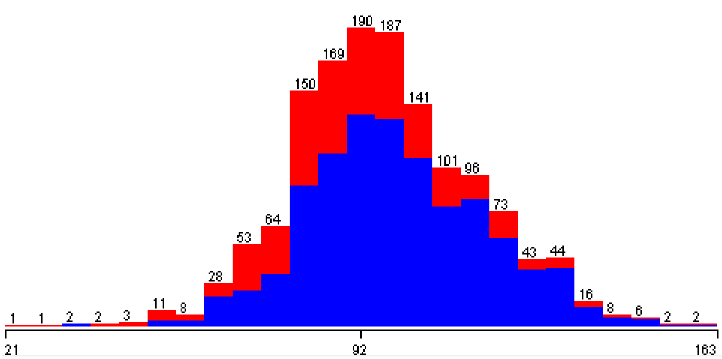
J. Minimum Systolic Blood Pressure on Day 2 among Survivors and Non-Survivors



Red:
Mean (Survivors) = 96.1 mmHg
Standard Deviation (Survivors) = 17.0 mmHg

Blue:
Mean (Non-Survivors) = 90.5 mmHg
Stand. Deviation (Non-Survivors) = 17.6 mmHg

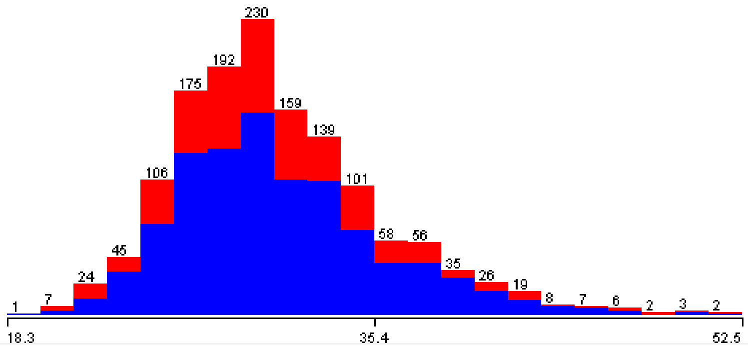
K. Minimum Systolic Blood Pressure on Day 3 among Survivors and Non-Survivors



Red:
Mean (Survivors) = 99.1 mmHg
Standard Deviation (Survivors) = 17.8 mmHg

Blue:
Mean (Non-Survivors) = 91.3 mmHg
Stand. Deviation (Non-Survivors) = 18.9 mmHg

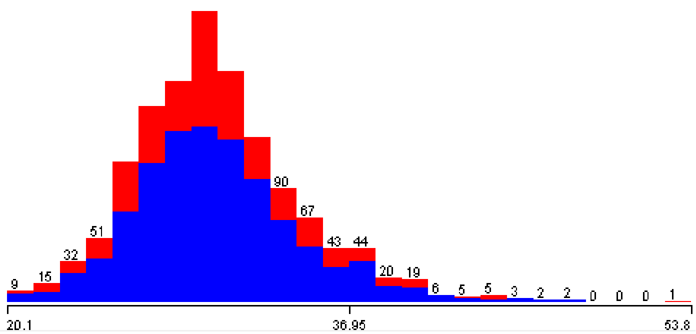
L. Average Serum Hematocrit on Day 1 among Survivors and Non-Survivors



Red:
Mean (Survivors) = 31.0%
Standard Deviation (Survivors) = 4.9%

Blue:
Mean (Non-Survivors) = 30.7%
Standard Deviation (Non-Survivors) = 4.9%

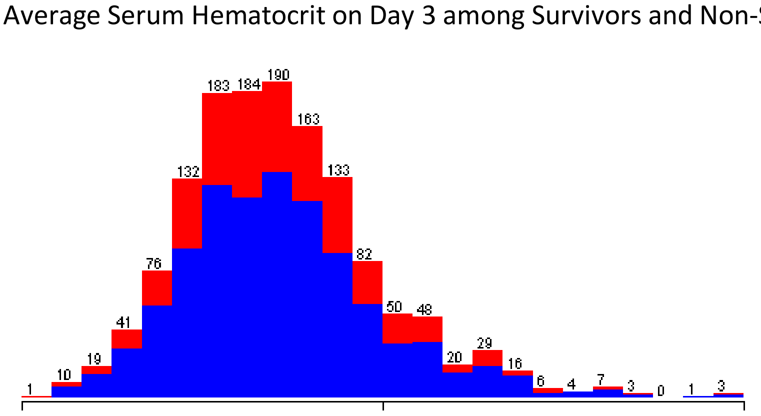
M. Average Serum Hematocrit on Day 2 among Survivors and Non-Survivors



Red:
Mean (Survivors) = 30.4 %
Standard Deviation (Survivors) = 4.2%

Blue:
Mean (Non-Survivors) = 30.4%
Standard Deviation (Non-Survivors) = 4.2%

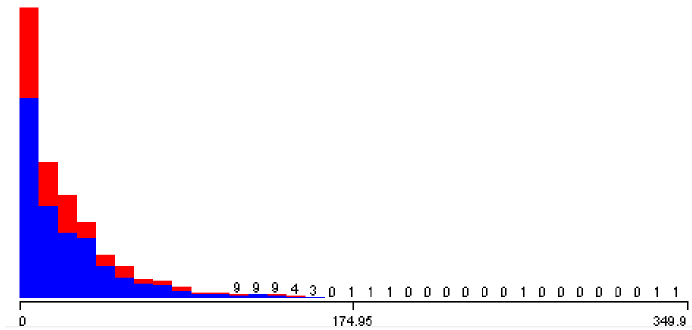
N. Average Serum Hematocrit on Day 3 among Survivors and Non-Survivors



Red:
Mean (Survivors) = 30.4 %
Standard Deviation (Survivors) = 4.0%

Blue:
Mean (Non-Survivors) = 30.4%
Standard Deviation (Non-Survivors) = 3.9%

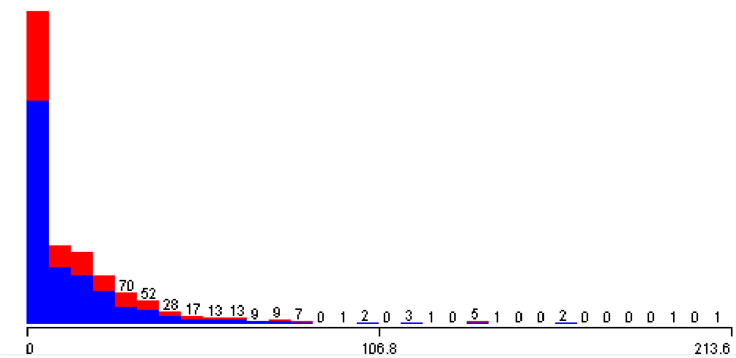
O. Glucose Variability (Standard Deviation) on Day 1 among Survivors and Non-Survivors



Red:
Mean (Survivors) = 26.1 mg/dl

Blue:
Mean (Non-Survivors) = 24.9 mg/L

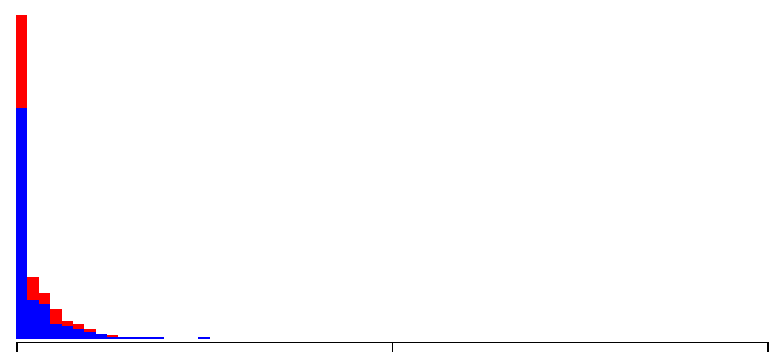
P. Glucose Variability (Standard Deviation) on Day 2 among Survivors and Non-Survivors



Red:
Mean (Survivors) = 13.3 mg/dl

Blue:
Mean (Non-Survivors) = 15.3 mg/L

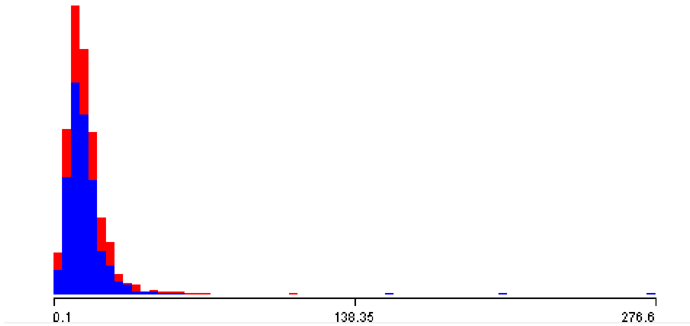
Q. Glucose Variability (Standard Deviation) on Day 3 among Survivors and Non-Survivors



Red:
Mean (Survivors) = 10.7 mg/dl

Blue:
Mean (Non-Survivors) = 12.8 mg/dl

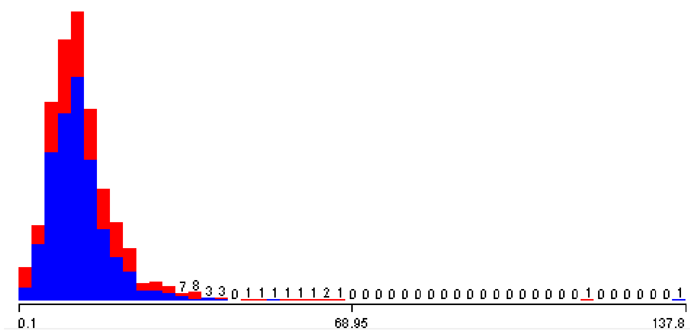
R. Maximum White Blood Count on Day 1 among Survivors and Non-Survivors



Red:
Mean (Survivors) = 14.5 x 10^3/uL
Standard Deviation (Survivors) = 13.5 x 10^3/uL

Blue:
Mean (Non-Survivors) = 15.8 x 10^3/uL
Stand. Deviation (Non-Survivors) =11.0 x 10^3/uL

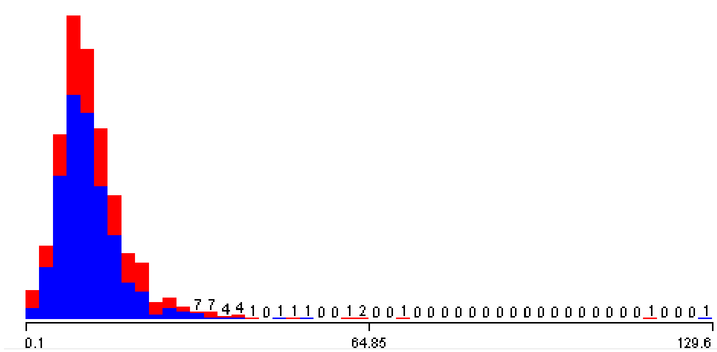
S. Maximum White Blood Count on Day 2 among Survivors and Non-Survivors



Red:
Mean (Survivors) =12.6 x 10^3/uL
Standard Deviation (Survivors) = 7.2 x 10^3/uL

Blue:
Mean (Non-Survivors) = 15.1 x 10^3/uL
Stand. Deviation (Non-Survivors) = 10.9 x 10^3/uL

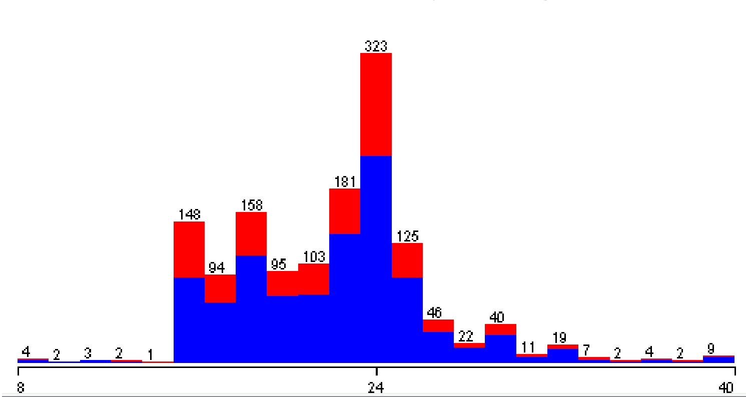
T. Maximum White Blood Count on Day 3 among Survivors and Non-Survivors



Red:
Mean (Survivors) =12.0 x 10^3/uL
Standard Deviation (Survivors) = 6.9 x 10^3/uL

Blue:
Mean (Non-Survivors) = 14.6 x 10^3/uL
Stand. Deviation (Non-Survivors) = 10.4 x 10^3/uL

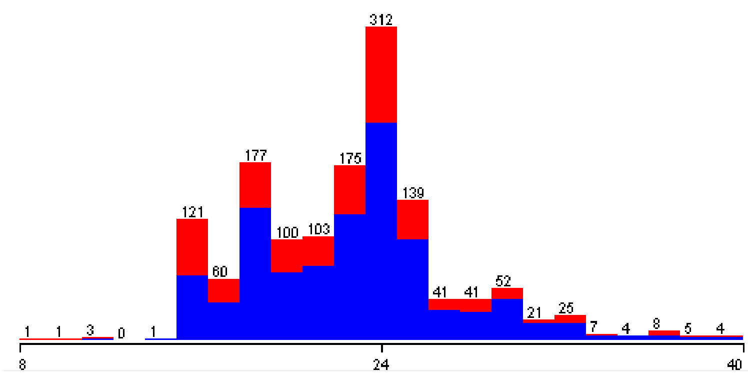
U. Minimum Serum Bicarbonate on Day 1 among Survivors and Non-Survivors



Red:
Mean (Survivors) = 22.1 mEq/L
Standard Deviation (Survivors) = 4.5 mEq/L

Blue:
Mean (Non-Survivors) = 21.8 mEq/L
Standard Deviation (Non-Survivors) = 4.5 mEq/L

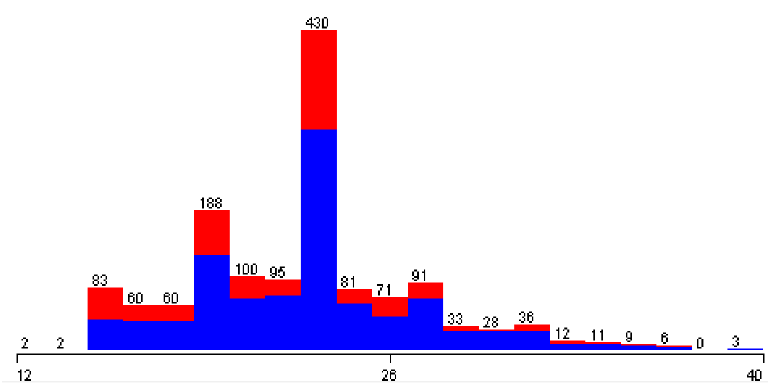
V. Minimum Serum Bicarbonate on Day 2 among Survivors and Non-Survivors



Red:
Mean (Survivors) = 22.8 mEq/L
Standard Deviation (Survivors) = 4.4 mEq/L

Blue:
Mean (Non-Survivors) = 22.0 mEq/L
Standard Deviation (Non-Survivors) = 4.7 mEq/L

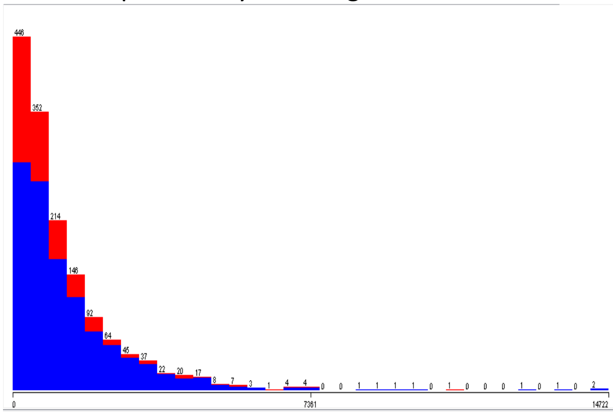
W. Minimum Serum Bicarbonate on Day 3 among Survivors and Non-Survivors



Red:
Mean (Survivors) = 23.3 mEq/L
Standard Deviation (Survivors) = 4.2 mEq/L

Blue:
Mean (Non-Survivors) = 22.4 mEq/L
Standard Deviation (Non-Survivors) = 4.4 mEq/L

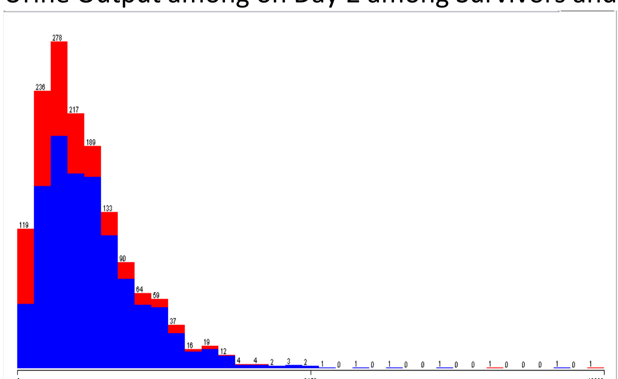
X. Urine Output on Day 1 among Survivors and Non-Survivors



Red:
Mean (Survivors) = 1380.6 ml
Standard Deviation (Survivors) = 1530.4 ml

Blue:
Mean (Non-Survivors) = 919.6 ml
Standard Deviation (Non-Survivors) = 1209.5 ml

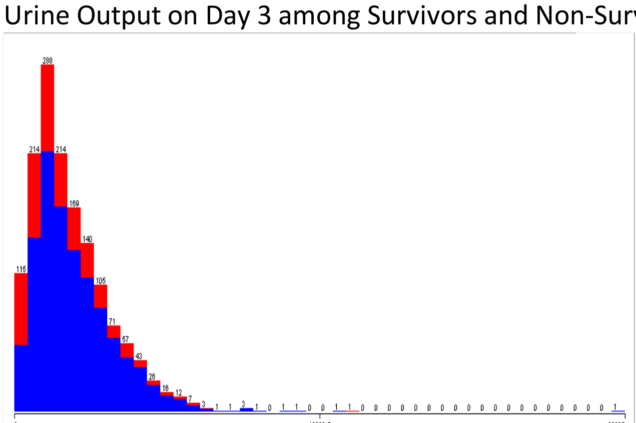
Y. Urine Output among on Day 2 among Survivors and Non-Survivors



Red:
Mean (Survivors) = 2116.9 ml
Standard Deviation (Survivors) = 1474.9 ml

Blue:
Mean (Non-Survivors) = 1525.8 ml
Standard Deviation (Non-Survivors) = 1533.1 ml

Z. Urine Output on Day 3 among Survivors and Non-Survivors



Red:
Mean (Survivors) = 1998.0 ml
Standard Deviation (Survivors) = 1428.6 ml

Blue:
Mean (Non-Survivors) = 1525.0 ml
Standard Deviation (Non-Survivors) = 1292.1 ml

AA. P Values of the Variables using Univariate Logistic Regression

SAPS 3.44e-16	**AGE 0.0122**	SEX 0.822
	MIN_GCS_1ST_DAY 0.00363	
OUT_1ST_DAY 0.000474 **OUT_2ND_DAY 0.000225** **OUT_3RD_DAY 0.000189**	MAX_CREAT_1ST_DAY 0.317 MAX_CREAT_2ND_DAY 0.427 MAX_CREAT_3RD_DAY 0.096	**MAX_BUN_1ST_DAY 0.00195** **MAX_BUN_2ND_DAY 6.4e-07** **MAX_BUN_3RD_DAY 3.27e-10**
MAX_BILI_1ST_DAY 6.49e-05 **MAX_BILI_2ND_DAY 4.53e-05** **MAX_BILI_3RD_DAY 5.35e-05**	MAX_TEMP_1ST_DAY 0.221 MAX_TEMP_2ND_DAY 0.0926 MAX_TEMP_3RD_DAY 0.917	**MAX_RESP_1ST_DAY 0.0126** MAX_RESP_2ND_DAY 0.196 MAX_RESP_3RD_DAY 0.144
MIN_HR_1ST_DAY 0.64496 MAX_HR_1ST_DAY 0.066 STDDEV_HR_1ST_DAY 0.228 AVG_HR_1ST_DAY 0.24688	MIN_HR_2ND_DAY 0.122415 **MAX_HR_2ND_DAY 0.0274** STDDEV_HR_2ND_DAY 0.256 AVG_HR_2ND_DAY 0.0573	MIN_HR_3RD_DAY 0.58534 **MAX_HR_3RD_DAY 0.0131** **STDDEV_HR_3RD_DAY 0.00734** AVG_HR_3RD_DAY 0.0561
MIN_SYSBP_1ST_DAY 8.8e-06 MAX_SYSBP_1ST_DAY 0.00419 **STDDEV_SYSBP_1ST_DAY 0.992** **AVG_SYSBP_1ST_DAY 1.12e-06**	**MIN_SYSBP_2ND_DAY 1.39e-06** MAX_SYSBP_2ND_DAY 0.000287 **STDDEV_SYSBP_2ND_DAY 0.436** **AVG_SYSBP_2ND_DAY 9.81e-07**	**MIN_SYSBP_3RD_DAY 2.89e-09** **MAX_SYSBP_3RD_DAY 0.000264** STDDEV_SYSBP_3RD_DAY 0.503179 **AVG_SYSBP_3RD_DAY 1.07e-07**
MIN_SODIUM_1ST_DAY 0.884 MAX_SODIUM_1ST_DAY 0.546 STDDEV_SODIUM_1ST_DAY 0.166 AVG_SODIUM_1ST_DAY 0.775	MIN_SODIUM_2ND_DAY 0.456 MAX_SODIUM_2ND_DAY 0.699 STDDEV_SODIUM_2ND_DAY 0.353 AVG_SODIUM_2ND_DAY 0.588	MIN_SODIUM_3RD_DAY 0.106 MAX_SODIUM_3RD_DAY 0.283 STDDEV_SODIUM_3RD_DAY 0.093 AVG_SODIUM_3RD_DAY 0.124
MIN_POTASSIUM_1ST_DAY 0.16325 MAX_POTASSIUM_1ST_DAY 0.9065 STDDEV_POTASSIUM_1ST_DAY 0.319 AVG_POTASSIUM_1ST_DAY 0.3212	MIN_POTASSIUM_2ND_DAY 0.1869 **MAX_POTASSIUM_2ND_DAY 0.027899** STDDEV_POTASSIUM_2ND_DAY 0.119 AVG_POTASSIUM_2ND_DAY 0.06747	MIN_POTASSIUM_3RD_DAY 0.4029 **MAX_POTASSIUM_3RD_DAY 0.00269** **STDDEV_POTASSIUM_3RD_DAY 0.00231** **AVG_POTASSIUM_3RD_DAY 0.0392**
MIN_GLUCOSE_1ST_DAY 0.53679 MAX_GLUCOSE_1ST_DAY 0.570680 STDDEV_GLUCOSE_1ST_DAY 0.973 AVG_GLUCOSE_1ST_DAY 0.56901	MIN_GLUCOSE_2ND_DAY 0.608 MAX_GLUCOSE_2ND_DAY 0.154 STDDEV_GLUCOSE_2ND_DAY 0.141 AVG_GLUCOSE_2ND_DAY 0.235	MIN_GLUCOSE_3RD_DAY 0.528 **MAX_GLUCOSE_3RD_DAY 0.0242** **STDDEV_GLUCOSE_3RD_DAY 0.0454** AVG_GLUCOSE_3RD_DAY 0.0781
MIN_BICARBONATE_1ST_DAY 0.134 MAX_BICARBONATE_1ST_DAY 0.066 STDDEV_BICARBONATE_1ST_DAY 0.637 AVG_BICARBONATE_1ST_DAY 0.0691	**MIN_BICARBONATE_2ND_DAY 0.0107** **MAX_BICARBONATE_2ND_DAY 0.0246** STDDEV_BICARBONATE_2ND_DAY 0.427 **AVG_BICARBONATE_2ND_DAY 0.0143**	**MIN_BICARBONATE_3RD_DAY 0.00239** **MAX_BICARBONATE_3RD_DAY 0.0455** **STDDEV_BICARBONATE_3RD_DAY 0.00835** **AVG_BICARBONATE_3RD_DAY 0.0125**
MIN_WBC_1ST_DAY 0.00226 MAX_WBC_1ST_DAY 0.127 STDDEV_WBC_1ST_DAY 0.835 **AVG_WBC_1ST_DAY 0.0282**	**MIN_WBC_2ND_DAY 0.000142** **MAX_WBC_2ND_DAY 3.81e-05** **STDDEV_WBC_2ND_DAY 0.0167** **AVG_WBC_2ND_DAY 6.5e-05**	**MIN_WBC_3RD_DAY 5.77e-05** **MAX_WBC_3RD_DAY 5.83e-06** **STDDEV_WBC_3RD_DAY 0.00217** **AVG_WBC_3RD_DAY 1.64e-05**
MIN_HEMATOCRIT_1ST_DAY 0.170 MAX_HEMATOCRIT_1ST_DAY 0.638 STDDEV_HEMATOCRIT_1ST_DAY 0.110 AVG_HEMATOCRIT_1ST_DAY 0.273	**MIN_HEMATOCRIT_2ND_DAY 0.0223** MAX_HEMATOCRIT_2ND_DAY 0.966 **STDDEV_HEMATOCRIT_2ND_DAY 0.000994** AVG_HEMATOCRIT_2ND_DAY 0.239	**MIN_HEMATOCRIT_3RD_DAY 0.0245** MAX_HEMATOCRIT_3RD_DAY 0.935 **STDDEV_HEMATOCRIT_3RD_DAY 0.000376** AVG_HEMATOCRIT_3RD_DAY 0.22

BB. SAPS vs. Second Level Customization

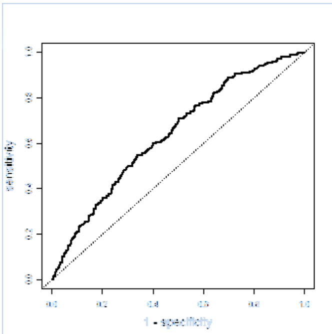
The SAPS physiologic variables are as follows:

SAPS Variables	
AGE	MAX_BUN_1ST_DAY
MIN_GCS_1ST_DAY	MIN_WBC_1ST_DAY
MIN_SYSBP_1ST_DAY	MIN_POTASSIUM_1ST_DAY
MAX_HR_1ST_DAY	MAX_SODIUM_1ST_DAY
MAX_TEMP_1ST_DAY	MIN_BICARBONATE_1ST_DAY
OUTPUT_1ST_DAY	MAX_BILIRUBIN_1ST_DAY

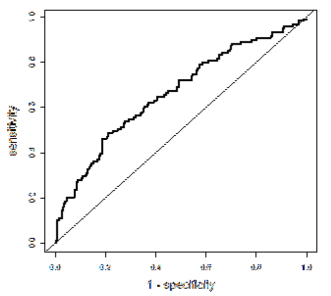
The table below shows the AUC and Hosmer-Lemeshow p value **on the test data** of SAPS and the best fitted multivariate model of the SAPS physiologic variables using the training data.

Model	Area under the ROC Curve	Hosmer-Lemeshow P Value
SAPS Predicted Death Rate = $e^{(Logit)}/(1+e^{(Logit)})$ Logit = -7.7631+0.0737*SAPS+0.9971*ln(SAPS+1)	0.6419	0
Second Level Customization (Multivariate Regression using SAPS Variables)	0.6639	0.2056

The ROC Curves are shown below.



SAPS



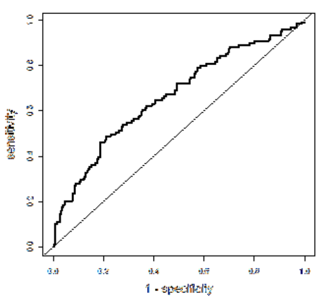
Second Level Customization

CC. SAPS Physiologic Variables on Day 1 vs. Day 2 vs. Day 3

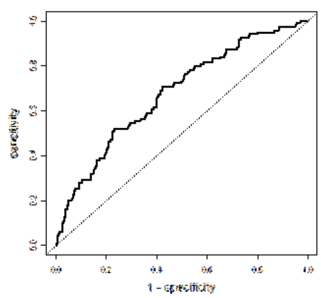
The table below shows the AUC and Hosmer-Lemeshow p value **on the test data** of the best fitted logistic regression models of the SAPS physiologic variables from Day 1, Day 2 and Day 3 using the training data.

Model	Area Under the ROC Curve	Hosmer-Lemeshow p Value
Day 1 SAPS Variables	0.6639	0.2056
Day 2 SAPS Variables	0.6678	0.1332
Day 3 SAPS Variables	0.7030	0.5208

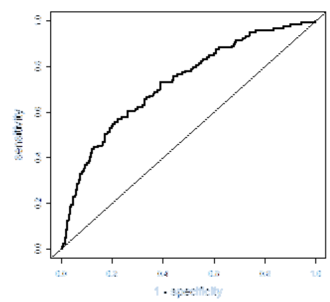
The ROC Curves are shown below.



Day 1 SAPS Variables



Day 2 SAPS Variables



Day 3 Variables

The p values of the variables in the best fitted logistic regression model using the SAPS physiologic variables from Day 1, Day 2 and Day 3 are shown below. The variables whose p values are less than 0.05 are boxed.

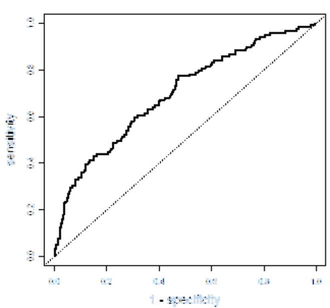
Day 1		Day 2		Day 3	
AGE	0.0032	AGE	0.0142	AGE	0.0109
MIN_SYSBP_1ST_DAY	0.0057	MIN_SYSBP_2ND_DAY	0.0002	MIN_SYSBP_3RD_DAY	1.12e-06
MAX_HR_1ST_DAY	0.0548	MAX_HR_2ND_DAY	0.0499	MAX_HR_3RD_DAY	0.0731
MAX_TEMP_1ST_DAY	0.2178	MAX_TEMP_2ND_DAY	0.6265	MAX_TEMP_3RD_DAY	0.1990
OUT_1ST_DAY	0.0178	OUT_2ND_DAY	0.0032	OUT_3RD_DAY	0.0032
MAX_BUN_1ST_DAY	0.0327	MAX_BUN_2ND_DAY	0.0010	MAX_BUN_3RD_DAY	7.22e-07
MIN_WBC_1ST_DAY	0.0093	MIN_WBC_2ND_DAY	0.0100	MIN_WBC_3RD_DAY	0.0102
MIN_POTSM_1ST_DAY	0.5553	MIN_POTSM_2ND_DAY	0.7154	MIN_POTSM_3RD_DAY	0.8232
MAX_SODM_1ST_DAY	0.2490	MAX_SODM_2ND_DAY	0.4760	MAX_SODM_3RD_DAY	0.9921
MIN_BICARB_1ST_DAY	0.6979	MIN_BICARB_2ND_DAY	0.6440	MIN_BICARB_3RD_DAY	0.5889
MAX_BILI_1ST_DAY	3.76e-05	MAX_BILI_2ND_DAY	7.94e-05	MAX_BILI_3RD_DAY	0.0001
MIN_GCS_1ST_DAY	0.0074				

P Values of the Variables in the Best Fitted Models

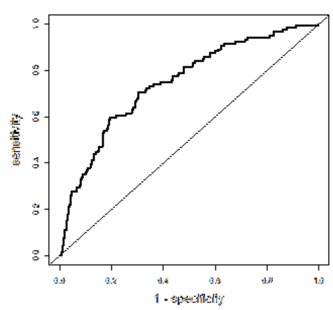
DD. Logistic Regression Models using Combination of Day 1, Day 2 and Day 3 SAPS Physiologic Variables

Model	Area Under the ROC Curve	Hosmer-Lemeshow p Value
Day 3 SAPS Variables	0.7030	0.5208
Day 1 SAPS Variables + Day 2 SAPS Variables	0.6964	0.5032
Day 2 SAPS Variables + Day 3 SAPS Variables	0.7454	0.4640
Day 1 SAPS Variables + Day 3 SAPS Variables	0.7352	0.3778
Day 1 SAPS Variables + Day 2 SAPS Variables Day 3 SAPS Variables	0.7419	0.6738

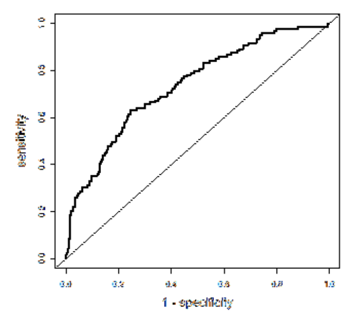
The ROC Curves are shown below.



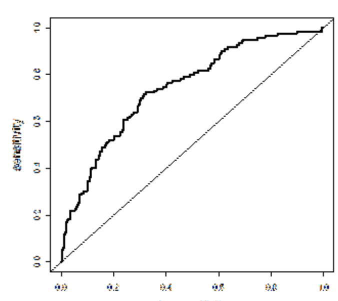
Day 1 and 2 SAPS Variables



Day 2 and 3 SAPS Variables



Day 1 and 3 SAPS Variables



Day 1, 2 and 3 SAPS Variables

The p values of the variables in the best fitted logistic regression model using the SAPS physiologic variables from Day 1, Day 2 and Day 3 are shown below. The variables whose p values are less than 0.05 are boxed.

	P Value		P Value		P Value
MIN_SYSBP_1ST_DAY	0.442367	MIN_SYSBP_2ND_DAY	0.895948	MIN_SYSBP_3RD_DAY	0.000705
MAX_HR_1ST_DAY	0.403927	MAX_HR_2ND_DAY	0.480650	MAX_HR_3RD_DAY	0.587860
MAX_TEMP_1ST_DAY	0.709826	MAX_TEMP_2ND_DAY	0.080553	MAX_TEMP_3RD_DAY	0.083751
OUTPUT_1ST_DAY	0.079730	OUTPUT_2ND_DAY	0.208689	OUTPUT_3RD_DAY	0.280979
MAX_BUN_1ST_DAY	0.225744	MAX_BUN_2ND_DAY	0.780805	MAX_BUN_3RD_DAY	0.005804
MIN_WBC_1ST_DAY	0.821370	MIN_WBC_2ND_DAY	0.677556	MIN_WBC_3RD_DAY	0.109671
MIN_POTSM_1ST_DAY	0.597589	MIN_POTSM_2ND_DAY	0.835240	MIN_POTSM_3RD_DAY	0.979239
MAX_SODIUM_1ST_DAY	0.380977	MAX_SODIUM_2ND_DAY	0.051881	MAX_SODIUM_3RD_DAY	0.119049
MIN_BICARB_1ST_DAY	0.579316	MIN_BICARB_2ND_DAY	0.927018	MIN_BICARB_3RD_DAY	0.813154
MAX_BILI_1ST_DAY	0.063459	MAX_BILI_2ND_DAY	0.184281	MAX_BILI_3RD_DAY	0.985616
MIN_GCS_1ST_DAY	0.009026			AGE	0.032383

EE. Comparison of Models using Variables selected by CFS and Consistency Subset Evaluator Algorithms

The table below lists the variables selected by the two feature selection algorithms. Except for a few exceptions (boxed), the two algorithms came up with identical variables.

Variables Selected by CFS Subset Evaluator Algorithm			Variables Selected by Consistency Subset Evaluator Algorithm		
AGE	AVG_SYSBP_ 1ST_DAY	STDDEV_HCT_ 2ND_DAY	AGE	MIN_SYSBP_ 1ST_DAY	MAX_WBC_ 1ST_DAY
MIN_GCS_ 1ST_DAY	MIN_SYSBP_ 2ND_DAY	MIN_WBC_ 1ST_DAY	MIN_GCS_ 1ST_DAY	AVG_SYSBP_ 1ST_DAY	AVG_WBC_ 2ND_DAY
MIN_HR_ 2ND_DAY	MIN_SYSBP_ 3RD_DAY	MIN_WBC_ 2ND_DAY	MAX_RESP_ 3RD_DAY	MIN_SYSBP_ 2ND_DAY	MIN_WBC_ 3RD_DAY
MAX_RESP_ 3RD_DAY	AVG_SYSBP_ 3RD_DAY	AVG_WBC_ 2ND_DAY	MAX_BUN_ 2ND_DAY	AVG_SYSBP_ 2ND_DAY	STDDEV_WBC_ 3RD_DAY
OUT_1ST_DAY	MAX_BUN_ 2ND_DAY	MIN_WBC_ 3RD_DAY	MAX_BUN_ 3RD_DAY	MIN_SYSBP_ 3RD_DAY	STDDEV_HCT_ 2ND_DAY
OUT_2ND_DAY	MAX_BUN_ 3RD_DAY	MAX_BILI_ 2ND_DAY	OUT_1ST_DAY	AVG_SYSBP_ 3RD_DAY	STDDEV_HCT_ 3RD_DAY
OUT_3RD_DAY	MAX_BICARB_ 1ST_DAY	MAX_BILI_ 3RD_DAY	OUT_2ND_DAY	MAX_BICARB _1ST_DAY	MAX_BILI_ 2ND_DAY
			OUT_3RD_DAY		MAX_BILI_ 3RD_DAY

The AUC and HL p value of the best fitted logistic regression models using combination of Day 1, Day 2 and Day 3 SAPS physiologic variables, and the variables selected by CFS and Consistency Subset Evaluator algorithms are tabulated below.

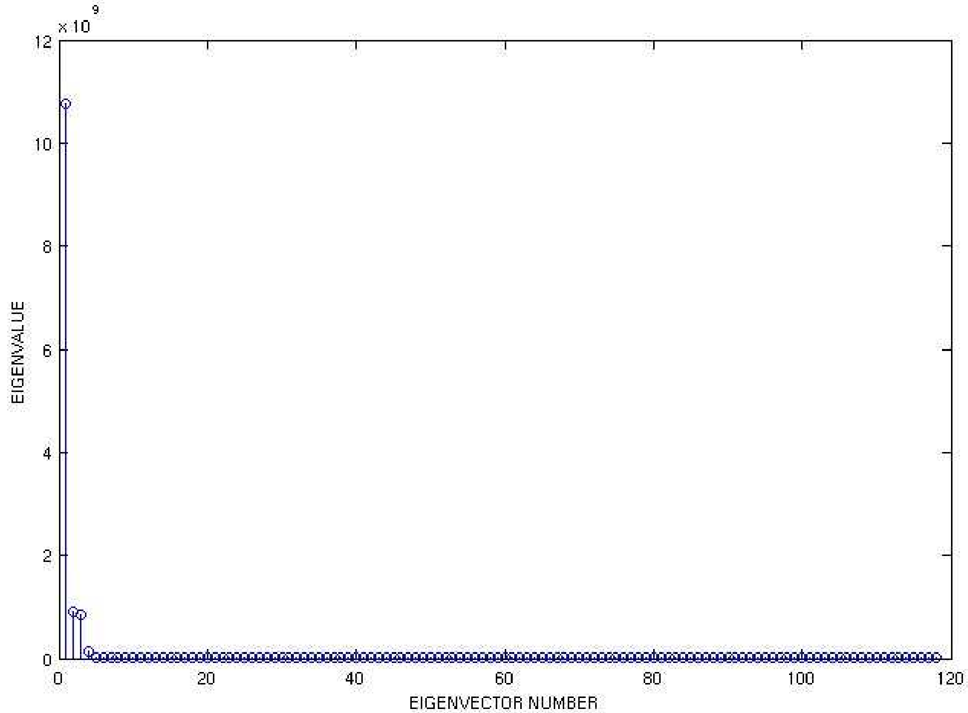
Model	Area Under the ROC Curve	Hosmer-Lemeshow p Value
Day 1 SAPS Variables + Day 2 SAPS Variables + Day 3 SAPS Variables	0.7419	0.6738
Variables selected by CFS Subset Evaluation Algorithm	0.7332	0.5945
Variables selected by Consistency Subset Evaluation Algorithm	0.7289	0.6279

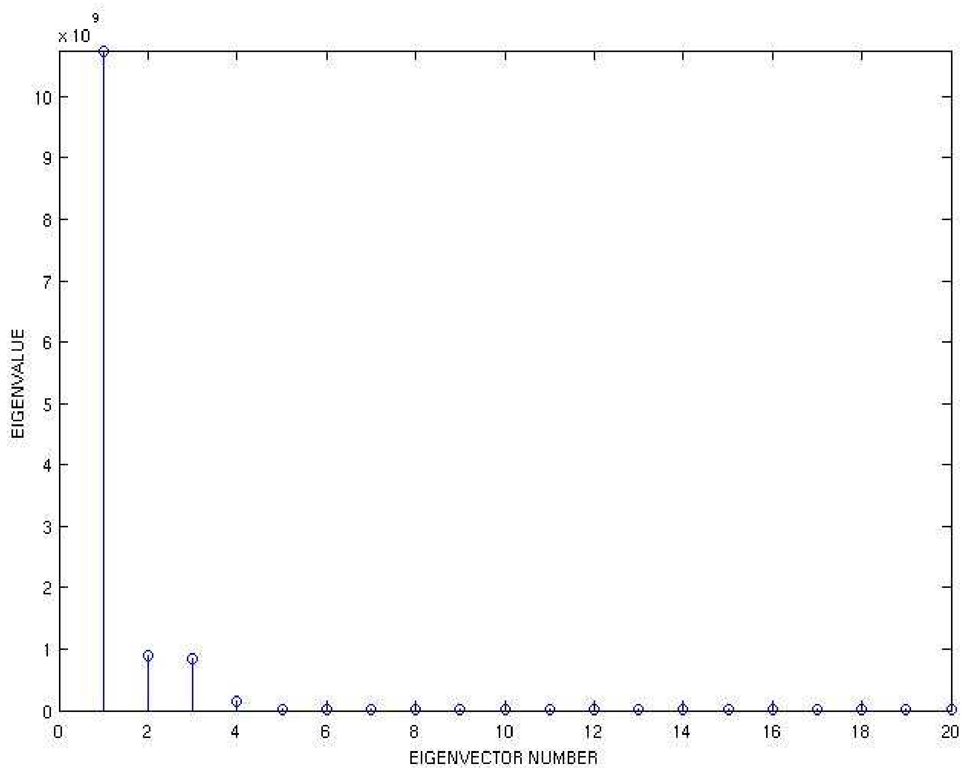
FF. Effect of Interaction Terms on the Performance of the Best Fitted Logistic Regression Model

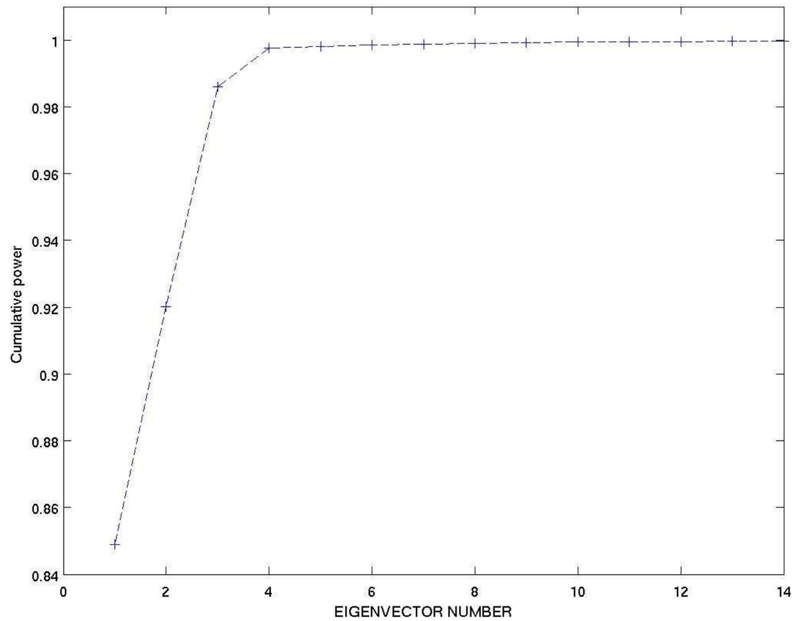
The effect of the addition of heuristically selected interaction terms on the performance of the best fitted logistic regression model using combination of Day 1, Day 2 and Day 3 SAPS physiologic variables is shown below.

Model	Area Under the ROC Curve	Hosmer-Lemeshow p Value
Best Fitted Model (Day 1 SAPS Variables + Day 2 SAPS Variables + Day 3 SAPS Variables)	0.7419	0.6738
Best Fitted Model with Blood Pressure*Heart rate	0.7421	0.4558
Best Fitted Model with Age*Serum Creatinine	0.7239	0.1296
Best Fitted Model with BUN*Serum Creatinine	0.7440	0.0624
Best Fitted Model with Sex*Serum Creatinine	0.7310	0.1745
Best Fitted Model with Glucose Standard Deviation*GCS	0.7384	0.3866

GG. Principal Component Analysis of the Variables







HH. Best Fitted Logistic Regression Model using Filtered Data

	Original Data AUC	Filtered Data AUC
Day 1 SAPS + Day 3 SAPS	0.735	0.610
Day 2 SAPS + Day 3 SAPS	0.745	0.607
Day 1 SAPS + Day 2 SAPS + Day 3 SAPS	0.742	0.612

II. Best Fitted Logistic Regression Models using Different Machine Learning Algorithms

Below is the performance of five machine-learning algorithms in predicting hospital mortality among patients with AKI using combination of Day 1, Day 2 and Day 3 SAPS physiologic variables.

	Accuracy	Mean Absolute Error	Area under the ROC Curve
Logistic Regression	70.68%	0.3796	0.685
Bayes Net	68.13%	0.3587	0.682
Naïve Bayes	71.91%	0.2906	0.691
Classification and Regression Tree	67.52%	0.4154	0.562
Artificial Neural Network	65.58%	0.3428	0.633

Below is the performance of five machine-learning algorithms in predicting hospital mortality among patients with AKI using variables selected by the CFS Subset Evaluator algorithm.

	Accuracy	Mean Absolute Error	Area under the ROC Curve
Logistic Regression	71.20%	0.3762	0.704
Bayes Net	68.85%	0.3490	0.687
Naïve Bayes	71.20%	0.2910	0.698
Classification and Regression Tree	68.03%	0.3980	0.619
Artificial Neural Network	64.45%	0.3615	0.645

Below is the performance of five machine-learning algorithms in predicting hospital mortality among patients with AKI using variables selected by the Consistency Subset Evaluator algorithm.

	Accuracy	Mean Absolute Error	Area under the ROC Curve
Logistic Regression	71.20%	0.3726	0.706
Bayes Net	67.93%	0.3497	0.689
Naïve Bayes	70.07%	0.2997	0.692
Classification and Regression Tree	67.31%	0.4129	0.583
Artificial Neural Network	66.19%	0.3438	0.651

Discussion

The physiologic response of a patient to stress or disease is the main determinant of the outcome [20]. Bion [21] suggests this response is dependent on three factors; severity of the acute insult, the treatment given, and the patient's degree of physiological reserve.

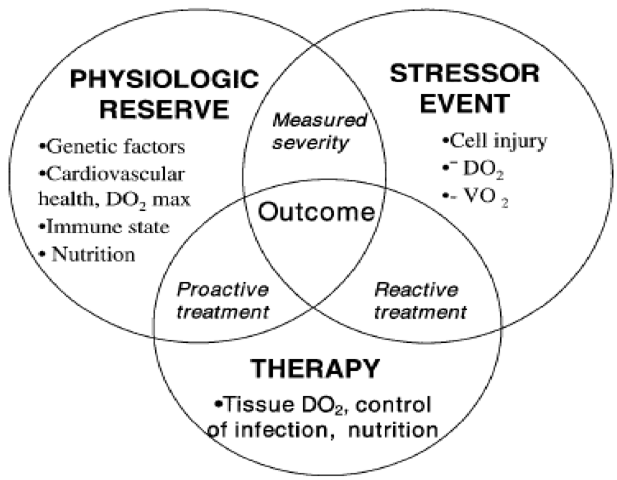


Fig. 1

Physiological Response. Bion (2000)

Severity scoring systems capture the stressor event. Over the last decade, there has been a push to measure quality of therapy, but this has not been incorporated into mortality prediction. Of the three factors, the physiologic reserve is the least characterized.

Physiologic reserve accounts for the difference in clinical outcome that two patients with identical mortality risks and treatment may have. It is defined as the body's response to stress and disease at a cellular level and the variation between individuals is thought to be largely influenced by genetic factors. This reserve dictates how the patient responds to and heals from the acute insult, such as sepsis, trauma, burns or major surgery, regardless of the treatments provided in the ICU, and may be the most significant of the three factors. There is currently no biomarker for physiologic reserve, but age, cardio-pulmonary reserve, immune and nutritional status have been used in various combinations as a surrogate marker.

At present, only the patients at extremely high risk are able to have their outcome predicted with high specificity (i.e. a low number of false negative predictions), and there is still a relatively low sensitivity to pick up patients that will die despite having a less than extremely high risk of death.

The 31% mortality of this subgroup of patients with AKI is higher than the 12% mortality of patients in the MIMIC II database [22], and is consistent with published literature. Chertow and colleagues previously showed that even small changes in serum creatinine while in the hospital were independently associated with an increased risk of dying [23].

Distribution of Variables among Survivors and Non-Survivors

The non-survivors had a higher SAPS and were older than the patients who survived to hospital discharge. Among this subset of ICU patients who developed AKI, there was no significant difference between those who survived and those who died as regards the initial and subsequent serum creatinine levels during the first 72 hours in the ICU. This suggests that although AKI contributes to hospital mortality, the initial degree of renal dysfunction, as measured by the serum creatinine, does not exert a significant influence on whether the patient recovers or not, as reflected by hospital mortality. This is supported further by the finding that there was likewise no significant difference in the serum bicarbonate levels between the survivors and non-survivors over the first three ICU days. The serum bicarbonate is measured primarily to detect the presence of metabolic acidosis. Renal dysfunction is one of the most common etiologies of metabolic acidosis in the ICU.

The minimum GCS score on the first ICU day, a surrogate for the worst level of consciousness at the time of presentation in the ICU, is lower among the non-survivors. The patients who died had a higher maximum heart rate and a lower minimum systolic blood pressure for each of the first three days in the ICU as compared to those who survived to hospital discharge.

The hematocrit did not significantly differ between the survivors and non-survivors during the first 72 hours in the ICU. The WBC, however, was higher among the non-survivors, with the gap widening from the first to the third ICU day, suggesting a more intense initial inflammatory response among the non-survivors.

The patients who died had a lower urine output than those who survived on each of the first three days in the ICU.

Finally, with a few exceptions (systolic blood pressure, serum bicarbonate), for this group of patients who survived more than 72 hours in the ICU, the variance (as measured by the standard deviation) of the physiologic variables decreased over the first three days in the ICU, whether or not the patient survived to hospital discharge. This likely reflects interventions to correct abnormal physiologic parameters as is customary in the ICU in an attempt to influence clinical outcomes, e.g. transfusion to correct anemia, potassium replacement, beta-blockade for tachycardia.

Mortality Prediction by SAPS and Second Level Customization

The first question we wanted to address is whether customization using local institutional data on patients with AKI will perform better compared to SAPS in predicting mortality. The accuracy of mortality prediction models is assessed based on discrimination between survivors and non-survivors and on correspondence between observed and predicted mortality across the entire range of risk (calibration). We reported the AUC and the Hosmer-Lemeshow p value as measures of discrimination and calibration, respectively, for all the logistic regression models. Given that we had a total of 118 variables we extracted from each of the patients in the cohort, an exhaustive search for the combination of variables that would yield the most accurate model in terms of mortality prediction was out of the question. To get some idea whether a variable is correlated with mortality, we performed univariate regression analysis for each the 118 variables. However, we did not select variables based solely on the p value in the univariate analysis. We know that variables with significant p values on univariate analysis may lose their significance once it is adjusted for other variables in a multivariate analysis. We also know that although variables may not be significantly correlated with the outcome of interest in a multivariate model (as reflected by the p value), they may still contribute to the accuracy of the entire model.

Based on the p value on univariate analysis, SAPS ($p = 3.44 \times 10^{-16}$) is the variable that is most correlated with hospital mortality, followed by the maximum blood urea nitrogen ($p = 3.27 \times 10^{-10}$) and the minimum systolic blood pressure from the third day ($p = 2.89 \times 10^{-09}$). Most variables had increasing significance from the first to the third day based on progressively lower p values. This is most evident with the urine output, maximum heart rate, systolic blood pressure measurements, maximum serum potassium, maximum BUN, minimum serum bicarbonate, serum glucose measurements, and white blood count measurements. Based on univariate analysis, the third ICU day had the most variables correlated with hospital mortality. The exception is the maximum respiratory rate. This is the only variable that reached significance during the first, but not on the second and third day. This is easily explained by the fact that the sickest patients tend to be mechanically ventilated by the second and third day.

Among patients who developed acute renal failure, it appears that the serum creatinine is NOT a determinant of outcome in terms of survival. Neither serum sodium nor glucose also appears to be correlated with mortality among this subset of patients.

The AUC and Hosmer-Lemeshow p value of SAPS among the MIMIC II patients with AKI that we obtained (AUC = 0.6419, p = 0) are consistent with the performance of SAPS in predicting mortality among ICU patients in the US (AUC = 0.67, p = 0.05) [5]. In a UK study cited earlier, APACHE II, another severity scoring system looking at physiologic variables during the first 24 hours in the ICU, also had poor calibration (Hosmer-Lemeshow $p < 0.001$) when used to predict death among patients with AKI [3]. This poor performance of current predictive models when applied to (1) regions different from where the model was built and (2) specific subsets of ICU patients is the main impetus for this research.

Second level customization involved performing a multivariate logistic regression using the SAPS physiologic variables. There was improvement in both discrimination and calibration as evaluated by the AUC and Hosmer-Lemeshow p value, respectively. The improvement in the AUC is nicely depicted in the ROC curves.

Third Level Customization: Choosing Variables using Heuristics

We chose to evaluate physiologic variables extending past 24 hours and up to 72 hours of ICU admission based on studies suggesting that information from the third ICU day improves mortality prediction. Research conducted by Girou and colleagues [24] demonstrated that whereas severity scores on admission failed to predict mortality, APACHE II and SAPS on the third ICU day of the patients who died were significantly higher than those of the survivors. In another study from Mayo Clinic [25], among the sickest patients admitted to the ICU, only 6% of patients whose APACHE III scores on the third ICU day were higher than the admission scores survived to hospital discharge, as compared to 43% of patients whose third day APACHE II scores were lower or remained the same.

We then developed logistic regression models using the day 1, day 2 and day 3 SAPS physiologic variables separately. Consistent with the findings from the univariate analysis, the model using Day 3 SAPS variables had the best AUC and Hosmer-Lemeshow p value, supporting the hypothesis that the physiologic status of the patient with AKI on the third ICU day is most predictive of the clinical outcome. This finding is also consistent with the conclusions of the Girou study cited above. The same SAPS physiologic variables were significantly correlated with hospital mortality from each of the three days: minimum systolic blood pressure, urine output, maximum BUN, maximum WBC and maximum serum bilirubin. Of these, as in the univariate analyses, the maximum blood urea nitrogen and the minimum systolic blood pressure from the third ICU day had the lowest p value.

We proceeded with combining SAPS physiologic variables from two (Day 1 + Day 2, Day 2 + Day 3, Day 1 + day 3) and then all three days in order to see whether this strategy, which reflects the evolution of the patient's physiologic status, might improve the accuracy of mortality prediction. There is improvement in the AUC as we added SAPS physiologic variables from either Day 1 or Day 2 to Day 3 variables. But the model that performed the best when both discrimination and calibration are assessed is the one that incorporated SAPS physiologic variables from all three days. It is quite interesting to note that when multivariate analysis was performed on SAPS physiologic variables from the first 72 hours, there were only four variables that were significantly correlated with the hospital mortality: the age, the Glasgow Coma Score on the 1st day, and once again, the maximum BUN and the minimum systolic blood pressure from the 3rd ICU day.

Third Level Customization: Choosing Variables using Algorithms

We employed two algorithms - correlation-based feature subset selection (CFS) and the consistency subset evaluation - to choose variables for logistic regression. The variables selected by the two algorithms are almost identical, and are quite similar to the SAPS variables, which were originally identified by a panel of experts. We then compared the AUC and Hosmer-Lemeshow p value of the resulting models with those of the model using a combination of the Day 1, Day 2 and Day 3 SAPS physiologic variables. The performance of the three models did not differ significantly; they all have good discrimination and calibration. The feature selection algorithms matched the heuristics of expert clinicians in identifying variables that predict patient outcomes.

Evaluation of Interaction Terms in Logistic Regression

There are a number of ICU mortality studies suggesting the presence of effect modification. For example, liver failure increased the mortality of cirrhotic patients with AKI but had no effect on cirrhotic patients without AKI [26]. We explored possible effect modification and evaluated a number of interaction terms that were selected based on clinical knowledge.

None of the interaction terms improved the discrimination of the best fitted logistic regression model using Day 1, Day 2 and Day 3 SAPS physiologic variables. The models that included the interaction terms had lower Hosmer-Lemeshow p values, suggesting poorer calibration. Possible explanations for our finding include selection of the wrong interaction terms, effect modification that may not be constant over time, and effect modification involving three or more variables.

Filtering using Principal Component Analysis

We attempted to reduce the noise by filtering. Principal component analysis (PCA) was performed on the training set to filter these data using the first four largest eigenvectors, accounting for 99.75% of the variability. However, when logistic regression analysis was performed on the filtered data, the resulting AUC was much lower than that of the same model performed on the original data.

The goal of PCA is to seek new variables, which are orthogonal linear combinations of the old parameters, to better explain the variation in the data set. Since each new component is found by looking for the projection of maximum variance in the data, the smallest Eigenvalues are often assumed to be due to noise. We performed filtering by collapsing the Eigenvectors down and re-projecting the data back into the original space. However, if the signal-to-noise ratio of the data is low and the noise dominates, then the larger Eigenvalues may correspond to noise, and we might have inadvertently filtered out data, and retained noise.

Another reason PCA may be a poor method for separating noise from signal is that PCA assumes that the underlying variables are Gaussian. Non-Gaussian variables will therefore not be well captured by this method.

Performance of Different Machine Learning Algorithms

Finally, we investigated how other machine learning algorithms perform against logistic regression using the variables from the best fitted regression models, i.e. those using the SAPS physiologic variables from the first 72 hours and the variables chosen by the feature selection algorithms. Naïve Bayes performed as well as logistic regression based on accuracy, mean absolute error and area under the ROC curve. It performed better than Bayesian Network in all three models. This is a bit surprising given that the variables we evaluated are not independent. However, Bayesian Network may overfit more than Naïve Bayes which might explain its poorer performance.

Limitations of the Study

Despite a significant improvement in the AUC and Hosmer-Lemeshow p value with third level customization using the SAPS physiologic variables from the first 72 hours in the ICU, a much higher AUC would have been more convincing evidence to support our hypothesis. There are several reasons why a higher AUC was not seen with the logistic regression models that were presented. The first is data noise. During pre-processing, some of the variables that were initially considered had to be dropped because inspection of the values revealed a significant fraction of the data was inaccurately captured. We decided to exclude total fluid input after finding that 30% of patients were administered less than a liter of fluid for the entire day. From clinical experience, patients who are admitted to the ICU are given at least a liter of fluid to replace insensible losses (which are higher in a critically-ill patient) even in the presence of acute kidney injury. We also removed minimum temperature after finding that about a third of the patients had temperature below normal. We suspect that this is likely a result of a displaced rectal probe capturing inaccurate data. Finally, minimum respiratory rate was excluded with respiratory rates below 8. This is unusual in the ICU even when the patient is mechanically ventilated. We suspect that these measurements were taken when the chest wall motion detector was dislodged which happens often (and for the most part ignored by ICU nurses). We can only speculate how much inaccuracy is present with the data that we ended up using to develop the models. We see data noise as the main disadvantage in developing classification and regression algorithms using retrospective data. The data that were used to build the original models that are the basis of existing severity scoring systems were entered manually from paper records, and were more likely verified at the time of entry by reviewing the progress notes if there is inaccuracy suspected.

Another possible reason why it is difficult to develop a model based on retrospective clinical data with excellent discrimination and calibration is because quality of care, an important determinant of clinical outcome, requires more meticulous data extraction. To illustrate, two hypovolemic patients who have the same severity of illness might both have gotten 2 liters of crystalloid solution. This level of information is captured in the database and is routinely extracted for clinical studies. However, one patient might have received it over 30 minutes, while the other was given this amount over 2 hours. This information might seem trivial but may be the reason why one patient develops an acute kidney injury while the other does not. Another illustration would be two identical patients who develop an acute abdomen requiring a surgical intervention. The extracted data accurately documents that both patients went to surgery. However, one patient might have taken an hour longer before surgery was started, or one patient's surgery took longer because it was a senior surgical resident who performed the operation. The last example to drive the point would be two similar patients presenting with the same infection of identical severity. They are administered the same antibiotic but one patient received it sooner than the other. The interval between the times of administration might be enough to lead to different outcomes.

But this is not to say that none of this information is present in a data set similar to the one that was used for this project. In the first example, the patient who received the fluid over a longer period of time will eventually have a rise in serum creatinine, which will predict a worse clinical outcome. In the second illustration, the patient where surgical intervention was delayed or took longer may develop complications that will be reflected by a different course of blood pressure measurements. And in the last example, the patient who did not get the antibiotics in a more timely fashion may take more days to defervesce. The question is how much detail is required to be incorporated in a data set to adequately capture the contribution of quality of care to clinical outcome.

Clinical Application

The optimal mortality prediction model should capture the heuristics employed by expert clinicians as they look at the evolution of physiologic variables over time to assess whether a patient is responding favorably to treatment and is likely to have a good clinical outcome. There is a large variation among ICU clinicians in terms of when end-of-life discussions are initiated among patients who do not survive their hospitalization [27, 28]. Experienced doctors and nurses are more likely to confidently predict a poor outcome and recommend switching to comfort measures sooner than novice clinicians. What we are aiming to build is a patient-subset specific model that captures all the determinants of the patient's clinical outcome - the severity of illness, how the patient is handling the physiologic insult on his own, how well he is responding to the treatments being administered, and the quality of the care he is provided.

The gold standard in evidence-based medicine is a well-designed, well-executed multi-center prospective randomized controlled trial. Even when such trials are performed and subsequently published, they very rarely, if ever, provide clear evidence upon which to base the management of an individual patient. Patient prognostication is no exception. There is an abundance of literature on risk assessment performed prospectively. However, patients enrolled in prospective randomized controlled trials are heterogeneous, and conclusions are valid for the "average" patient. In addition, these trials are executed in very strictly monitored, and thus artificial, conditions, and often, findings in these studies do not translate to the real world ICU. It is difficult to predict whether an individual patient is likely to behave like the "average" patient in the multi-center prospective randomized trial. Hence, day-to-day clinical decisions are still based mostly on personal experience, experiences shared with colleagues, and consideration of reported data if they exist.

Data mining may provide an additional tool for decision support [17]. The main objective of this project is to determine whether customization of predictive models for specific subgroup of patients yields more accurate predictions. As more ICUs switch to a paperless system, large regional or even local ICU database become available for building models. Rather than developing models with good external validity by including a heterogeneous patient population from various continents as has been traditionally done, an alternative approach would be to build models for specific patient subsets using one's own local or regional database.

We suspect that we will not be able to capture fully the heuristics of an experienced clinician. For that reason, we will not be able to build a model that can replace years of ICU experience. We see the value of decision support systems, including predictive models, based on data mining of empiric data in assisting novice attending physicians, fellows and residents in the ICU. Learning the right amount of fluid to administer, for example, or developing the intuition of whether a patient will benefit from a specific intervention or not, usually requires years of practice. By being able to extract information about similar patients from a database, specifically how they responded to certain treatments, learning may be accelerated and may allow junior doctors to make decisions with more precision and confidence.

All machine learning algorithms assume that real world situations are similar to the training data. This exposes them to the problem of induction in logic. The classic example is the Black Swan phenomenon which argues that rare events are more common than has been traditionally assumed [29]. Nassim Taleb, who popularized the phenomenon, divides real world events into Mediocristan, which fit the bell curve model, and Extremistan, which don't. He suggests that most real-world phenomena actually inhabit Extramistan rather than Mediocristan. We suspect these phenomena co-exist. Nevertheless, this is the reason why it is crucial that we prospectively evaluate whether the use of data mining to assist clinical decision making will lead to better clinical outcomes.

But even if we come up with a mortality prediction model based on local institutional empiric data that has excellent sensitivity and specificity, the question remains whether clinicians will embrace this approach. Will we be able to convince them that information from a very large cohort of patients whose clinical course is stored in an electronic database might be more reliable than a composite of the patients they have encountered in the past whose clinical course may be imperfectly stored in their memory? Will we be able to convince them that their clinical intuition on an individual patient might be enhanced by the experience of numerous clinicians who have taken care of clinically similar patients? Qualitative studies addressing these issues will be the subject of my thesis project at the Harvard School of Public Health.

References:

1. Strand K, Flaaten H. Severity scoring in the ICU: a review. *Acta Anaesthesiol Scand* 2008; 52: 467-478.

2. Uchino S, Kellum J, Bellomo R, *et al*. Acute renal failure in critically ill patients: A multinational, multicenter study. *JAMA* 2005; 294: 813-818.
3. Kolhe N, Stevens P, Crowe A, *et al*. Case mix, outcome and activity for patients with severe acute kidney injury during the first 24 hours after admission to an adult general critical care unit: application of predictive models from a secondary analysis of the ICNARC Case Mix Programme Database. *Crit Care* 2008; 12:S2.
4. 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.
5. Patel P, Grant B. Application of mortality prediction systems to individual intensive care units. *Intensive Care Med* 1999; 25: 977-82.
6. Knaus W, Draper E, Wagner D, *et al*. APACHE II: a severity of disease classification system. *Crit Care Med* 1985; 13: 818-29.
7. Zimmerman J, Kramer A, McNair D, *et al*. Acute Physiology and Chronic Health Evaluation (APACHE) IV: hospital mortality assessment for today's critically ill patients. *Crit Care Med* 2006; 34: 1297-310.
8. Marshall J, Cook D, Christou N, et al. Multiple organ dysfunction score: a reliable descriptor of a complex clinical outcome. *Crit Care Med* 1995; 23: 1638-52.
9. Le Gall J, Neumann A, Hemery F, et al. Mortality prediction using SAPS II: an update for French intensive care units. *Crit Care* 2005; 9: R645-52.
10. Le Gall, Lemeshow S, Leleu G, *et al*. Customized probability models for early severe sepsis in adult intensive care patients. Intensive Care Unit Scoring Group. *JAMA* 1995; 273: 644-50.
11. Rue M, Quintana S, Alvarez M, et al. Daily assessment of severity of illness and mortality prediction for individual patients. *Crit Care Med* 2001; 29: 45-50.
12. Wagner DP, Knaus WA, Harrell FE, et al. Daily prognostic estimates for critically ill adults in intensive care units: Results from a prospective, multicenter, inception cohort analysis. Crit Care Med 1994; 22: 1359-72.
13. Timsit JF, Fosse JP, Troche G, et al. Calibration and discrimination by daily Logistic Organ Dysfunction scoring comparatively with Sequential Organ Failure Assessment scoring for predicting hospital mortality in critically ill patients. Crit Care Med 2002; 30: 2003-13.
14. Cook R, Cook D, Tilley J, et al. Multiple organ dysfunction: Baseline and serial component scores. Crit Care Med 2001; 29: 2046-50.
15. Ympa YP, Sakr Y, Reinhart K, Vincent JL. Has mortality from acute renal failure decreased? A systematic review of the literature. *Am J Med* 2005; 118: 827-32.
16. Eachempati SR, Wang JC, Hydo LJ, Shou J, Barie PS. Acute renal failure in critically ill surgical patients: persistent lethality despite new modes of renal replacement therapy. *J Trauma* 2007; 63: 987-93.
17. Celi LA, Hinske LC, Alterovitz G, Szolovits P. Artificial intelligence to predict fluid requirement in the ICU: a proof-of-concept study. *Crit Care* 2008: 12(6):R151.
18. Guyon I, Elisseeff. An introduction to variable selection. JMLR 2003; 1157-82.
19. Witten IH, Frank E (2005). *Data Mining: Practical Machine Learning Tools and Techniques, 2nd Edition.* San Francisco: Morgan Kaufmann Publishers.
20. Atkinson S, Bihari D, Smithies M, Daly K. Identification of futility in intensive care. *Lancet* 1994; 344: 1203-1206.
21. Bion J. Susceptibiliy to critical illness: reserve, response and therapy. *Intensive Care Med* 2000; 26: S57-S63.
22. Gari D. Clifford, Daniel J. Scott and Mauricio Villarroel. User Guide and Documentation for the MIMIC II Database, version 2, release 1. Rev: 179 , 2009-04-16. http://mimic.mit.edu/documentation.html

23. Chertow GM, Burdick E, Honour M, Bonventre JV, Bates DW. Acute kidney injury, mortality, length of stay, and costs in hospitalized patients. *J AM Soc Nephrol* 2005; 16: 3365-70.
24. Girou E, Stephan F, Novara A, et al. Risk factor and outcome of nosocomial infections: Results of a matched case-control study of ICU patients. *Am J Respir Crit Care* 1998; 157: 1151-58.
25. Afessa B, Keegan MT, Mohammad Z, Finkelman JD, Peters SG. Identifying potentially ineffective care in the sickest critically ill patents on the third ICU day. Chest 2004; 126: 1905-9.
26. Du Cheyron D, Bouchet B, Parienti JJ, Ramakers M, Charbonneau P. The attributable mortality of acute renal failure in critically ill patients with liver cirrhosis. Intensive Care Med 2005; 31: 1693-99.
27. Youngner SJ. Medical futility. *Crit Care Clin* 1996; 12: 165-78.
28. Cantor MD, Braddock CH II, Derse AR, et al. Do-not-resuscitate orders and medical futility. *Arch Intern Med* 2003; 163: 2689-94.
29. Taleb NN (2007). *The Black Swan: The Impact of the Highly Improbable*. New York: Random House.

Acknowledgements:

I would like to thank Mauricio Villaroel, Christian Hinske, Guido Davidzon, Richard Lu, Ronilda Lacson, Caleb Hug, Gari Clifford, the Laboratory of Computer Science, and Professors Roger Mark and Peter Szolovits, for all their help and support.

Appendix A. SQL Query to Extract Patient Cohort from MIMIC II

```
CREATE OR REPLACE VIEW LEO_EVOLUTION (subject_id,expire_flg, age, sex,
       intime, outtime, los,
       max_bili_1st_day,  max_bili_2nd_day,  max_bili_3rd_day,
       max_creat_1st_day, max_creat_2nd_day, max_creat_3rd_day,
       min_hr_1st_day,    max_hr_1st_day,    stddev_hr_1st_day,
avg_hr_1st_day,
       min_hr_2nd_day,    max_hr_2nd_day,    stddev_hr_2nd_day,
avg_hr_2nd_day,
       min_hr_3rd_day,    max_hr_3rd_day,    stddev_hr_3rd_day,
avg_hr_3rd_day,
       min_sodium_1st_day,max_sodium_1st_day,stddev_sodium_1st_day,
avg_sodium_1st_day,
       min_sodium_2nd_day,max_sodium_2nd_day,stddev_sodium_2nd_day,
avg_sodium_2nd_day,
       min_sodium_3rd_day,max_sodium_3rd_day,stddev_sodium_3rd_day,
avg_sodium_3rd_day,
       min_sysbp_1st_day,max_sysbp_1st_day,stddev_sysbp_1st_day,
avg_sysbp_1st_day,
       min_sysbp_2nd_day,max_sysbp_2nd_day,stddev_sysbp_2nd_day,
avg_sysbp_2nd_day,
       min_sysbp_3rd_day,max_sysbp_3rd_day,stddev_sysbp_3rd_day,
avg_sysbp_3rd_day,
       max_resp_1st_day,   max_resp_2nd_day,  max_resp_3rd_day,
       min_hematocrit_1st_day, max_hematocrit_1st_day,
stddev_hematocrit_1st_day, avg_hematocrit_1st_day,
       min_hematocrit_2nd_day, max_hematocrit_2nd_day,
stddev_hematocrit_2nd_day, avg_hematocrit_2nd_day,
       min_hematocrit_3rd_day, max_hematocrit_3rd_day,
stddev_hematocrit_3rd_day, avg_hematocrit_3rd_day,
       min_glucose_1st_day, max_glucose_1st_day, stddev_glucose_1st_day,
avg_glucose_1st_day,
       min_glucose_2nd_day, max_glucose_2nd_day, stddev_glucose_2nd_day,
avg_glucose_2nd_day,
       min_glucose_3rd_day, max_glucose_3rd_day, stddev_glucose_3rd_day,
avg_glucose_3rd_day,
       min_wbc_1st_day,     max_wbc_1st_day,     stddev_wbc_1st_day,
avg_wbc_1st_day,
       min_wbc_2nd_day,     max_wbc_2nd_day,     stddev_wbc_2nd_day,
avg_wbc_2nd_day,
       min_wbc_3rd_day,     max_wbc_3rd_day,     stddev_wbc_3rd_day,
avg_wbc_3rd_day,
       min_potassium_1st_day, max_potassium_1st_day,
stddev_potassium_1st_day, avg_potassium_1st_day,
       min_potassium_2nd_day, max_potassium_2nd_day,
stddev_potassium_2nd_day, avg_potassium_2nd_day,
       min_potassium_3rd_day, max_potassium_3rd_day,
stddev_potassium_3rd_day, avg_potassium_3rd_day,
       max_bun_1st_day,    max_bun_2nd_day,    max_bun_3rd_day,
       min_gcs_1st_day,
       min_temp_1st_day,   max_temp_1st_day,   stddev_temp_1st_day,
avg_temp_1st_day,
       min_temp_2nd_day,   max_temp_2nd_day,   stddev_temp_2nd_day,
avg_temp_2nd_day,
       min_temp_3rd_day,   max_temp_3rd_day,   stddev_temp_3rd_day,
avg_temp_3rd_day,
```

```
       min_bicarbonate_1st_day, max_bicarbonate_1st_day,
stddev_bicarbonate_1st_day, avg_bicarbonate_1st_day,
       min_bicarbonate_2nd_day, max_bicarbonate_2nd_day,
stddev_bicarbonate_2nd_day, avg_bicarbonate_2nd_day,
       min_bicarbonate_3rd_day, max_bicarbonate_3rd_day,
stddev_bicarbonate_3rd_day, avg_bicarbonate_3rd_day,
       in_1st_day, out_1st_day,
       in_2nd_day, out_2nd_day,
       in_3rd_day, out_3rd_day
      ) AS
-- Query returns the last ICU stay for adult patients with
-- Subarachnoid Hemorrage (ICD9 code 584.9) that were at least 3 days
-- in the ICU
WITH lastStay as (
  select distinct subject_id,
         first_value(intime)
           over ( partition by subject_id
                  order by subject_id, intime DESC) as intime,
         first_value(outtime)
           over ( partition by subject_id
                  order by subject_id, intime DESC) as outtime,
         first_value(expire_flg)
           over ( partition by subject_id
                  order by subject_id, intime DESC) as expire_flg,
         first_value(age)
           over ( partition by subject_id
                  order by subject_id, intime DESC) as age,
         first_value(sex)
           over ( partition by subject_id
                  order by subject_id, intime DESC) as sex
    from (
          -- Returns any ICU stay that falls between the
          -- hospitalization period
          select had.subject_id, icu.icustay_id, icu.intime, icu.outtime,
                 had.expire_flg, icu.age, icu.sex
            from (
                  -- Returns the last hospital admissions for patients
                  -- with the required ICD9 code
                  select distinct subject_id,
                         first_value(adm_dt)
                           over(partition by subject_id
                                order by subject_id, adm_dt DESC) as adm_dt,
                         first_value(disch_dt)
                           over(partition by subject_id
                                order by subject_id, adm_dt DESC) as
disch_dt,
                         first_value(expire_flg)
                           over(partition by subject_id
                                order by subject_id, adm_dt DESC) as
expire_flg
                    from (
                          -- Returns all the hospital admissions for people
                          -- who had the ICD9 code: Subarachnoid Hemorrage
                          select a.hadm_id, a.subject_id, a.adm_dt,
                                 a.disch_dt, a.expire_flg
                            from mimic2v21.admissions a,
                                 mimic2v21.icd9 icd
```

```
                        where icd.hadm_id = a.hadm_id
                          and icd.code in ('584.9')
                     )
                 order by  subject_id
               ) had,
               (
                -- Returns all the adult ICU stays where the patient
                -- was least 3 days in the ICU
                select icustay_id, subject_id, intime, outtime,
                       (months_between(intime, dob) /12) age,
                       sex
                  from (
                        -- Returns all ICU stays
                        SELECT s.icustay_id, cen.subject_id,dob,
                               p.sex,
                               min(cen.intime) as intime,
                               max(cen.outtime) as outtime
                          FROM mimic2v21.icu_stay s,
                               mimic2v21.censusevents cen,
                               mimic2v21.d_patients p
                         WHERE cen.census_id = s.census_id
                           and p.subject_id = cen.subject_id
                           and p.dob is not null
                         GROUP BY s.icustay_id, cen.subject_id, dob, p.sex
                       )
                where (months_between(intime, dob) /12) >= 15
                  and (outtime - intime) >= 3
               ) icu
         where icu.subject_id = had.subject_id
           and icu.intime >= had.adm_dt
           and icu.intime <= had.disch_dt
       )
   --where subject_id in (117, 145, 250, 252)
   order by subject_id
),
-- Returns the max, min, stddev, average calues for each category
-- and each of the 3 days of the last ICU stay
RawData as (
    select distinct subject_id,expire_flg, round(age) age, sex,
           intime, outtime, round(outtime - intime, 1) los,
           max(bili_1st_day)
             over (partition by subject_id)     as max_bili_1st_day,
           max(bili_2nd_day)
             over (partition by subject_id)     as max_bili_2nd_day,
           max(bili_3rd_day)
             over (partition by subject_id)     as max_bili_3rd_day,
           max(creat_1st_day)
             over (partition by subject_id)     as max_creat_1st_day,
           max(creat_2nd_day)
             over (partition by subject_id)     as max_creat_2nd_day,
           max(creat_3rd_day)
             over (partition by subject_id)     as max_creat_3rd_day,
           min(hr_1st_day)
             over (partition by subject_id)     as min_hr_1st_day,
           max(hr_1st_day)
             over (partition by subject_id)     as max_hr_1st_day,
           round(stddev(hr_1st_day)
```

```
  over (partition by subject_id), 1) as stddev_hr_1st_day,
round(avg(hr_1st_day)
  over (partition by subject_id), 1) as avg_hr_1st_day,
min(hr_2nd_day)
  over (partition by subject_id)     as min_hr_2nd_day,
max(hr_2nd_day)
  over (partition by subject_id)     as max_hr_2nd_day,
round(stddev(hr_2nd_day)
  over (partition by subject_id), 1) as stddev_hr_2nd_day,
round(avg(hr_2nd_day)
  over (partition by subject_id), 1) as avg_hr_2nd_day,
min(hr_3rd_day)
  over (partition by subject_id)     as min_hr_3rd_day,
max(hr_3rd_day)
  over (partition by subject_id)     as max_hr_3rd_day,
round(stddev(hr_3rd_day)
  over (partition by subject_id), 1) as stddev_hr_3rd_day,
round(avg(hr_3rd_day)
  over (partition by subject_id), 1) as avg_hr_3rd_day,
min(sodium_1st_day)
  over (partition by subject_id)     as min_sodium_1st_day,
max(sodium_1st_day)
  over (partition by subject_id)     as max_sodium_1st_day,
round(stddev(sodium_1st_day)
  over (partition by subject_id), 1) as stddev_sodium_1st_day,
round(avg(sodium_1st_day)
  over (partition by subject_id), 1) as avg_sodium_1st_day,
min(sodium_2nd_day)
  over (partition by subject_id)     as min_sodium_2nd_day,
max(sodium_2nd_day)
  over (partition by subject_id)     as max_sodium_2nd_day,
round(stddev(sodium_2nd_day)
  over (partition by subject_id), 1) as stddev_sodium_2nd_day,
round(avg(sodium_2nd_day)
  over (partition by subject_id), 1) as avg_sodium_2nd_day,
min(sodium_3rd_day)
  over (partition by subject_id)     as min_sodium_3rd_day,
max(sodium_3rd_day)
  over (partition by subject_id)     as max_sodium_3rd_day,
round(stddev(sodium_3rd_day)
  over (partition by subject_id), 1) as stddev_sodium_3rd_day,
round(avg(sodium_3rd_day)
  over (partition by subject_id), 1) as avg_sodium_3rd_day,
min(sysbp_1st_day)
  over (partition by subject_id)     as min_sysbp_1st_day,
max(sysbp_1st_day)
  over (partition by subject_id)     as max_sysbp_1st_day,
round(stddev(sysbp_1st_day)
  over (partition by subject_id), 1) as stddev_sysbp_1st_day,
round(avg(sysbp_1st_day)
  over (partition by subject_id), 1) as avg_sysbp_1st_day,
min(sysbp_2nd_day)
  over (partition by subject_id)     as min_sysbp_2nd_day,
max(sysbp_2nd_day)
  over (partition by subject_id)     as max_sysbp_2nd_day,
round(stddev(sysbp_2nd_day)
  over (partition by subject_id), 1) as stddev_sysbp_2nd_day,
```

```
round(avg(sysbp_2nd_day)
  over (partition by subject_id), 1) as avg_sysbp_2nd_day,
min(sysbp_3rd_day)
  over (partition by subject_id)     as min_sysbp_3rd_day,
max(sysbp_3rd_day)
  over (partition by subject_id)     as max_sysbp_3rd_day,
round(stddev(sysbp_3rd_day)
  over (partition by subject_id), 1) as stddev_sysbp_3rd_day,
round(avg(sysbp_3rd_day)
  over (partition by subject_id), 1) as avg_sysbp_3rd_day,
min(resp_1st_day)
  over (partition by subject_id)     as min_resp_1st_day,
max(resp_1st_day)
  over (partition by subject_id)     as max_resp_1st_day,
round(stddev(resp_1st_day)
  over (partition by subject_id), 1) as stddev_resp_1st_day,
round(avg(resp_1st_day)
  over (partition by subject_id), 1) as avg_resp_1st_day,
min(resp_2nd_day)
  over (partition by subject_id)     as min_resp_2nd_day,
max(resp_2nd_day)
  over (partition by subject_id)     as max_resp_2nd_day,
round(stddev(resp_2nd_day)
  over (partition by subject_id), 1) as stddev_resp_2nd_day,
round(avg(resp_2nd_day)
  over (partition by subject_id), 1) as avg_resp_2nd_day,
min(resp_3rd_day)
  over (partition by subject_id)     as min_resp_3rd_day,
max(resp_3rd_day)
  over (partition by subject_id)     as max_resp_3rd_day,
round(stddev(resp_3rd_day)
  over (partition by subject_id), 1) as stddev_resp_3rd_day,
round(avg(resp_3rd_day)
  over (partition by subject_id), 1) as avg_resp_3rd_day,
round(min(hematocrit_1st_day)
  over (partition by subject_id), 1) as min_hematocrit_1st_day,
round(max(hematocrit_1st_day)
  over (partition by subject_id), 1) as max_hematocrit_1st_day,
round(stddev(hematocrit_1st_day)
  over (partition by subject_id), 1) as stddev_hematocrit_1st_day,
round(avg(hematocrit_1st_day)
  over (partition by subject_id), 1) as avg_hematocrit_1st_day,
round(min(hematocrit_2nd_day)
  over (partition by subject_id), 1) as min_hematocrit_2nd_day,
round(max(hematocrit_2nd_day)
  over (partition by subject_id), 1) as max_hematocrit_2nd_day,
round(stddev(hematocrit_2nd_day)
  over (partition by subject_id), 1) as stddev_hematocrit_2nd_day,
round(avg(hematocrit_2nd_day)
  over (partition by subject_id), 1) as avg_hematocrit_2nd_day,
round(min(hematocrit_3rd_day)
  over (partition by subject_id), 1) as min_hematocrit_3rd_day,
round(max(hematocrit_3rd_day)
  over (partition by subject_id), 1) as max_hematocrit_3rd_day,
round(stddev(hematocrit_3rd_day)
  over (partition by subject_id), 1) as stddev_hematocrit_3rd_day,
round(avg(hematocrit_3rd_day)
```

```
  over (partition by subject_id), 1) as avg_hematocrit_3rd_day,
min(glucose_1st_day)
  over (partition by subject_id)     as min_glucose_1st_day,
max(glucose_1st_day)
  over (partition by subject_id)     as max_glucose_1st_day,
round(stddev(glucose_1st_day)
  over (partition by subject_id), 1) as stddev_glucose_1st_day,
round(avg(glucose_1st_day)
  over (partition by subject_id), 1) as avg_glucose_1st_day,
min(glucose_2nd_day)
  over (partition by subject_id)     as min_glucose_2nd_day,
max(glucose_2nd_day)
  over (partition by subject_id)     as max_glucose_2nd_day,
round(stddev(glucose_2nd_day)
  over (partition by subject_id), 1) as stddev_glucose_2nd_day,
round(avg(glucose_2nd_day)
  over (partition by subject_id), 1) as avg_glucose_2nd_day,
min(glucose_3rd_day)
  over (partition by subject_id)     as min_glucose_3rd_day,
max(glucose_3rd_day)
  over (partition by subject_id)     as max_glucose_3rd_day,
round(stddev(glucose_3rd_day)
  over (partition by subject_id), 1) as stddev_glucose_3rd_day,
round(avg(glucose_3rd_day)
  over (partition by subject_id), 1) as avg_glucose_3rd_day,
min(wbc_1st_day)
  over (partition by subject_id)     as min_wbc_1st_day,
max(wbc_1st_day)
  over (partition by subject_id)     as max_wbc_1st_day,
round(stddev(wbc_1st_day)
  over (partition by subject_id), 1) as stddev_wbc_1st_day,
round(avg(wbc_1st_day)
  over (partition by subject_id), 1) as avg_wbc_1st_day,
min(wbc_2nd_day)
  over (partition by subject_id)     as min_wbc_2nd_day,
max(wbc_2nd_day)
  over (partition by subject_id)     as max_wbc_2nd_day,
round(stddev(wbc_2nd_day)
  over (partition by subject_id), 1) as stddev_wbc_2nd_day,
round(avg(wbc_2nd_day)
  over (partition by subject_id), 1) as avg_wbc_2nd_day,
min(wbc_3rd_day)
  over (partition by subject_id)     as min_wbc_3rd_day,
max(wbc_3rd_day)
  over (partition by subject_id)     as max_wbc_3rd_day,
round(stddev(wbc_3rd_day)
  over (partition by subject_id), 1) as stddev_wbc_3rd_day,
round(avg(wbc_3rd_day)
  over (partition by subject_id), 1) as avg_wbc_3rd_day,
min(potassium_1st_day)
  over (partition by subject_id)     as min_potassium_1st_day,
max(potassium_1st_day)
  over (partition by subject_id)     as max_potassium_1st_day,
round(stddev(potassium_1st_day)
  over (partition by subject_id), 1) as stddev_potassium_1st_day,
round(avg(potassium_1st_day)
  over (partition by subject_id), 1) as avg_potassium_1st_day,
```

```
            min(potassium_2nd_day)
              over (partition by subject_id)     as min_potassium_2nd_day,
            max(potassium_2nd_day)
              over (partition by subject_id)     as max_potassium_2nd_day,
            round(stddev(potassium_2nd_day)
              over (partition by subject_id), 1) as stddev_potassium_2nd_day,
            round(avg(potassium_2nd_day)
              over (partition by subject_id), 1) as avg_potassium_2nd_day,
            min(potassium_3rd_day)
              over (partition by subject_id)     as min_potassium_3rd_day,
            max(potassium_3rd_day)
              over (partition by subject_id)     as max_potassium_3rd_day,
            round(stddev(potassium_3rd_day)
              over (partition by subject_id), 1) as stddev_potassium_3rd_day,
            round(avg(potassium_3rd_day)
              over (partition by subject_id), 1) as avg_potassium_3rd_day,
            max(bun_1st_day)
              over (partition by subject_id)     as max_bun_1st_day,
            max(bun_2nd_day)
              over (partition by subject_id)     as max_bun_2nd_day,
            max(bun_3rd_day)
              over (partition by subject_id)     as max_bun_3rd_day,
            min(gcs_1st_day)
              over (partition by subject_id)     as min_gcs_1st_day,
            round(min(temp_1st_day)
              over (partition by subject_id), 1) as min_temp_1st_day,
            round(max(temp_1st_day)
              over (partition by subject_id), 1) as max_temp_1st_day,
            round(stddev(temp_1st_day)
              over (partition by subject_id), 1) as stddev_temp_1st_day,
            round(avg(temp_1st_day)
              over (partition by subject_id), 1) as avg_temp_1st_day,
            round(min(temp_2nd_day)
              over (partition by subject_id), 1) as min_temp_2nd_day,
            round(max(temp_2nd_day)
              over (partition by subject_id), 1) as max_temp_2nd_day,
            round(stddev(temp_2nd_day)
              over (partition by subject_id), 1) as stddev_temp_2nd_day,
            round(avg(temp_2nd_day)
              over (partition by subject_id), 1) as avg_temp_2nd_day,
            round(min(temp_3rd_day)
              over (partition by subject_id), 1) as min_temp_3rd_day,
            round(max(temp_3rd_day)
              over (partition by subject_id), 1) as max_temp_3rd_day,
            round(stddev(temp_3rd_day)
              over (partition by subject_id), 1) as stddev_temp_3rd_day,
            round(avg(temp_3rd_day)
              over (partition by subject_id), 1) as avg_temp_3rd_day,
            min(bicarbonate_1st_day)
              over (partition by subject_id)     as min_bicarbonate_1st_day,
            max(bicarbonate_1st_day)
              over (partition by subject_id)     as max_bicarbonate_1st_day,
            round(stddev(bicarbonate_1st_day)
              over (partition by subject_id), 1) as
stddev_bicarbonate_1st_day,
            round(avg(bicarbonate_1st_day)
              over (partition by subject_id), 1) as avg_bicarbonate_1st_day,
```

```
           min(bicarbonate_2nd_day)
             over (partition by subject_id)     as min_bicarbonate_2nd_day,
           max(bicarbonate_2nd_day)
             over (partition by subject_id)     as max_bicarbonate_2nd_day,
           round(stddev(bicarbonate_2nd_day)
             over (partition by subject_id), 1) as
stddev_bicarbonate_2nd_day,
           round(avg(bicarbonate_2nd_day)
             over (partition by subject_id), 1) as avg_bicarbonate_2nd_day,
           min(bicarbonate_3rd_day)
             over (partition by subject_id)     as min_bicarbonate_3rd_day,
           max(bicarbonate_3rd_day)
             over (partition by subject_id)     as max_bicarbonate_3rd_day,
           round(stddev(bicarbonate_3rd_day)
             over (partition by subject_id), 1) as
stddev_bicarbonate_3rd_day,
           round(avg(bicarbonate_3rd_day)
             over (partition by subject_id), 1) as avg_bicarbonate_3rd_day,
           round(first_value(in_1st_day)
              over(partition by subject_id
                   order by subject_id, intime), 1) as in_1st_day,
           round(first_value(out_1st_day)
              over(partition by subject_id
                   order by subject_id, intime), 1) as out_1st_day,
           round(first_value(in_2nd_day)
              over(partition by subject_id
                   order by subject_id, intime), 1) as in_2nd_day,
           round(first_value(out_2nd_day)
              over(partition by subject_id
                   order by subject_id, intime), 1) as out_2nd_day,
           round(first_value(in_3rd_day)
              over(partition by subject_id
                   order by subject_id, intime), 1) as in_3rd_day,
           round(first_value(out_3rd_day)
              over(partition by subject_id
                   order by subject_id, intime), 1) as out_3rd_day
      from (
            -- Tags each value in each category as comming from the first,
            -- second or third day in the ICU
            select subject_id, expire_flg, age, sex, intime, outtime,
                   case
                      when parameter = 'BILIRUBIN' and day_in_icu = 1
                          then value1num
                      else
                          null
                   end as bili_1st_day,
                   case
                      when parameter = 'BILIRUBIN' and day_in_icu = 2
                          then value1num
                      else
                          null
                   end as bili_2nd_day,
                   case
                      when parameter = 'BILIRUBIN' and day_in_icu = 3
                          then value1num
                      else
                          null
```

```
end as bili_3rd_day,
case
   when parameter = 'CREATININE' and day_in_icu = 1
       then value1num
   else
       null
end as creat_1st_day,
case
   when parameter = 'CREATININE' and day_in_icu = 2
       then value1num
   else
       null
end as creat_2nd_day,
case
   when parameter = 'CREATININE' and day_in_icu = 3
       then value1num
   else
       null
end as creat_3rd_day,
case
   when parameter = 'HR' and day_in_icu = 1
       then value1num
   else
       null
end as hr_1st_day,
case
   when parameter = 'HR' and day_in_icu = 2
       then value1num
   else
       null
end as hr_2nd_day,
case
   when parameter = 'HR' and day_in_icu = 3
       then value1num
   else
       null
end as hr_3rd_day,
case
   when parameter = 'SODIUM' and day_in_icu = 1
       then value1num
   else
       null
end as sodium_1st_day,
case
   when parameter = 'SODIUM' and day_in_icu = 2
       then value1num
   else
       null
end as sodium_2nd_day,
case
   when parameter = 'SODIUM' and day_in_icu = 3
       then value1num
   else
       null
end as sodium_3rd_day,
case
   when parameter = 'SYS_BP' and day_in_icu = 1
```

```
        then value1num
    else
        null
end as sysbp_1st_day,
case
    when parameter = 'SYS_BP' and day_in_icu = 2
        then value1num
    else
        null
end as sysbp_2nd_day,
case
    when parameter = 'SYS_BP' and day_in_icu = 3
        then value1num
    else
        null
end as sysbp_3rd_day,
case
    when parameter = 'RESPIRATION' and day_in_icu = 1
        then value1num
    else
        null
end as resp_1st_day,
case
    when parameter = 'RESPIRATION' and day_in_icu = 2
        then value1num
    else
        null
end as resp_2nd_day,
case
    when parameter = 'RESPIRATION' and day_in_icu = 3
        then value1num
    else
        null
end as resp_3rd_day,
case
    when parameter = 'HEMATOCRIT' and day_in_icu = 1
        then value1num
    else
        null
end as hematocrit_1st_day,
case
    when parameter = 'HEMATOCRIT' and day_in_icu = 2
        then value1num
    else
        null
end as hematocrit_2nd_day,
case
    when parameter = 'HEMATOCRIT' and day_in_icu = 3
        then value1num
    else
        null
end as hematocrit_3rd_day,
case
    when parameter = 'GLUCOSE' and day_in_icu = 1
        then value1num
    else
        null
```

```
end as glucose_1st_day,
case
   when parameter = 'GLUCOSE' and day_in_icu = 2
       then value1num
   else
       null
end as glucose_2nd_day,
case
   when parameter = 'GLUCOSE' and day_in_icu = 3
       then value1num
   else
       null
end as glucose_3rd_day,
case
   when parameter = 'WBC' and day_in_icu = 1
       then value1num
   else
       null
end as wbc_1st_day,
case
   when parameter = 'WBC' and day_in_icu = 2
       then value1num
   else
       null
end as wbc_2nd_day,
case
   when parameter = 'WBC' and day_in_icu = 3
       then value1num
   else
       null
end as wbc_3rd_day,
case
   when parameter = 'POTASSIUM' and day_in_icu = 1
       then value1num
   else
       null
end as potassium_1st_day,
case
   when parameter = 'POTASSIUM' and day_in_icu = 2
       then value1num
   else
       null
end as potassium_2nd_day,
case
   when parameter = 'POTASSIUM' and day_in_icu = 3
       then value1num
   else
       null
end as potassium_3rd_day,
case
   when parameter = 'BUN' and day_in_icu = 1
       then value1num
   else
       null
end as bun_1st_day,
case
   when parameter = 'BUN' and day_in_icu = 2
```

```
        then value1num
    else
        null
end as bun_2nd_day,
case
    when parameter = 'BUN' and day_in_icu = 3
        then value1num
    else
        null
end as bun_3rd_day,
case
    when parameter = 'GCS' and day_in_icu = 1
        then value1num
    else
        null
end as gcs_1st_day,
case
    when parameter = 'TEMP' and day_in_icu = 1
        then value1num
    else
        null
end as temp_1st_day,
case
    when parameter = 'TEMP' and day_in_icu = 2
        then value1num
    else
        null
end as temp_2nd_day,
case
    when parameter = 'TEMP' and day_in_icu = 3
        then value1num
    else
        null
end as temp_3rd_day,
case
    when parameter = 'BICARBONATE' and day_in_icu = 1
        then value1num
    else
        null
end as bicarbonate_1st_day,
case
    when parameter = 'BICARBONATE' and day_in_icu = 2
        then value1num
    else
        null
end as bicarbonate_2nd_day,
case
    when parameter = 'BICARBONATE' and day_in_icu = 3
        then value1num
    else
        null
end as bicarbonate_3rd_day,
case
    when parameter = 'TOTAL_IN' and day_in_icu = 1
        then value1num
    else
        null
```

```
        end as in_1st_day,
        case
           when parameter = 'TOTAL_IN' and day_in_icu = 2
               then value1num
           else
               null
        end as in_2nd_day,
        case
           when parameter = 'TOTAL_IN' and day_in_icu = 3
               then value1num
           else
               null
        end as in_3rd_day,
        case
           when parameter = 'TOTAL_OUT' and day_in_icu = 1
               then value1num
           else
               null
        end as out_1st_day,
        case
           when parameter = 'TOTAL_OUT' and day_in_icu = 2
               then value1num
           else
               null
        end as out_2nd_day,
        case
           when parameter = 'TOTAL_OUT' and day_in_icu = 3
               then value1num
           else
               null
        end as out_3rd_day
from (
         -- The whole raw data, grouping itemids in categories and
         -- the day in the ICU the measurement occurs.
         select subject_id, expire_flg, age, sex,
                intime, outtime,
                case
                     when itemid in (848, 1538)
                         then 'BILIRUBIN'
                     when itemid in (791, 3750, 1525)
                         then 'CREATININE'
                     when itemid in (211)
                         then 'HR'
                     when itemid in (837, 1536, 3803)
                         then 'SODIUM'
                     when itemid in (51, 455)
                         then 'SYS_BP'
                     when itemid in (618)
                         then 'RESPIRATION'
                     when itemid in (813)
                         then 'HEMATOCRIT'
                     when itemid in (811, 1529)
                         then 'GLUCOSE'
                     when itemid in (1542, 1127, 861, 4200)
                         then 'WBC'
                     when itemid in (829, 1535, 3792)
                         then 'POTASSIUM'
```

```
                                when itemid in (781)
                                    then 'BUN'
                                when itemid in (198)
                                    then 'GCS'
                                when itemid in (676, 677, 678, 679)
                                    then 'TEMP'
                                when itemid in (787)
                                    then 'BICARBONATE'
                                when itemid in (1)
                                    then 'TOTAL_IN'
                                when itemid in (2)
                                    then 'TOTAL_OUT'
                               else 'unknown'
                           end as parameter,
                           case
                               when charttime <= intime + 1
                                  then 1
                               when (charttime > intime + 1)
                                     and (charttime <= intime + 2)
                                  then 2
                               when (charttime > intime + 2)
                                     and (charttime <= intime + 3)
                                  then 3
                               else
                                   -1
                           end as day_in_icu,
                           case
                              -- convert farenheit to celsius
                               when itemid in (678, 679) and (value1 is not
null)
                                   then ((value1 - 32) * 5/9)
                               else
                                   value1
                           end as value1num
                      from (
                            -- Get the charted events for the required
itemids
                            -- from patients in "lastStay"
                            select pts.subject_id, pts.expire_flg, pts.age,
                                   pts.sex, pts.intime, pts.outtime,
                                   c.itemid, c.charttime, c.value1num as
value1,
                                   c.value2num as value2
                              from lastStay pts,
                                   mimic2v21.chartevents c
                             where c.subject_id = pts.subject_id
                               and c.itemid in (848, 1538,
                                               791, 3750, 1525,
                                               211,
                                               837, 1536,3803,
                                               51, 455,
                                               618,
                                               813,
                                               811, 1529,
                                               1542, 1127, 861, 4200,
                                               829, 1535, 3792,
                                               781,
```

```
                                                   198,
                                                   676, 677, 678, 679,
                                                   787)
                               and c.charttime >= pts.intime
                               and c.charttime <= (pts.intime + 3)
                            UNION
                            -- Get the total INPUT/OUTPUT from patients
                            -- in "lastStay"
                            select pts.subject_id, pts.expire_flg, pts.age,
                                   pts.sex, pts.intime, pts.outtime,
                                   te.itemid, te.charttime, te.cumvolume as
value1,
                                   0 as value2
                              from lastStay pts,
                                   mimic2v21.totalbalevents te
                             where te.subject_id = pts.subject_id
                               and te.itemid in (1, 2)
                               and te.charttime >= pts.intime
                               and te.charttime <= (pts.intime + 3)
                          )
                   )
             ORDER BY subject_id
           )
       order by subject_id, intime
),
-- 2.Bilirubin (max)
--     a.If all 3 values are missing, put 0.7.  Reason:  This is the middle
--       of the normal range.
--     b.If 2 values are missing, replace them with the level that is
present.
--     c.If the middle value is missing, replace it with the average of
--       the other 2 values.  Otherwise, replace it with the value thatâ€™s
--       closer time-wise.
Rule2Bili as (
 select subject_id,
         case
            when (max_bili_1st_day is null)     and (max_bili_2nd_day is
null)     and (max_bili_3rd_day is null)     then
                0.7
            when (max_bili_1st_day is null)     and (max_bili_2nd_day is
null)     and (max_bili_3rd_day is not null) then
               max_bili_3rd_day
            when (max_bili_1st_day is null)     and (max_bili_2nd_day is not
null) and (max_bili_3rd_day is null)     then
               max_bili_2nd_day
            else
               max_bili_1st_day
         end max_bili_1st_day,
         case
            when (max_bili_1st_day is null)     and (max_bili_2nd_day is
null)     and (max_bili_3rd_day is null)     then
                0.7
            when (max_bili_1st_day is null)     and (max_bili_2nd_day is
null)     and (max_bili_3rd_day is not null) then
               max_bili_3rd_day
            when (max_bili_1st_day is not null) and (max_bili_2nd_day is
null)     and (max_bili_3rd_day is null)     then
```

```
               max_bili_1st_day
            when (max_bili_1st_day is not null) and (max_bili_2nd_day is
null)     and (max_bili_3rd_day is not null) then
               (max_bili_1st_day + max_bili_3rd_day) / 2
            else
               max_bili_2nd_day
         end max_bili_2nd_day,
         case
            when (max_bili_1st_day is null)     and (max_bili_2nd_day is
null)     and (max_bili_3rd_day is null)     then
                0.7
            when (max_bili_1st_day is not null) and (max_bili_2nd_day is
null)     and (max_bili_3rd_day is null)     then
               max_bili_1st_day
            when (max_bili_1st_day is null)     and (max_bili_2nd_day is not
null) and (max_bili_3rd_day is null)     then
               max_bili_2nd_day
            else
               max_bili_3rd_day
         end max_bili_3rd_day
    from RawData
),
-- 3.Creatinine (max)
--    a.If 2 values are missing, replace them with the level that is present.
--    b.If the middle value is missing, replace it with the average of the
--      other 2 values.  Otherwise, replace it with the value thatâ€™s
--      closer time-wise.
Rule3Creat as (
  select subject_id,
         case
            when (max_creat_1st_day is null)     and (max_creat_2nd_day is
null)     and (max_creat_3rd_day is not null) then
               max_creat_3rd_day
            when (max_creat_1st_day is null)     and (max_creat_2nd_day is
not null) and (max_creat_3rd_day is null)     then
               max_creat_2nd_day
            else
               max_creat_1st_day
         end max_creat_1st_day,
         case
            when (max_creat_1st_day is null)     and (max_creat_2nd_day is
null)     and (max_creat_3rd_day is not null) then
               max_creat_3rd_day
            when (max_creat_1st_day is not null) and (max_creat_2nd_day is
null)     and (max_creat_3rd_day is null)     then
               max_creat_1st_day
            when (max_creat_1st_day is not null) and (max_creat_2nd_day is
null)     and (max_creat_3rd_day is not null) then
               (max_creat_1st_day + max_creat_3rd_day) / 2
            else
               max_creat_2nd_day
         end max_creat_2nd_day,
         case
            when (max_creat_1st_day is not null) and (max_creat_2nd_day is
null)     and (max_creat_3rd_day is null)     then
               max_creat_1st_day
```

```
            when (max_creat_1st_day is null)     and (max_creat_2nd_day is
not null) and (max_creat_3rd_day is null)     then
               max_creat_2nd_day
            else
               max_creat_3rd_day
         end max_creat_3rd_day
    from RawData
),
-- 4.Heart Rate (min/max/average/SD)
--     a.If the first or third day values are missing, use the second day
values.
--     b.If the second day values are missing, use the average of the first
--       and third day values.
Rule4HR as (
 select subject_id,
         case
            when (min_hr_1st_day is null) then
                min_hr_2nd_day
            else
               min_hr_1st_day
         end min_hr_1st_day,
         case
            when (min_hr_1st_day is not null) and (min_hr_2nd_day is null)
and (min_hr_3rd_day is not null) then
               (min_hr_1st_day + min_hr_3rd_day) / 2
            else
               min_hr_2nd_day
         end min_hr_2nd_day,
         case
            when (min_hr_3rd_day is null) then
                min_hr_2nd_day
            else
               min_hr_3rd_day
         end min_hr_3rd_day,
         case
            when (max_hr_1st_day is null) then
                max_hr_2nd_day
            else
               max_hr_1st_day
         end max_hr_1st_day,
         case
            when (max_hr_1st_day is not null) and (max_hr_2nd_day is null)
and (max_hr_3rd_day is not null) then
               (max_hr_1st_day + max_hr_3rd_day) / 2
            else
               max_hr_2nd_day
         end max_hr_2nd_day,
         case
            when (max_hr_3rd_day is null) then
                max_hr_2nd_day
            else
               max_hr_3rd_day
         end max_hr_3rd_day,
         case
            when (stddev_hr_1st_day is null) then
                stddev_hr_2nd_day
            else
```

```
                stddev_hr_1st_day
          end stddev_hr_1st_day,
          case
             when (stddev_hr_1st_day is not null) and (stddev_hr_2nd_day is
null)     and (stddev_hr_3rd_day is not null) then
                (stddev_hr_1st_day + stddev_hr_3rd_day) / 2
             else
                stddev_hr_2nd_day
          end stddev_hr_2nd_day,
          case
             when (stddev_hr_3rd_day is null) then
                 stddev_hr_2nd_day
             else
                stddev_hr_3rd_day
          end stddev_hr_3rd_day,
          case
             when (avg_hr_1st_day is null) then
                 avg_hr_2nd_day
             else
                avg_hr_1st_day
          end avg_hr_1st_day,
          case
             when (avg_hr_1st_day is not null) and (avg_hr_2nd_day is null)
and (avg_hr_3rd_day is not null) then
                (avg_hr_1st_day + avg_hr_3rd_day) / 2
             else
                avg_hr_2nd_day
          end avg_hr_2nd_day,
          case
             when (avg_hr_3rd_day is null) then
                 avg_hr_2nd_day
             else
                avg_hr_3rd_day
          end avg_hr_3rd_day
    from RawData
),
-- 5.Sodium (min/max/average/SD)
--     a.If 2 days of min/max/average/SD values are missing, replace them
--       with min/max/average/SD value that is present.
--     b.If the second day min/max/average/SD values are missing, replace
--       them with the average of the other 2 daysâ€™ min/max/average/SD
values.
--       Otherwise, replace them with the min/max/average/SD values that are
--       closer time-wise.
Rule5Sodium as (
  select subject_id,
          case
             when (min_sodium_1st_day is null)     and (min_sodium_2nd_day is
null)     and (min_sodium_3rd_day is not null) then
                min_sodium_3rd_day
             when (min_sodium_1st_day is null)     and (min_sodium_2nd_day is
not null) and (min_sodium_3rd_day is null)     then
                min_sodium_2nd_day
             else
                min_sodium_1st_day
          end min_sodium_1st_day,
          case
```

```
            when (min_sodium_1st_day is null)     and (min_sodium_2nd_day is
null)     and (min_sodium_3rd_day is not null) then
               min_sodium_3rd_day
            when (min_sodium_1st_day is not null) and (min_sodium_2nd_day is
null)     and (min_sodium_3rd_day is null)     then
               min_sodium_1st_day
            when (min_sodium_1st_day is not null) and (min_sodium_2nd_day is
null)     and (min_sodium_3rd_day is not null) then
               (min_sodium_1st_day + min_sodium_3rd_day) / 2
            else
               min_sodium_2nd_day
         end min_sodium_2nd_day,
         case
            when (min_sodium_1st_day is not null) and (min_sodium_2nd_day is
null)     and (min_sodium_3rd_day is null)     then
               min_sodium_1st_day
            when (min_sodium_1st_day is null)     and (min_sodium_2nd_day is
not null) and (min_sodium_3rd_day is null)     then
               min_sodium_2nd_day
            else
               min_sodium_3rd_day
         end min_sodium_3rd_day,
         case
            when (max_sodium_1st_day is null)     and (max_sodium_2nd_day is
null)     and (max_sodium_3rd_day is not null) then
               max_sodium_3rd_day
            when (max_sodium_1st_day is null)     and (max_sodium_2nd_day is
not null) and (max_sodium_3rd_day is null)     then
               max_sodium_2nd_day
            else
               max_sodium_1st_day
         end max_sodium_1st_day,
         case
            when (max_sodium_1st_day is null)     and (max_sodium_2nd_day is
null)     and (max_sodium_3rd_day is not null) then
               max_sodium_3rd_day
            when (max_sodium_1st_day is not null) and (max_sodium_2nd_day is
null)     and (max_sodium_3rd_day is null)     then
               max_sodium_1st_day
            when (max_sodium_1st_day is not null) and (max_sodium_2nd_day is
null)     and (max_sodium_3rd_day is not null) then
               (max_sodium_1st_day + max_sodium_3rd_day) / 2
            else
               max_sodium_2nd_day
         end max_sodium_2nd_day,
         case
            when (max_sodium_1st_day is not null) and (max_sodium_2nd_day is
null)     and (max_sodium_3rd_day is null)     then
               max_sodium_1st_day
            when (max_sodium_1st_day is null)     and (max_sodium_2nd_day is
not null) and (max_sodium_3rd_day is null)     then
               max_sodium_2nd_day
            else
               max_sodium_3rd_day
         end max_sodium_3rd_day,
         case
```

```
            when (stddev_sodium_1st_day is null)     and
(stddev_sodium_2nd_day is null)     and (stddev_sodium_3rd_day is not null)
then
               stddev_sodium_3rd_day
            when (stddev_sodium_1st_day is null)     and
(stddev_sodium_2nd_day is not null) and (stddev_sodium_3rd_day is null)
then
               stddev_sodium_2nd_day
            else
               stddev_sodium_1st_day
         end stddev_sodium_1st_day,
         case
            when (stddev_sodium_1st_day is null)     and
(stddev_sodium_2nd_day is null)     and (stddev_sodium_3rd_day is not null)
then
               stddev_sodium_3rd_day
            when (stddev_sodium_1st_day is not null) and
(stddev_sodium_2nd_day is null)     and (stddev_sodium_3rd_day is null)
then
               stddev_sodium_1st_day
            when (stddev_sodium_1st_day is not null) and
(stddev_sodium_2nd_day is null)     and (stddev_sodium_3rd_day is not null)
then
               (stddev_sodium_1st_day + stddev_sodium_3rd_day) / 2
            else
               stddev_sodium_2nd_day
         end stddev_sodium_2nd_day,
         case
            when (stddev_sodium_1st_day is not null) and
(stddev_sodium_2nd_day is null)     and (stddev_sodium_3rd_day is null)
then
               stddev_sodium_1st_day
            when (stddev_sodium_1st_day is null)     and
(stddev_sodium_2nd_day is not null) and (stddev_sodium_3rd_day is null)
then
               stddev_sodium_2nd_day
            else
               stddev_sodium_3rd_day
         end stddev_sodium_3rd_day,
         case
            when (avg_sodium_1st_day is null)     and (avg_sodium_2nd_day is
null)     and (avg_sodium_3rd_day is not null) then
               avg_sodium_3rd_day
            when (avg_sodium_1st_day is null)     and (avg_sodium_2nd_day is
not null) and (avg_sodium_3rd_day is null)     then
               avg_sodium_2nd_day
            else
               avg_sodium_1st_day
         end avg_sodium_1st_day,
         case
            when (avg_sodium_1st_day is null)     and (avg_sodium_2nd_day is
null)     and (avg_sodium_3rd_day is not null) then
               avg_sodium_3rd_day
            when (avg_sodium_1st_day is not null) and (avg_sodium_2nd_day is
null)     and (avg_sodium_3rd_day is null)     then
               avg_sodium_1st_day
```

```
            when (avg_sodium_1st_day is not null) and (avg_sodium_2nd_day is
null)     and (avg_sodium_3rd_day is not null) then
               (avg_sodium_1st_day + avg_sodium_3rd_day) / 2
            else
               avg_sodium_2nd_day
         end avg_sodium_2nd_day,
         case
            when (avg_sodium_1st_day is not null) and (avg_sodium_2nd_day is
null)     and (avg_sodium_3rd_day is null)     then
               avg_sodium_1st_day
            when (avg_sodium_1st_day is null)     and (avg_sodium_2nd_day is
not null) and (avg_sodium_3rd_day is null)     then
               avg_sodium_2nd_day
            else
               avg_sodium_3rd_day
         end avg_sodium_3rd_day
    from RawData
),
--
--6.Respiratory Rate (max)
--  a.Use only maximum respiratory rate as a variable.  Reason:  There are
--    too many patients with 0 as the minimum respiratory rate.
--    Maximum respiratory rate is the important variable anyway.
--  b.If all 3 maximum RR are missing, put 20.  Reason:  This is the default
--    number that nurses write (even though it is wrong).
--  c.If 2 maximum RR are missing, replace them with the level that is
present.
--  d.If the middle maximum RR is missing, replace it with the average of the
--    other 2 maximum RR.  Otherwise, replace it with the maximum RR thatâ€™s
--    closer time-wise.
--
Rule6Resp as (
 select subject_id,
         case
            when (max_resp_1st_day is null)     and (max_resp_2nd_day is
null)     and (max_resp_3rd_day is null)     then
                20
            when (max_resp_1st_day is null)     and (max_resp_2nd_day is
null)     and (max_resp_3rd_day is not null) then
               max_resp_3rd_day
            when (max_resp_1st_day is null)     and (max_resp_2nd_day is not
null) and (max_resp_3rd_day is null)     then
               max_resp_2nd_day
            else
               max_resp_1st_day
         end max_resp_1st_day,
         case
            when (max_resp_1st_day is null)     and (max_resp_2nd_day is
null)     and (max_resp_3rd_day is null)     then
                20
            when (max_resp_1st_day is null)     and (max_resp_2nd_day is
null)     and (max_resp_3rd_day is not null) then
               max_resp_3rd_day
            when (max_resp_1st_day is not null) and (max_resp_2nd_day is
null)     and (max_resp_3rd_day is null)     then
               max_resp_1st_day
```

```
            when (max_resp_1st_day is not null) and (max_resp_2nd_day is
null)     and (max_resp_3rd_day is not null) then
               (max_resp_1st_day + max_resp_3rd_day) / 2
            else
               max_resp_2nd_day
         end max_resp_2nd_day,
         case
            when (max_resp_1st_day is null)     and (max_resp_2nd_day is
null)     and (max_resp_3rd_day is null)     then
                20
            when (max_resp_1st_day is not null) and (max_resp_2nd_day is
null)     and (max_resp_3rd_day is null)     then
               max_resp_1st_day
            when (max_resp_1st_day is null)     and (max_resp_2nd_day is not
null) and (max_resp_3rd_day is null)     then
               max_resp_2nd_day
            else
               max_resp_3rd_day
         end max_resp_3rd_day
    from RawData
),
--
--7.Hemotocrit (min/max/average/SD)
--  a.If 2 days of min/max/average/SD values are missing, replace them with
--    min/max/average/SD value that is present.
--  b.If the second day min/max/average/SD values are missing, replace them
--    with the average of the other 2 daysâ€™ min/max/average/SD values.
--    Otherwise, replace them with the min/max/average/SD values that are
--    closer time-wise.
--
Rule7Hemat as (
  select subject_id,
         case
            when (min_hematocrit_1st_day is null)     and
(min_hematocrit_2nd_day is null)     and (min_hematocrit_3rd_day is not null)
then
               min_hematocrit_3rd_day
            when (min_hematocrit_1st_day is null)     and
(min_hematocrit_2nd_day is not null) and (min_hematocrit_3rd_day is null)
then
               min_hematocrit_2nd_day
            else
               min_hematocrit_1st_day
         end min_hematocrit_1st_day,
         case
            when (min_hematocrit_1st_day is null)     and
(min_hematocrit_2nd_day is null)     and (min_hematocrit_3rd_day is not null)
then
               min_hematocrit_3rd_day
            when (min_hematocrit_1st_day is not null) and
(min_hematocrit_2nd_day is null)     and (min_hematocrit_3rd_day is null)
then
               min_hematocrit_1st_day
            when (min_hematocrit_1st_day is not null) and
(min_hematocrit_2nd_day is null)     and (min_hematocrit_3rd_day is not null)
then
               (min_hematocrit_1st_day + min_hematocrit_3rd_day) / 2
```

```
            else
               min_hematocrit_2nd_day
         end min_hematocrit_2nd_day,
         case
            when (min_hematocrit_1st_day is not null) and
(min_hematocrit_2nd_day is null)     and (min_hematocrit_3rd_day is null)
then
               min_hematocrit_1st_day
            when (min_hematocrit_1st_day is null)     and
(min_hematocrit_2nd_day is not null) and (min_hematocrit_3rd_day is null)
then
               min_hematocrit_2nd_day
            else
               min_hematocrit_3rd_day
         end min_hematocrit_3rd_day,
         case
            when (max_hematocrit_1st_day is null)     and
(max_hematocrit_2nd_day is null)     and (max_hematocrit_3rd_day is not null)
then
               max_hematocrit_3rd_day
            when (max_hematocrit_1st_day is null)     and
(max_hematocrit_2nd_day is not null) and (max_hematocrit_3rd_day is null)
then
               max_hematocrit_2nd_day
            else
               max_hematocrit_1st_day
         end max_hematocrit_1st_day,
         case
            when (max_hematocrit_1st_day is null)     and
(max_hematocrit_2nd_day is null)     and (max_hematocrit_3rd_day is not null)
then
               max_hematocrit_3rd_day
            when (max_hematocrit_1st_day is not null) and
(max_hematocrit_2nd_day is null)     and (max_hematocrit_3rd_day is null)
then
               max_hematocrit_1st_day
            when (max_hematocrit_1st_day is not null) and
(max_hematocrit_2nd_day is null)     and (max_hematocrit_3rd_day is not null)
then
               (max_hematocrit_1st_day + max_hematocrit_3rd_day) / 2
            else
               max_hematocrit_2nd_day
         end max_hematocrit_2nd_day,
         case
            when (max_hematocrit_1st_day is not null) and
(max_hematocrit_2nd_day is null)     and (max_hematocrit_3rd_day is null)
then
               max_hematocrit_1st_day
            when (max_hematocrit_1st_day is null)     and
(max_hematocrit_2nd_day is not null) and (max_hematocrit_3rd_day is null)
then
               max_hematocrit_2nd_day
            else
               max_hematocrit_3rd_day
         end max_hematocrit_3rd_day,
         case
```

```
            when (stddev_hematocrit_1st_day is null)     and
(stddev_hematocrit_2nd_day is null)     and (stddev_hematocrit_3rd_day is not
null) then
               stddev_hematocrit_3rd_day
            when (stddev_hematocrit_1st_day is null)     and
(stddev_hematocrit_2nd_day is not null) and (stddev_hematocrit_3rd_day is
null)     then
               stddev_hematocrit_2nd_day
            else
               stddev_hematocrit_1st_day
         end stddev_hematocrit_1st_day,
         case
            when (stddev_hematocrit_1st_day is null)     and
(stddev_hematocrit_2nd_day is null)     and (stddev_hematocrit_3rd_day is not
null) then
               stddev_hematocrit_3rd_day
            when (stddev_hematocrit_1st_day is not null) and
(stddev_hematocrit_2nd_day is null)     and (stddev_hematocrit_3rd_day is
null)     then
               stddev_hematocrit_1st_day
            when (stddev_hematocrit_1st_day is not null) and
(stddev_hematocrit_2nd_day is null)     and (stddev_hematocrit_3rd_day is not
null) then
               (stddev_hematocrit_1st_day + stddev_hematocrit_3rd_day) / 2
            else
               stddev_hematocrit_2nd_day
         end stddev_hematocrit_2nd_day,
         case
            when (stddev_hematocrit_1st_day is not null) and
(stddev_hematocrit_2nd_day is null)     and (stddev_hematocrit_3rd_day is
null)     then
               stddev_hematocrit_1st_day
            when (stddev_hematocrit_1st_day is null)     and
(stddev_hematocrit_2nd_day is not null) and (stddev_hematocrit_3rd_day is
null)     then
               stddev_hematocrit_2nd_day
            else
               stddev_hematocrit_3rd_day
         end stddev_hematocrit_3rd_day,
         case
            when (avg_hematocrit_1st_day is null)     and
(avg_hematocrit_2nd_day is null)     and (avg_hematocrit_3rd_day is not null)
then
               avg_hematocrit_3rd_day
            when (avg_hematocrit_1st_day is null)     and
(avg_hematocrit_2nd_day is not null) and (avg_hematocrit_3rd_day is null)
then
               avg_hematocrit_2nd_day
            else
               avg_hematocrit_1st_day
         end avg_hematocrit_1st_day,
         case
            when (avg_hematocrit_1st_day is null)     and
(avg_hematocrit_2nd_day is null)     and (avg_hematocrit_3rd_day is not null)
then
               avg_hematocrit_3rd_day
```

```
            when (avg_hematocrit_1st_day is not null) and
(avg_hematocrit_2nd_day is null)     and (avg_hematocrit_3rd_day is null)
then
               avg_hematocrit_1st_day
            when (avg_hematocrit_1st_day is not null) and
(avg_hematocrit_2nd_day is null)     and (avg_hematocrit_3rd_day is not null)
then
               (avg_hematocrit_1st_day + avg_hematocrit_3rd_day) / 2
            else
               avg_hematocrit_2nd_day
         end avg_hematocrit_2nd_day,
         case
            when (avg_hematocrit_1st_day is not null) and
(avg_hematocrit_2nd_day is null)     and (avg_hematocrit_3rd_day is null)
then
               avg_hematocrit_1st_day
            when (avg_hematocrit_1st_day is null)     and
(avg_hematocrit_2nd_day is not null) and (avg_hematocrit_3rd_day is null)
then
               avg_hematocrit_2nd_day
            else
               avg_hematocrit_3rd_day
         end avg_hematocrit_3rd_day
    from RawData
),
--
--8.Glucose (min/max/average/SD)
--  a.If 2 days of min/max/average/SD values are missing, replace them with
--    min/max/average/SD value that is present.
--  b.If the second day min/max/average/SD values are missing, replace them
--    with the average of the other 2 daysâ€™ min/max/average/SD values.
--    Otherwise, replace them with the min/max/average/SD values that are
--    closer time-wise.
--
Rule8Glucose as (
  select subject_id,
         case
            when (min_glucose_1st_day is null)     and (min_glucose_2nd_day
is null)     and (min_glucose_3rd_day is not null) then
               min_glucose_3rd_day
            when (min_glucose_1st_day is null)     and (min_glucose_2nd_day
is not null) and (min_glucose_3rd_day is null)     then
               min_glucose_2nd_day
            else
               min_glucose_1st_day
         end min_glucose_1st_day,
         case
            when (min_glucose_1st_day is null)     and (min_glucose_2nd_day
is null)     and (min_glucose_3rd_day is not null) then
               min_glucose_3rd_day
            when (min_glucose_1st_day is not null) and (min_glucose_2nd_day
is null)     and (min_glucose_3rd_day is null)     then
               min_glucose_1st_day
            when (min_glucose_1st_day is not null) and (min_glucose_2nd_day
is null)     and (min_glucose_3rd_day is not null) then
               (min_glucose_1st_day + min_glucose_3rd_day) / 2
            else
```

```
               min_glucose_2nd_day
         end min_glucose_2nd_day,
         case
            when (min_glucose_1st_day is not null) and (min_glucose_2nd_day
is null)     and (min_glucose_3rd_day is null)     then
               min_glucose_1st_day
            when (min_glucose_1st_day is null)     and (min_glucose_2nd_day
is not null) and (min_glucose_3rd_day is null)     then
               min_glucose_2nd_day
            else
               min_glucose_3rd_day
         end min_glucose_3rd_day,
         case
            when (max_glucose_1st_day is null)     and (max_glucose_2nd_day
is null)     and (max_glucose_3rd_day is not null) then
               max_glucose_3rd_day
            when (max_glucose_1st_day is null)     and (max_glucose_2nd_day
is not null) and (max_glucose_3rd_day is null)     then
               max_glucose_2nd_day
            else
               max_glucose_1st_day
         end max_glucose_1st_day,
         case
            when (max_glucose_1st_day is null)     and (max_glucose_2nd_day
is null)     and (max_glucose_3rd_day is not null) then
               max_glucose_3rd_day
            when (max_glucose_1st_day is not null) and (max_glucose_2nd_day
is null)     and (max_glucose_3rd_day is null)     then
               max_glucose_1st_day
            when (max_glucose_1st_day is not null) and (max_glucose_2nd_day
is null)     and (max_glucose_3rd_day is not null) then
               (max_glucose_1st_day + max_glucose_3rd_day) / 2
            else
               max_glucose_2nd_day
         end max_glucose_2nd_day,
         case
            when (max_glucose_1st_day is not null) and (max_glucose_2nd_day
is null)     and (max_glucose_3rd_day is null)     then
               max_glucose_1st_day
            when (max_glucose_1st_day is null)     and (max_glucose_2nd_day
is not null) and (max_glucose_3rd_day is null)     then
               max_glucose_2nd_day
            else
               max_glucose_3rd_day
         end max_glucose_3rd_day,
         case
            when (stddev_glucose_1st_day is null)     and
(stddev_glucose_2nd_day is null)     and (stddev_glucose_3rd_day is not null)
then
               stddev_glucose_3rd_day
            when (stddev_glucose_1st_day is null)     and
(stddev_glucose_2nd_day is not null) and (stddev_glucose_3rd_day is null)
then
               stddev_glucose_2nd_day
            else
               stddev_glucose_1st_day
         end stddev_glucose_1st_day,
```

```
         case
            when (stddev_glucose_1st_day is null)     and
(stddev_glucose_2nd_day is null)     and (stddev_glucose_3rd_day is not null)
then
               stddev_glucose_3rd_day
            when (stddev_glucose_1st_day is not null) and
(stddev_glucose_2nd_day is null)     and (stddev_glucose_3rd_day is null)
then
               stddev_glucose_1st_day
            when (stddev_glucose_1st_day is not null) and
(stddev_glucose_2nd_day is null)     and (stddev_glucose_3rd_day is not null)
then
               (stddev_glucose_1st_day + stddev_glucose_3rd_day) / 2
            else
               stddev_glucose_2nd_day
         end stddev_glucose_2nd_day,
         case
            when (stddev_glucose_1st_day is not null) and
(stddev_glucose_2nd_day is null)     and (stddev_glucose_3rd_day is null)
then
               stddev_glucose_1st_day
            when (stddev_glucose_1st_day is null)     and
(stddev_glucose_2nd_day is not null) and (stddev_glucose_3rd_day is null)
then
               stddev_glucose_2nd_day
            else
               stddev_glucose_3rd_day
         end stddev_glucose_3rd_day,
         case
            when (avg_glucose_1st_day is null)     and (avg_glucose_2nd_day
is null)     and (avg_glucose_3rd_day is not null) then
               avg_glucose_3rd_day
            when (avg_glucose_1st_day is null)     and (avg_glucose_2nd_day
is not null) and (avg_glucose_3rd_day is null)     then
               avg_glucose_2nd_day
            else
               avg_glucose_1st_day
         end avg_glucose_1st_day,
         case
            when (avg_glucose_1st_day is null)     and (avg_glucose_2nd_day
is null)     and (avg_glucose_3rd_day is not null) then
               avg_glucose_3rd_day
            when (avg_glucose_1st_day is not null) and (avg_glucose_2nd_day
is null)     and (avg_glucose_3rd_day is null)     then
               avg_glucose_1st_day
            when (avg_glucose_1st_day is not null) and (avg_glucose_2nd_day
is null)     and (avg_glucose_3rd_day is not null) then
               (avg_glucose_1st_day + avg_glucose_3rd_day) / 2
            else
               avg_glucose_2nd_day
         end avg_glucose_2nd_day,
         case
            when (avg_glucose_1st_day is not null) and (avg_glucose_2nd_day
is null)     and (avg_glucose_3rd_day is null)     then
               avg_glucose_1st_day
            when (avg_glucose_1st_day is null)     and (avg_glucose_2nd_day
is not null) and (avg_glucose_3rd_day is null)     then
```

```
               avg_glucose_2nd_day
            else
               avg_glucose_3rd_day
         end avg_glucose_3rd_day
    from RawData
),
--
--9.WBC (min/max/average/SD)
--  a.If 2 days of min/max/average/SD values are missing, replace them with
--    min/max/average/SD value that is present.
--  b.If the second day min/max/average/SD values are missing, replace them
--    with the average of the other 2 daysâ€™ min/max/average/SD values.
--    Otherwise, replace them with the min/max/average/SD values that are
--    closer time-wise.
--
Rule9WBC as (
  select subject_id,
         case
            when (min_wbc_1st_day is null)     and (min_wbc_2nd_day is null)
and (min_wbc_3rd_day is not null) then
               min_wbc_3rd_day
            when (min_wbc_1st_day is null)     and (min_wbc_2nd_day is not
null) and (min_wbc_3rd_day is null)     then
               min_wbc_2nd_day
            else
               min_wbc_1st_day
         end min_wbc_1st_day,
         case
            when (min_wbc_1st_day is null)     and (min_wbc_2nd_day is null)
and (min_wbc_3rd_day is not null) then
               min_wbc_3rd_day
            when (min_wbc_1st_day is not null) and (min_wbc_2nd_day is null)
and (min_wbc_3rd_day is null)     then
               min_wbc_1st_day
            when (min_wbc_1st_day is not null) and (min_wbc_2nd_day is null)
and (min_wbc_3rd_day is not null) then
               (min_wbc_1st_day + min_wbc_3rd_day) / 2
            else
               min_wbc_2nd_day
         end min_wbc_2nd_day,
         case
            when (min_wbc_1st_day is not null) and (min_wbc_2nd_day is null)
and (min_wbc_3rd_day is null)     then
               min_wbc_1st_day
            when (min_wbc_1st_day is null)     and (min_wbc_2nd_day is not
null) and (min_wbc_3rd_day is null)     then
               min_wbc_2nd_day
            else
               min_wbc_3rd_day
         end min_wbc_3rd_day,
         case
            when (max_wbc_1st_day is null)     and (max_wbc_2nd_day is null)
and (max_wbc_3rd_day is not null) then
               max_wbc_3rd_day
            when (max_wbc_1st_day is null)     and (max_wbc_2nd_day is not
null) and (max_wbc_3rd_day is null)     then
               max_wbc_2nd_day
```

```
            else
               max_wbc_1st_day
         end max_wbc_1st_day,
         case
            when (max_wbc_1st_day is null)     and (max_wbc_2nd_day is null)
and (max_wbc_3rd_day is not null) then
               max_wbc_3rd_day
            when (max_wbc_1st_day is not null) and (max_wbc_2nd_day is null)
and (max_wbc_3rd_day is null)     then
               max_wbc_1st_day
            when (max_wbc_1st_day is not null) and (max_wbc_2nd_day is null)
and (max_wbc_3rd_day is not null) then
               (max_wbc_1st_day + max_wbc_3rd_day) / 2
            else
               max_wbc_2nd_day
         end max_wbc_2nd_day,
         case
            when (max_wbc_1st_day is not null) and (max_wbc_2nd_day is null)
and (max_wbc_3rd_day is null)     then
               max_wbc_1st_day
            when (max_wbc_1st_day is null)     and (max_wbc_2nd_day is not
null) and (max_wbc_3rd_day is null)     then
               max_wbc_2nd_day
            else
               max_wbc_3rd_day
         end max_wbc_3rd_day,
         case
            when (stddev_wbc_1st_day is null)     and (stddev_wbc_2nd_day is
null)     and (stddev_wbc_3rd_day is not null) then
               stddev_wbc_3rd_day
            when (stddev_wbc_1st_day is null)     and (stddev_wbc_2nd_day is
not null) and (stddev_wbc_3rd_day is null)     then
               stddev_wbc_2nd_day
            else
               stddev_wbc_1st_day
         end stddev_wbc_1st_day,
         case
            when (stddev_wbc_1st_day is null)     and (stddev_wbc_2nd_day is
null)     and (stddev_wbc_3rd_day is not null) then
               stddev_wbc_3rd_day
            when (stddev_wbc_1st_day is not null) and (stddev_wbc_2nd_day is
null)     and (stddev_wbc_3rd_day is null)     then
               stddev_wbc_1st_day
            when (stddev_wbc_1st_day is not null) and (stddev_wbc_2nd_day is
null)     and (stddev_wbc_3rd_day is not null) then
               (stddev_wbc_1st_day + stddev_wbc_3rd_day) / 2
            else
               stddev_wbc_2nd_day
         end stddev_wbc_2nd_day,
         case
            when (stddev_wbc_1st_day is not null) and (stddev_wbc_2nd_day is
null)     and (stddev_wbc_3rd_day is null)     then
               stddev_wbc_1st_day
            when (stddev_wbc_1st_day is null)     and (stddev_wbc_2nd_day is
not null) and (stddev_wbc_3rd_day is null)     then
               stddev_wbc_2nd_day
            else
```

```
               stddev_wbc_3rd_day
         end stddev_wbc_3rd_day,
         case
            when (avg_wbc_1st_day is null)     and (avg_wbc_2nd_day is null)
and (avg_wbc_3rd_day is not null) then
               avg_wbc_3rd_day
            when (avg_wbc_1st_day is null)     and (avg_wbc_2nd_day is not
null) and (avg_wbc_3rd_day is null)     then
               avg_wbc_2nd_day
            else
               avg_wbc_1st_day
         end avg_wbc_1st_day,
         case
            when (avg_wbc_1st_day is null)     and (avg_wbc_2nd_day is null)
and (avg_wbc_3rd_day is not null) then
               avg_wbc_3rd_day
            when (avg_wbc_1st_day is not null) and (avg_wbc_2nd_day is null)
and (avg_wbc_3rd_day is null)     then
               avg_wbc_1st_day
            when (avg_wbc_1st_day is not null) and (avg_wbc_2nd_day is null)
and (avg_wbc_3rd_day is not null) then
               (avg_wbc_1st_day + avg_wbc_3rd_day) / 2
            else
               avg_wbc_2nd_day
         end avg_wbc_2nd_day,
         case
            when (avg_wbc_1st_day is not null) and (avg_wbc_2nd_day is null)
and (avg_wbc_3rd_day is null)     then
               avg_wbc_1st_day
            when (avg_wbc_1st_day is null)     and (avg_wbc_2nd_day is not
null) and (avg_wbc_3rd_day is null)     then
               avg_wbc_2nd_day
            else
               avg_wbc_3rd_day
         end avg_wbc_3rd_day
    from RawData
),
--
--10.Potassium (min/max/average/SD)
--  a.If 2 days of min/max/average/SD values are missing, replace them with
--    min/max/average/SD value that is present.
--  b.If the second day min/max/average/SD values are missing, replace them
--    with the average of the other 2 daysâ€™ min/max/average/SD values.
--    Otherwise, replace them with the min/max/average/SD values that are
--    closer time-wise.
--
Rule10Potassium as (
  select subject_id,
         case
            when (min_potassium_1st_day is null)     and
(min_potassium_2nd_day is null)     and (min_potassium_3rd_day is not null)
then
               min_potassium_3rd_day
            when (min_potassium_1st_day is null)     and
(min_potassium_2nd_day is not null) and (min_potassium_3rd_day is null)
then
               min_potassium_2nd_day
```

```
            else
               min_potassium_1st_day
         end min_potassium_1st_day,
         case
            when (min_potassium_1st_day is null)     and
(min_potassium_2nd_day is null)     and (min_potassium_3rd_day is not null)
then
               min_potassium_3rd_day
            when (min_potassium_1st_day is not null) and
(min_potassium_2nd_day is null)     and (min_potassium_3rd_day is null)
then
               min_potassium_1st_day
            when (min_potassium_1st_day is not null) and
(min_potassium_2nd_day is null)     and (min_potassium_3rd_day is not null)
then
               (min_potassium_1st_day + min_potassium_3rd_day) / 2
            else
               min_potassium_2nd_day
         end min_potassium_2nd_day,
         case
            when (min_potassium_1st_day is not null) and
(min_potassium_2nd_day is null)     and (min_potassium_3rd_day is null)
then
               min_potassium_1st_day
            when (min_potassium_1st_day is null)     and
(min_potassium_2nd_day is not null) and (min_potassium_3rd_day is null)
then
               min_potassium_2nd_day
            else
               min_potassium_3rd_day
         end min_potassium_3rd_day,
         case
            when (max_potassium_1st_day is null)     and
(max_potassium_2nd_day is null)     and (max_potassium_3rd_day is not null)
then
               max_potassium_3rd_day
            when (max_potassium_1st_day is null)     and
(max_potassium_2nd_day is not null) and (max_potassium_3rd_day is null)
then
               max_potassium_2nd_day
            else
               max_potassium_1st_day
         end max_potassium_1st_day,
         case
            when (max_potassium_1st_day is null)     and
(max_potassium_2nd_day is null)     and (max_potassium_3rd_day is not null)
then
               max_potassium_3rd_day
            when (max_potassium_1st_day is not null) and
(max_potassium_2nd_day is null)     and (max_potassium_3rd_day is null)
then
               max_potassium_1st_day
            when (max_potassium_1st_day is not null) and
(max_potassium_2nd_day is null)     and (max_potassium_3rd_day is not null)
then
               (max_potassium_1st_day + max_potassium_3rd_day) / 2
            else
```

```
               max_potassium_2nd_day
         end max_potassium_2nd_day,
         case
            when (max_potassium_1st_day is not null) and
(max_potassium_2nd_day is null)     and (max_potassium_3rd_day is null)
then
               max_potassium_1st_day
            when (max_potassium_1st_day is null)     and
(max_potassium_2nd_day is not null) and (max_potassium_3rd_day is null)
then
               max_potassium_2nd_day
            else
               max_potassium_3rd_day
         end max_potassium_3rd_day,
         case
            when (stddev_potassium_1st_day is null)     and
(stddev_potassium_2nd_day is null)     and (stddev_potassium_3rd_day is not
null) then
               stddev_potassium_3rd_day
            when (stddev_potassium_1st_day is null)     and
(stddev_potassium_2nd_day is not null) and (stddev_potassium_3rd_day is null)
then
               stddev_potassium_2nd_day
            else
               stddev_potassium_1st_day
         end stddev_potassium_1st_day,
         case
            when (stddev_potassium_1st_day is null)     and
(stddev_potassium_2nd_day is null)     and (stddev_potassium_3rd_day is not
null) then
               stddev_potassium_3rd_day
            when (stddev_potassium_1st_day is not null) and
(stddev_potassium_2nd_day is null)     and (stddev_potassium_3rd_day is null)
then
               stddev_potassium_1st_day
            when (stddev_potassium_1st_day is not null) and
(stddev_potassium_2nd_day is null)     and (stddev_potassium_3rd_day is not
null) then
               (stddev_potassium_1st_day + stddev_potassium_3rd_day) / 2
            else
               stddev_potassium_2nd_day
         end stddev_potassium_2nd_day,
         case
            when (stddev_potassium_1st_day is not null) and
(stddev_potassium_2nd_day is null)     and (stddev_potassium_3rd_day is null)
then
               stddev_potassium_1st_day
            when (stddev_potassium_1st_day is null)     and
(stddev_potassium_2nd_day is not null) and (stddev_potassium_3rd_day is null)
then
               stddev_potassium_2nd_day
            else
               stddev_potassium_3rd_day
         end stddev_potassium_3rd_day,
         case
```

```
            when (avg_potassium_1st_day is null)     and
(avg_potassium_2nd_day is null)     and (avg_potassium_3rd_day is not null)
then
               avg_potassium_3rd_day
            when (avg_potassium_1st_day is null)     and
(avg_potassium_2nd_day is not null) and (avg_potassium_3rd_day is null)
then
               avg_potassium_2nd_day
            else
               avg_potassium_1st_day
         end avg_potassium_1st_day,
         case
            when (avg_potassium_1st_day is null)     and
(avg_potassium_2nd_day is null)     and (avg_potassium_3rd_day is not null)
then
               avg_potassium_3rd_day
            when (avg_potassium_1st_day is not null) and
(avg_potassium_2nd_day is null)     and (avg_potassium_3rd_day is null)
then
               avg_potassium_1st_day
            when (avg_potassium_1st_day is not null) and
(avg_potassium_2nd_day is null)     and (avg_potassium_3rd_day is not null)
then
               (avg_potassium_1st_day + avg_potassium_3rd_day) / 2
            else
               avg_potassium_2nd_day
         end avg_potassium_2nd_day,
         case
            when (avg_potassium_1st_day is not null) and
(avg_potassium_2nd_day is null)     and (avg_potassium_3rd_day is null)
then
               avg_potassium_1st_day
            when (avg_potassium_1st_day is null)     and
(avg_potassium_2nd_day is not null) and (avg_potassium_3rd_day is null)
then
               avg_potassium_2nd_day
            else
               avg_potassium_3rd_day
         end avg_potassium_3rd_day
    from RawData
),
--
--11.BUN (max)
--  a.If 2 values are missing, replace them with the level that is present.
--  b.If the middle value is missing, replace it with the average of the
other
--    2 values.  Otherwise, replace it with the value thatâ€™s closer time-
wise.
--
Rule11BUN as (
  select subject_id,
         case
            when (max_bun_1st_day is null)     and (max_bun_2nd_day is null)
and (max_bun_3rd_day is not null) then
               max_bun_3rd_day
            when (max_bun_1st_day is null)     and (max_bun_2nd_day is not
null) and (max_bun_3rd_day is null)     then
```

```
               max_bun_2nd_day
            else
               max_bun_1st_day
         end max_bun_1st_day,
         case
            when (max_bun_1st_day is null)     and (max_bun_2nd_day is null)
and (max_bun_3rd_day is not null) then
               max_bun_3rd_day
            when (max_bun_1st_day is not null) and (max_bun_2nd_day is null)
and (max_bun_3rd_day is null)     then
               max_bun_1st_day
            when (max_bun_1st_day is not null) and (max_bun_2nd_day is null)
and (max_bun_3rd_day is not null) then
               (max_bun_1st_day + max_bun_3rd_day) / 2
            else
               max_bun_2nd_day
         end max_bun_2nd_day,
         case
            when (max_bun_1st_day is not null) and (max_bun_2nd_day is null)
and (max_bun_3rd_day is null)     then
               max_bun_1st_day
            when (max_bun_1st_day is null)     and (max_bun_2nd_day is not
null) and (max_bun_3rd_day is null)     then
               max_bun_2nd_day
            else
               max_bun_3rd_day
         end max_bun_3rd_day
    from RawData
),
--
--12. Temperature (min/max/average/SD):
--
--  If the second day min/max/average/SD values are missing, replace them
with
--  the average of the other 2 daysâ€™ min/max/ average/SD values.
Otherwise,
--  replace them with the min/max/average/SD values that are closer time-
wise.
--
Rule12Temp as (
  select subject_id,
         min_temp_1st_day,
         case
            when (min_temp_1st_day is not null) and (min_temp_2nd_day is
null)     and (min_temp_3rd_day is not null) then
               (min_temp_1st_day + min_temp_3rd_day) / 2
            else
               min_temp_2nd_day
         end min_temp_2nd_day,
         min_temp_3rd_day,
         max_temp_1st_day,
         case
            when (max_temp_1st_day is not null) and (max_temp_2nd_day is
null)     and (max_temp_3rd_day is not null) then
               (max_temp_1st_day + max_temp_3rd_day) / 2
            else
               max_temp_2nd_day
```

```
         end max_temp_2nd_day,
         max_temp_3rd_day,
         stddev_temp_1st_day,
         case
            when (stddev_temp_1st_day is not null) and (stddev_temp_2nd_day
is null)     and (stddev_temp_3rd_day is not null) then
               (stddev_temp_1st_day + stddev_temp_3rd_day) / 2
            else
               stddev_temp_2nd_day
         end stddev_temp_2nd_day,
         stddev_temp_3rd_day,
         avg_temp_1st_day,
         case
            when (avg_temp_1st_day is not null) and (avg_temp_2nd_day is
null)     and (avg_temp_3rd_day is not null) then
               (avg_temp_1st_day + avg_temp_3rd_day) / 2
            else
               avg_temp_2nd_day
         end avg_temp_2nd_day,
         avg_temp_3rd_day
    from RawData
),
--
--13. GCS (min), Day 1 only: Replace missing value with 15.
--
Rule13GCS as (
 select subject_id,
        nvl(min_gcs_1st_day, 15)min_gcs_1st_day
   from RawData
),
--
--14. Systolic Blood Pressure (min/max/average/SD):
--
--  If the second day min/max/ average/SD values are missing, replace them
--  with the average of the other 2 daysâ€™ min/max/ average/SD values.
--  Otherwise, replace them with the min/max/ average/SD values that are
--  closer time-wise.
--
Rule14SysBP as (
  select subject_id,
         min_sysbp_1st_day,
         case
            when (min_sysbp_1st_day is not null) and (min_sysbp_2nd_day is
null)     and (min_sysbp_3rd_day is not null) then
               (min_sysbp_1st_day + min_sysbp_3rd_day) / 2
            else
               min_sysbp_2nd_day
         end min_sysbp_2nd_day,
         min_sysbp_3rd_day,
         max_sysbp_1st_day,
         case
            when (max_sysbp_1st_day is not null) and (max_sysbp_2nd_day is
null)     and (max_sysbp_3rd_day is not null) then
               (max_sysbp_1st_day + max_sysbp_3rd_day) / 2
            else
               max_sysbp_2nd_day
         end max_sysbp_2nd_day,
```

```
         max_sysbp_3rd_day,
         stddev_sysbp_1st_day,
         case
            when (stddev_sysbp_1st_day is not null) and (stddev_sysbp_2nd_day
is null)     and (stddev_sysbp_3rd_day is not null) then
               (stddev_sysbp_1st_day + stddev_sysbp_3rd_day) / 2
            else
               stddev_sysbp_2nd_day
         end stddev_sysbp_2nd_day,
         stddev_sysbp_3rd_day,
         avg_sysbp_1st_day,
         case
            when (avg_sysbp_1st_day is not null) and (avg_sysbp_2nd_day is
null)     and (avg_sysbp_3rd_day is not null) then
               (avg_sysbp_1st_day + avg_sysbp_3rd_day) / 2
            else
               avg_sysbp_2nd_day
         end avg_sysbp_2nd_day,
         avg_sysbp_3rd_day
    from RawData
),
--
--15. Bicarbonate (min/max/average/SD)
--  a.Delete 2 patients without min/max/average/SD values for all the three
days.
--  b.If 2 days of min/max/average/SD values are missing, replace them
--    with 24/24/24/0.
--  c.If the second day min/max/average/SD values are missing, replace them
--    with the average of the other 2 daysâ€™ min/max/average/SD values.
--    Otherwise, replace them 24/24/24/0.
--
Rule15Bicarbonate as (
 select subject_id,
         case
            when (min_bicarbonate_1st_day is not null) then
               min_bicarbonate_1st_day
            else
               24
         end min_bicarbonate_1st_day,
         case
            when (min_bicarbonate_2nd_day is not null) then
               min_bicarbonate_2nd_day
            when (min_bicarbonate_1st_day is not null) and
(min_bicarbonate_2nd_day is null)     and (min_bicarbonate_3rd_day is not
null) then
               (min_bicarbonate_1st_day + min_bicarbonate_3rd_day) / 2
            else
               24
         end min_bicarbonate_2nd_day,
         case
            when (min_bicarbonate_3rd_day is not null) then
               min_bicarbonate_3rd_day
            else
               24
         end min_bicarbonate_3rd_day,
         case
            when (max_bicarbonate_1st_day is not null) then
```

```
               max_bicarbonate_1st_day
            else
               24
         end max_bicarbonate_1st_day,
         case
            when (max_bicarbonate_2nd_day is not null) then
               max_bicarbonate_2nd_day
            when (max_bicarbonate_1st_day is not null) and
(max_bicarbonate_2nd_day is null)     and (max_bicarbonate_3rd_day is not
null) then
               (max_bicarbonate_1st_day + max_bicarbonate_3rd_day) / 2
            else
               24
         end max_bicarbonate_2nd_day,
         case
            when (max_bicarbonate_3rd_day is not null) then
               max_bicarbonate_3rd_day
            else
               24
         end max_bicarbonate_3rd_day,
         case
            when (stddev_bicarbonate_1st_day is not null) then
               stddev_bicarbonate_1st_day
            else
               0
         end stddev_bicarbonate_1st_day,
         case
            when (stddev_bicarbonate_2nd_day is not null) then
               stddev_bicarbonate_2nd_day
            when (stddev_bicarbonate_1st_day is not null) and
(stddev_bicarbonate_2nd_day is null)     and (stddev_bicarbonate_3rd_day is
not null) then
               (stddev_bicarbonate_1st_day + stddev_bicarbonate_3rd_day) / 2
            else
               0
         end stddev_bicarbonate_2nd_day,
         case
            when (stddev_bicarbonate_3rd_day is not null) then
               stddev_bicarbonate_3rd_day
            else
               0
         end stddev_bicarbonate_3rd_day,
         case
            when (avg_bicarbonate_1st_day is not null) then
               avg_bicarbonate_1st_day
            else
               24
         end avg_bicarbonate_1st_day,
         case
            when (avg_bicarbonate_2nd_day is not null) then
               avg_bicarbonate_2nd_day
            when (avg_bicarbonate_1st_day is not null) and
(avg_bicarbonate_2nd_day is null)     and (avg_bicarbonate_3rd_day is not
null) then
               (avg_bicarbonate_1st_day + avg_bicarbonate_3rd_day) / 2
            else
               24
```

```
          end avg_bicarbonate_2nd_day,
          case
             when (avg_bicarbonate_3rd_day is not null) then
                avg_bicarbonate_3rd_day
             else
                24
          end avg_bicarbonate_3rd_day
     from RawData
)
select d.subject_id, d.expire_flg, d.age, d.sex,
       d.intime, d.outtime, d.los,
       r2.max_bili_1st_day,  r2.max_bili_2nd_day,  r2.max_bili_3rd_day,
       r3.max_creat_1st_day, r3.max_creat_2nd_day, r3.max_creat_3rd_day,
       r4.min_hr_1st_day,    r4.max_hr_1st_day,    r4.stddev_hr_1st_day,
r4.avg_hr_1st_day,
       r4.min_hr_2nd_day,    r4.max_hr_2nd_day,    r4.stddev_hr_2nd_day,
r4.avg_hr_2nd_day,
       r4.min_hr_3rd_day,    r4.max_hr_3rd_day,    r4.stddev_hr_3rd_day,
r4.avg_hr_3rd_day,
       r5.min_sodium_1st_day,r5.max_sodium_1st_day,r5.stddev_sodium_1st_day,
r5.avg_sodium_1st_day,
       r5.min_sodium_2nd_day,r5.max_sodium_2nd_day,r5.stddev_sodium_2nd_day,
r5.avg_sodium_2nd_day,
       r5.min_sodium_3rd_day,r5.max_sodium_3rd_day,r5.stddev_sodium_3rd_day,
r5.avg_sodium_3rd_day,
       r14.min_sysbp_1st_day,r14.max_sysbp_1st_day,r14.stddev_sysbp_1st_day,
r14.avg_sysbp_1st_day,
       r14.min_sysbp_2nd_day,r14.max_sysbp_2nd_day,r14.stddev_sysbp_2nd_day,
r14.avg_sysbp_2nd_day,
       r14.min_sysbp_3rd_day,r14.max_sysbp_3rd_day,r14.stddev_sysbp_3rd_day,
r14.avg_sysbp_3rd_day,
       r6.max_resp_1st_day,   r6.max_resp_2nd_day,  r6.max_resp_3rd_day,
       r7.min_hematocrit_1st_day, r7.max_hematocrit_1st_day,
r7.stddev_hematocrit_1st_day, r7.avg_hematocrit_1st_day,
       r7.min_hematocrit_2nd_day, r7.max_hematocrit_2nd_day,
r7.stddev_hematocrit_2nd_day, r7.avg_hematocrit_2nd_day,
       r7.min_hematocrit_3rd_day, r7.max_hematocrit_3rd_day,
r7.stddev_hematocrit_3rd_day, r7.avg_hematocrit_3rd_day,
       r8.min_glucose_1st_day, r8.max_glucose_1st_day,
r8.stddev_glucose_1st_day, r8.avg_glucose_1st_day,
       r8.min_glucose_2nd_day, r8.max_glucose_2nd_day,
r8.stddev_glucose_2nd_day, r8.avg_glucose_2nd_day,
       r8.min_glucose_3rd_day, r8.max_glucose_3rd_day,
r8.stddev_glucose_3rd_day, r8.avg_glucose_3rd_day,
       r9.min_wbc_1st_day,     r9.max_wbc_1st_day,     r9.stddev_wbc_1st_day,
r9.avg_wbc_1st_day,
       r9.min_wbc_2nd_day,     r9.max_wbc_2nd_day,     r9.stddev_wbc_2nd_day,
r9.avg_wbc_2nd_day,
       r9.min_wbc_3rd_day,     r9.max_wbc_3rd_day,     r9.stddev_wbc_3rd_day,
r9.avg_wbc_3rd_day,
       r10.min_potassium_1st_day, r10.max_potassium_1st_day,
r10.stddev_potassium_1st_day, r10.avg_potassium_1st_day,
       r10.min_potassium_2nd_day, r10.max_potassium_2nd_day,
r10.stddev_potassium_2nd_day, r10.avg_potassium_2nd_day,
       r10.min_potassium_3rd_day, r10.max_potassium_3rd_day,
r10.stddev_potassium_3rd_day, r10.avg_potassium_3rd_day,
       r11.max_bun_1st_day,    r11.max_bun_2nd_day,    r11.max_bun_3rd_day,
```

```
       r13.min_gcs_1st_day,
       r12.min_temp_1st_day,   r12.max_temp_1st_day,
r12.stddev_temp_1st_day,   r12.avg_temp_1st_day,
       r12.min_temp_2nd_day,   r12.max_temp_2nd_day,
r12.stddev_temp_2nd_day,   r12.avg_temp_2nd_day,
       r12.min_temp_3rd_day,   r12.max_temp_3rd_day,
r12.stddev_temp_3rd_day,   r12.avg_temp_3rd_day,
       r15.min_bicarbonate_1st_day, r15.max_bicarbonate_1st_day,
r15.stddev_bicarbonate_1st_day, r15.avg_bicarbonate_1st_day,
       r15.min_bicarbonate_2nd_day, r15.max_bicarbonate_2nd_day,
r15.stddev_bicarbonate_2nd_day, r15.avg_bicarbonate_2nd_day,
       r15.min_bicarbonate_3rd_day, r15.max_bicarbonate_3rd_day,
r15.stddev_bicarbonate_3rd_day, r15.avg_bicarbonate_3rd_day,
       d.in_1st_day, d.out_1st_day,
       d.in_2nd_day, d.out_2nd_day,
       d.in_3rd_day, d.out_3rd_day
  from RawData d,
       Rule2Bili r2,
       Rule3Creat r3,
       Rule4HR r4,
       Rule5Sodium r5,
       Rule6Resp r6,
       Rule7Hemat r7,
       Rule8Glucose r8,
       Rule9WBC r9,
       Rule10Potassium r10,
       Rule11BUN r11,
       Rule12Temp r12,
       Rule13GCS r13,
       Rule14SysBP r14,
       Rule15Bicarbonate r15
   where r2.subject_id = d.subject_id
     and r3.subject_id = d.subject_id
     and r4.subject_id = d.subject_id
     and r5.subject_id = d.subject_id
     and r6.subject_id = d.subject_id
     and r7.subject_id = d.subject_id
     and r8.subject_id = d.subject_id
     and r9.subject_id = d.subject_id
     and r10.subject_id = d.subject_id
     and r11.subject_id = d.subject_id
     and r12.subject_id = d.subject_id
     and r13.subject_id = d.subject_id
     and r14.subject_id = d.subject_id
     and r15.subject_id = d.subject_id;
```

Appendix B. Bayesian Network Model to Predict Hospital Mortality using Tenfold Cross-Validation (Weka 3.5.7)

Network structure (nodes followed by parents)
EXPIRE_FLG(2):
AGE(1): EXPIRE_FLG
MAX_BILI_1ST_DAY(2): EXPIRE_FLG
MAX_BILI_2ND_DAY(2): EXPIRE_FLG
MAX_BILI_3RD_DAY(2): EXPIRE_FLG
MAX_HR_1ST_DAY(1): EXPIRE_FLG
MAX_HR_2ND_DAY(1): EXPIRE_FLG
MAX_HR_3RD_DAY(1): EXPIRE_FLG
MAX_SODIUM_1ST_DAY(1): EXPIRE_FLG
MAX_SODIUM_2ND_DAY(1): EXPIRE_FLG
MAX_SODIUM_3RD_DAY(1): EXPIRE_FLG
MIN_SYSBP_1ST_DAY(1): EXPIRE_FLG
MIN_SYSBP_2ND_DAY(2): EXPIRE_FLG
MIN_SYSBP_3RD_DAY(2): EXPIRE_FLG
MIN_WBC_1ST_DAY(2): EXPIRE_FLG
MIN_WBC_2ND_DAY(2): EXPIRE_FLG
MIN_WBC_3RD_DAY(3): EXPIRE_FLG
MIN_POTASSIUM_1ST_DAY(1): EXPIRE_FLG
MIN_POTASSIUM_2ND_DAY(1): EXPIRE_FLG
MIN_POTASSIUM_3RD_DAY(1): EXPIRE_FLG
MAX_BUN_1ST_DAY(1): EXPIRE_FLG
MAX_BUN_2ND_DAY(2): EXPIRE_FLG
MAX_BUN_3RD_DAY(2): EXPIRE_FLG
MIN_GCS_1ST_DAY(2): EXPIRE_FLG
MAX_TEMP_1ST_DAY(1): EXPIRE_FLG
MAX_TEMP_2ND_DAY(1): EXPIRE_FLG
MAX_TEMP_3RD_DAY(1): EXPIRE_FLG
MIN_BICARBONATE_1ST_DAY(1): EXPIRE_FLG
MIN_BICARBONATE_2ND_DAY(1): EXPIRE_FLG
MIN_BICARBONATE_3RD_DAY(1): EXPIRE_FLG
OUT_1ST_DAY(1): EXPIRE_FLG
OUT_2ND_DAY(2): EXPIRE_FLG
OUT_3RD_DAY(2): EXPIRE_FLG

LogScore Bayes: -6340.86525861412
LogScore BDeu: -6366.959980500857
LogScore MDL: -6379.0470996560925
LogScore ENTROPY: -6279.192390839399
LogScore AIC: -6308.192390839399

Correctly Classified Instances	667	68.1307 %
Incorrectly Classified Instances	312	31.8693 %
Kappa statistic	0.2192	
Mean absolute error	0.3587	

Root mean squared error	0.4729
Relative absolute error	82.7199 %
Root relative squared error	101.5712 %
Total Number of Instances	979

TP Rate	FP Rate	Precision	Recall	F-Measure	ROC Area	Class
0.817	0.611	0.742	0.817	0.778	0.682	N
0.389	0.183	0.498	0.389	0.437	0.682	Y

Appendix C. Naïve Bayes Model to Predict Hospital Mortality using Tenfold Cross-Validation (Weka 3.5.7)

Class N: Prior probability = 0.75
Day1_In: Normal Distribution. Mean = 3476.8456 StandardDev = 3990.0661 WeightSum = 1121 Precision = 26.3618304387292
Day1_Out: Normal Distribution. Mean = 1380.4209 StandardDev = 1529.7834 WeightSum = 1121 Precision = 18.041666666666668
Day2_In: Normal Distribution. Mean = 3402.7439 StandardDev = 2685.4955 WeightSum = 1121 Precision = 27.937100850978187
Day2_Out: Normal Distribution. Mean = 2116.9527 StandardDev = 1474.3805 WeightSum = 1121 Precision = 14.913083257090577
Day3_In: Normal Distribution. Mean = 2316.6936 StandardDev = 2160.9165 WeightSum = 1121 Precision = 23.395042858243453
Day3_Out: Normal Distribution. Mean = 1997.9051 StandardDev = 1428.1725 WeightSum = 1121 Precision = 19.74827245804541
Age: Normal Distribution. Mean = 67.9885 StandardDev = 15.9864 WeightSum = 1121 Precision = 0.06696414950419527
Sex: Normal Distribution. Mean = 0.5593 StandardDev = 0.4965 WeightSum = 1121 Precision = 1.0
SAPS: Normal Distribution. Mean = 15.645 StandardDev = 5.2134 WeightSum = 1121 Precision = 1.0

Class Y: Prior probability = 0.25
Day1_In: Normal Distribution. Mean = 3353.4386 StandardDev = 4357.0549 WeightSum = 370 Precision = 26.3618304387292
Day1_Out: Normal Distribution. Mean = 919.2473 StandardDev = 1207.536 WeightSum = 370 Precision = 18.041666666666668
Day2_In: Normal Distribution. Mean = 4551.4823 StandardDev = 3985.3966 WeightSum = 370 Precision = 27.937100850978187
Day2_Out: Normal Distribution. Mean = 1525.689 StandardDev = 1530.8948 WeightSum = 370 Precision = 14.913083257090577
Day3_In: Normal Distribution. Mean = 3178.4379 StandardDev = 2629.7816 WeightSum = 370 Precision = 23.395042858243453
Day3_Out: Normal Distribution. Mean = 1524.8335 StandardDev = 1289.8183 WeightSum = 370 Precision = 19.74827245804541
Age: Normal Distribution. Mean = 71.8786 StandardDev = 15.5724 WeightSum = 370 Precision = 0.06696414950419527
Sex: Normal Distribution. Mean = 0.5459 StandardDev = 0.4979 WeightSum = 370 Precision = 1.0
SAPS: Normal Distribution. Mean = 18.3135 StandardDev = 5.1574 WeightSum = 370 Precision = 1.0

Class N: Prior probability = 0.68
AGE: Normal Distribution. Mean = 73.1266 StandardDev = 29.8511 WeightSum = 668 Precision = 2.7761194029850746
MAX_BILI_1ST_DAY: Normal Distribution. Mean = 1.2636 StandardDev = 3.6117 WeightSum = 668 Precision = 0.7314606741573033
MAX_BILI_2ND_DAY: Normal Distribution. Mean = 1.1886 StandardDev = 3.5429 WeightSum = 668 Precision = 0.6694736842105263
MAX_BILI_3RD_DAY: Normal Distribution. Mean = 1.2842 StandardDev = 3.7578 WeightSum = 668 Precision = 0.8093023255813953
MAX_HR_1ST_DAY: Normal Distribution. Mean = 104.8686 StandardDev = 21.9376 WeightSum = 668 Precision = 1.4375
MAX_HR_2ND_DAY: Normal Distribution. Mean = 101.3868 StandardDev = 21.7548 WeightSum = 668 Precision = 1.4375
MAX_HR_3RD_DAY: Normal Distribution. Mean = 101.1086 StandardDev = 20.7177 WeightSum = 668 Precision = 1.3831775700934579
MAX_SODIUM_1ST_DAY: Normal Distribution. Mean = 139.5247 StandardDev = 5.0459 WeightSum = 668 Precision = 1.34375
MAX_SODIUM_2ND_DAY: Normal Distribution. Mean = 139.6047 StandardDev = 4.6469 WeightSum = 668 Precision = 0.6578947368421053
MAX_SODIUM_3RD_DAY: Normal Distribution. Mean = 139.8189 StandardDev = 4.8872 WeightSum = 668 Precision = 1.0
MIN_SYSBP_1ST_DAY: Normal Distribution. Mean = 91.3525 StandardDev = 16.8855 WeightSum = 668 Precision = 1.1649484536082475
MIN_SYSBP_2ND_DAY: Normal Distribution. Mean = 97.0265 StandardDev = 17.2594 WeightSum = 668 Precision = 1.1443298969072164
MIN_SYSBP_3RD_DAY: Normal Distribution. Mean = 99.5738 StandardDev = 17.4594 WeightSum = 668 Precision = 1.3786407766990292
MIN_WBC_1ST_DAY: Normal Distribution. Mean = 11.8063 StandardDev = 6.1696 WeightSum = 668 Precision = 0.3792
MIN_WBC_2ND_DAY: Normal Distribution. Mean = 11.7608 StandardDev = 5.365 WeightSum = 668 Precision = 0.42333333333333334
MIN_WBC_3RD_DAY: Normal Distribution. Mean = 11.4189 StandardDev = 5.3292 WeightSum = 668 Precision = 0.45903614457831327
MIN_POTASSIUM_1ST_DAY: Normal Distribution. Mean = 4.0287 StandardDev = 0.5769 WeightSum = 668 Precision = 0.11764705882352941
MIN_POTASSIUM_2ND_DAY: Normal Distribution. Mean = 3.9783 StandardDev = 0.538 WeightSum = 668 Precision = 0.10810810810810811
MIN_POTASSIUM_3RD_DAY: Normal Distribution. Mean = 3.9288 StandardDev = 0.4747 WeightSum = 668 Precision = 0.10967741935483871
MAX_BUN_1ST_DAY: Normal Distribution. Mean = 49.4843 StandardDev = 28.5867 WeightSum = 668 Precision = 1.5
MAX_BUN_2ND_DAY: Normal Distribution. Mean = 47.1404 StandardDev = 26.8195 WeightSum = 668 Precision = 1.1654676258992807
MAX_BUN_3RD_DAY: Normal Distribution. Mean = 45.248 StandardDev = 25.9049 WeightSum = 668 Precision = 1.2764227642276422
MIN_GCS_1ST_DAY: Normal Distribution. Mean = 10.3892 StandardDev = 4.5479 WeightSum = 668 Precision = 1.0
MAX_TEMP_1ST_DAY: Normal Distribution. Mean = 37.5639 StandardDev = 0.8753 WeightSum = 668

Precision = 0.10384615384615382
MAX_TEMP_2ND_DAY: Normal Distribution. Mean = 37.4802 StandardDev = 0.7806 WeightSum = 668 Precision = 0.11086956521739133
MAX_TEMP_3RD_DAY: Normal Distribution. Mean = 37.3857 StandardDev = 0.745 WeightSum = 668 Precision = 0.15000000000000013
MIN_BICARBONATE_1ST_DAY: Normal Distribution. Mean = 22.7115 StandardDev = 4.6051 WeightSum = 668 Precision = 1.103448275862069
MIN_BICARBONATE_2ND_DAY: Normal Distribution. Mean = 23.2721 StandardDev = 4.4531 WeightSum = 668 Precision = 0.8235294117647058
MIN_BICARBONATE_3RD_DAY: Normal Distribution. Mean = 23.6017 StandardDev = 4.3765 WeightSum = 668 Precision = 1.0384615384615385
OUT_1ST_DAY: Normal Distribution. Mean = 1007.1937 StandardDev = 965.292 WeightSum = 668 Precision = 17.14590747330961
OUT_2ND_DAY: Normal Distribution. Mean = 1914.055 StandardDev = 1300.6773 WeightSum = 668 Precision = 20.897435897435898
OUT_3RD_DAY: Normal Distribution. Mean = 1994.9639 StandardDev = 1285.7193 WeightSum = 668 Precision = 14.707057256990678

Class Y: Prior probability = 0.32
AGE: Normal Distribution. Mean = 78.7311 StandardDev = 34.4679 WeightSum = 311 Precision = 2.7761194029850746
MAX_BILI_1ST_DAY: Normal Distribution. Mean = 2.987 StandardDev = 7.6464 WeightSum = 311 Precision = 0.7314606741573033
MAX_BILI_2ND_DAY: Normal Distribution. Mean = 3.0546 StandardDev = 8.1274 WeightSum = 311 Precision = 0.6694736842105263
MAX_BILI_3RD_DAY: Normal Distribution. Mean = 3.1617 StandardDev = 8.2401 WeightSum = 311 Precision = 0.8093023255813953
MAX_HR_1ST_DAY: Normal Distribution. Mean = 107.7894 StandardDev = 24.7858 WeightSum = 311 Precision = 1.4375
MAX_HR_2ND_DAY: Normal Distribution. Mean = 104.7572 StandardDev = 23.7829 WeightSum = 311 Precision = 1.4375
MAX_HR_3RD_DAY: Normal Distribution. Mean = 104.8369 StandardDev = 24.0735 WeightSum = 311 Precision = 1.3831775700934579
MAX_SODIUM_1ST_DAY: Normal Distribution. Mean = 139.2834 StandardDev = 5.8926 WeightSum = 311 Precision = 1.34375
MAX_SODIUM_2ND_DAY: Normal Distribution. Mean = 139.4906 StandardDev = 5.4914 WeightSum = 311 Precision = 0.6578947368421053
MAX_SODIUM_3RD_DAY: Normal Distribution. Mean = 139.4437 StandardDev = 5.4857 WeightSum = 311 Precision = 1.0
MIN_SYSBP_1ST_DAY: Normal Distribution. Mean = 86.0638 StandardDev = 17.3366 WeightSum = 311 Precision = 1.1649484536082475
MIN_SYSBP_2ND_DAY: Normal Distribution. Mean = 91.0975 StandardDev = 17.7971 WeightSum = 311 Precision = 1.1443298969072164
MIN_SYSBP_3RD_DAY: Normal Distribution. Mean = 91.8148 StandardDev = 19.7577 WeightSum = 311 Precision = 1.3786407766990292
MIN_WBC_1ST_DAY: Normal Distribution. Mean = 13.4488 StandardDev = 9.5137 WeightSum = 311 Precision = 0.3792
MIN_WBC_2ND_DAY: Normal Distribution. Mean = 13.8734 StandardDev = 10.7895 WeightSum = 311

Precision = 0.4233333333333334
MIN_WBC_3RD_DAY: Normal Distribution. Mean = 13.653 StandardDev = 10.8276 WeightSum = 311 Precision = 0.45903614457831327
MIN_POTASSIUM_1ST_DAY: Normal Distribution. Mean = 4.0855 StandardDev = 0.5817 WeightSum = 311 Precision = 0.11764705882352941
MIN_POTASSIUM_2ND_DAY: Normal Distribution. Mean = 4.0257 StandardDev = 0.5706 WeightSum = 311 Precision = 0.10810810810810811
MIN_POTASSIUM_3RD_DAY: Normal Distribution. Mean = 3.9597 StandardDev = 0.5544 WeightSum = 311 Precision = 0.10967741935483871
MAX_BUN_1ST_DAY: Normal Distribution. Mean = 55.6977 StandardDev = 29.9635 WeightSum = 311 Precision = 1.5
MAX_BUN_2ND_DAY: Normal Distribution. Mean = 56.8456 StandardDev = 28.8885 WeightSum = 311 Precision = 1.1654676258992807
MAX_BUN_3RD_DAY: Normal Distribution. Mean = 57.398 StandardDev = 28.6635 WeightSum = 311 Precision = 1.2764227642276422
MIN_GCS_1ST_DAY: Normal Distribution. Mean = 9.4887 StandardDev = 4.3206 WeightSum = 311 Precision = 1.0
MAX_TEMP_1ST_DAY: Normal Distribution. Mean = 37.4885 StandardDev = 0.911 WeightSum = 311 Precision = 0.10384615384615382
MAX_TEMP_2ND_DAY: Normal Distribution. Mean = 37.3869 StandardDev = 0.8279 WeightSum = 311 Precision = 0.11086956521739133
MAX_TEMP_3RD_DAY: Normal Distribution. Mean = 37.3895 StandardDev = 0.8602 WeightSum = 311 Precision = 0.15000000000000013
MIN_BICARBONATE_1ST_DAY: Normal Distribution. Mean = 22.2641 StandardDev = 4.5883 WeightSum = 311 Precision = 1.103448275862069
MIN_BICARBONATE_2ND_DAY: Normal Distribution. Mean = 22.4736 StandardDev = 4.6945 WeightSum = 311 Precision = 0.8235294117647058
MIN_BICARBONATE_3RD_DAY: Normal Distribution. Mean = 22.6658 StandardDev = 4.5941 WeightSum = 311 Precision = 1.0384615384615385
OUT_1ST_DAY: Normal Distribution. Mean = 808.7796 StandardDev = 1027.1682 WeightSum = 311 Precision = 17.14590747330961
OUT_2ND_DAY: Normal Distribution. Mean = 1557.2285 StandardDev = 1566.2343 WeightSum = 311 Precision = 20.897435897435898
OUT_3RD_DAY: Normal Distribution. Mean = 1657.4523 StandardDev = 1279.0118 WeightSum = 311 Precision = 14.707057256990678

Correctly Classified Instances	704	71.9101 %
Incorrectly Classified Instances	275	28.0899 %
Kappa statistic	0.2374	
Mean absolute error	0.2906	
Root mean squared error	0.5018	
Relative absolute error	67.02 %	
Root relative squared error	107.7881 %	
Total Number of Instances	979	

TP Rate	FP Rate	Precision	Recall	F-Measure	ROC Area	Class
0.925	0.723	0.733	0.925	0.818	0.691	N
0.277	0.075	0.632	0.277	0.385	0.691	Y

Appendix D. Artificial Neural Network Model to Predict Hospital Mortality using Tenfold Cross-Validation (Weka 3.5.7)

Sigmoid Node 0
Inputs Weights
Threshold 2.5258119612290724
Node 2 -8.844468929277252
Node 3 10.691708054658712
Node 4 11.107198648382921
Node 5 -6.086167898616822
Node 6 -14.09548219333112
Node 7 -9.14185582586785
Node 8 5.9477896485724715
Node 9 12.048780813297025
Node 10 -9.909118916148998
Node 11 -11.979311179084942
Node 12 11.400376821620185
Node 13 11.079335342877991
Node 14 -10.721991890953783
Node 15 -4.962808889995599
Node 16 -6.547503102472978
Node 17 9.718553251965732
Node 18 -12.312469962295644

Sigmoid Node 1
Inputs Weights
Threshold -2.525866253650922
Node 2 8.844713888702545
Node 3 -10.692097335670164
Node 4 -11.107476189925103
Node 5 6.086318317864395
Node 6 14.095929036786316
Node 7 9.142100759986405
Node 8 -5.947933027144226
Node 9 -12.049135595614372
Node 10 9.909468228915939
Node 11 11.979645922026346
Node 12 -11.400696875430254
Node 13 -11.079632456565568
Node 14 10.722288065425365
Node 15 4.962945808332279
Node 16 6.547676504042953
Node 17 -9.718841719786557
Node 18 12.312794864709309

Sigmoid Node 2
Inputs Weights
Threshold 2.1734710551025094
Attrib AGE 3.011444954048342

Attrib MAX_BILI_1ST_DAY 2.431100946448702
Attrib MAX_BILI_2ND_DAY 1.5464510940634086
Attrib MAX_BILI_3RD_DAY 0.7355814781056973
Attrib MAX_HR_1ST_DAY -0.09986320578469327
Attrib MAX_HR_2ND_DAY 3.06998489870333
Attrib MAX_HR_3RD_DAY -4.68952739683681
Attrib MAX_SODIUM_1ST_DAY 0.653663840706018
Attrib MAX_SODIUM_2ND_DAY -2.0949964745471665
Attrib MAX_SODIUM_3RD_DAY -4.540032243689924
Attrib MIN_SYSBP_1ST_DAY -2.8552702920702964
Attrib MIN_SYSBP_2ND_DAY 0.7673156176541284
Attrib MIN_SYSBP_3RD_DAY 8.024315970632955
Attrib MIN_WBC_1ST_DAY 3.0714763524982938
Attrib MIN_WBC_2ND_DAY 6.009238798951789
Attrib MIN_WBC_3RD_DAY 8.102477511670925
Attrib MIN_POTASSIUM_1ST_DAY -3.99229519117954
Attrib MIN_POTASSIUM_2ND_DAY 1.678206335946652
Attrib MIN_POTASSIUM_3RD_DAY 9.305431518207378
Attrib MAX_BUN_1ST_DAY -5.749564980570519
Attrib MAX_BUN_2ND_DAY -0.9189037118801223
Attrib MAX_BUN_3RD_DAY 0.9967074071536108
Attrib MIN_GCS_1ST_DAY -5.576630579698842
Attrib MAX_TEMP_1ST_DAY 1.8628803800018758
Attrib MAX_TEMP_2ND_DAY -0.46950015911189685
Attrib MAX_TEMP_3RD_DAY -3.449938689673831
Attrib MIN_BICARBONATE_1ST_DAY 5.261278142374352
Attrib MIN_BICARBONATE_2ND_DAY -3.3240066307473173
Attrib MIN_BICARBONATE_3RD_DAY 8.246792526456543
Attrib OUT_1ST_DAY -0.9375725192816013
Attrib OUT_2ND_DAY -8.681170140382594
Attrib OUT_3RD_DAY -2.2276958809920324

Sigmoid Node 3
Inputs Weights
Threshold -2.2143117232441596
Attrib AGE 1.9866129125685366
Attrib MAX_BILI_1ST_DAY 3.616139552696913
Attrib MAX_BILI_2ND_DAY 0.9327706577032954
Attrib MAX_BILI_3RD_DAY 0.7646661188849441
Attrib MAX_HR_1ST_DAY 4.2499753537208935
Attrib MAX_HR_2ND_DAY -1.6041387445809268
Attrib MAX_HR_3RD_DAY 0.6140432779823328
Attrib MAX_SODIUM_1ST_DAY 4.088078991052013
Attrib MAX_SODIUM_2ND_DAY -2.7599939229144512
Attrib MAX_SODIUM_3RD_DAY 2.150889216771916
Attrib MIN_SYSBP_1ST_DAY 3.383475876050692
Attrib MIN_SYSBP_2ND_DAY 0.3423138749105682
Attrib MIN_SYSBP_3RD_DAY 5.762552107593115

Attrib MIN_WBC_1ST_DAY -0.632396786728002
Attrib MIN_WBC_2ND_DAY 1.8280508394191268
Attrib MIN_WBC_3RD_DAY 6.1067253819110485
Attrib MIN_POTASSIUM_1ST_DAY -1.1185198974014028
Attrib MIN_POTASSIUM_2ND_DAY 7.495319196334249
Attrib MIN_POTASSIUM_3RD_DAY 3.142674381602027
Attrib MAX_BUN_1ST_DAY 9.427900752670913
Attrib MAX_BUN_2ND_DAY -3.45153976000363
Attrib MAX_BUN_3RD_DAY -5.061584024587938
Attrib MIN_GCS_1ST_DAY 3.51427609768878
Attrib MAX_TEMP_1ST_DAY 2.0825449346890257
Attrib MAX_TEMP_2ND_DAY 0.34389071970147367
Attrib MAX_TEMP_3RD_DAY -0.8945489565172902
Attrib MIN_BICARBONATE_1ST_DAY -10.765061824947725
Attrib MIN_BICARBONATE_2ND_DAY -1.447055541198846
Attrib MIN_BICARBONATE_3RD_DAY -4.260108530984933
Attrib OUT_1ST_DAY -4.5617965180534235
Attrib OUT_2ND_DAY -2.153336331777369
Attrib OUT_3RD_DAY 0.6206738020451831

Sigmoid Node 4
Inputs Weights
Threshold -2.5906187117300887
Attrib AGE -0.04703878516703357
Attrib MAX_BILI_1ST_DAY 4.611084283371843
Attrib MAX_BILI_2ND_DAY 4.956216709597813
Attrib MAX_BILI_3RD_DAY 4.506473870821703
Attrib MAX_HR_1ST_DAY -12.113275422055144
Attrib MAX_HR_2ND_DAY -2.627960221939177
Attrib MAX_HR_3RD_DAY 4.184439595327411
Attrib MAX_SODIUM_1ST_DAY 7.171182198992128
Attrib MAX_SODIUM_2ND_DAY -3.195131190963797
Attrib MAX_SODIUM_3RD_DAY 3.8196211021922766
Attrib MIN_SYSBP_1ST_DAY -4.1852935935053095
Attrib MIN_SYSBP_2ND_DAY -0.5233076626299813
Attrib MIN_SYSBP_3RD_DAY 0.41347674704275633
Attrib MIN_WBC_1ST_DAY 0.6813313767223755
Attrib MIN_WBC_2ND_DAY 5.672860203551183
Attrib MIN_WBC_3RD_DAY 0.5838022284997787
Attrib MIN_POTASSIUM_1ST_DAY 9.909637704339321
Attrib MIN_POTASSIUM_2ND_DAY 5.008715506332269
Attrib MIN_POTASSIUM_3RD_DAY 8.996721707659502
Attrib MAX_BUN_1ST_DAY -8.578231796624793
Attrib MAX_BUN_2ND_DAY -4.024182612187897
Attrib MAX_BUN_3RD_DAY -4.135292334879276
Attrib MIN_GCS_1ST_DAY 8.395044647996542
Attrib MAX_TEMP_1ST_DAY 4.601658667494381
Attrib MAX_TEMP_2ND_DAY 1.2148144831470682

Attrib MAX_TEMP_3RD_DAY   3.9383279099175734
Attrib MIN_BICARBONATE_1ST_DAY   6.830821347600192
Attrib MIN_BICARBONATE_2ND_DAY   4.581285172105362
Attrib MIN_BICARBONATE_3RD_DAY   -3.3472048149285705
Attrib OUT_1ST_DAY   2.4667226422818684
Attrib OUT_2ND_DAY   -0.7525820319583082
Attrib OUT_3RD_DAY   -12.138358704368901

Sigmoid Node 5
Inputs   Weights
Threshold   -0.40298581543672196
Attrib AGE   3.2996392521205355
Attrib MAX_BILI_1ST_DAY   0.8142140064404398
Attrib MAX_BILI_2ND_DAY   1.1597527204356795
Attrib MAX_BILI_3RD_DAY   1.099024011158244
Attrib MAX_HR_1ST_DAY   3.101912481231386
Attrib MAX_HR_2ND_DAY   3.350620573675467
Attrib MAX_HR_3RD_DAY   2.4397068396490416
Attrib MAX_SODIUM_1ST_DAY   -0.533586304831406
Attrib MAX_SODIUM_2ND_DAY   -4.048852653084905
Attrib MAX_SODIUM_3RD_DAY   -1.8469950970695093
Attrib MIN_SYSBP_1ST_DAY   1.222382984227454
Attrib MIN_SYSBP_2ND_DAY   1.341972258606604
Attrib MIN_SYSBP_3RD_DAY   -0.09504894113598965
Attrib MIN_WBC_1ST_DAY   -2.363000339474406
Attrib MIN_WBC_2ND_DAY   -3.1912083111404606
Attrib MIN_WBC_3RD_DAY   -3.8067635672064486
Attrib MIN_POTASSIUM_1ST_DAY   0.5531162140951609
Attrib MIN_POTASSIUM_2ND_DAY   2.7308815818312118
Attrib MIN_POTASSIUM_3RD_DAY   -0.9979778327183316
Attrib MAX_BUN_1ST_DAY   0.5963324483762282
Attrib MAX_BUN_2ND_DAY   -0.37385377552140237
Attrib MAX_BUN_3RD_DAY   0.491343471885564
Attrib MIN_GCS_1ST_DAY   -4.574150353346714
Attrib MAX_TEMP_1ST_DAY   5.502345914927062
Attrib MAX_TEMP_2ND_DAY   -0.6101232112300015
Attrib MAX_TEMP_3RD_DAY   -3.4176015133968556
Attrib MIN_BICARBONATE_1ST_DAY   2.739559308831453
Attrib MIN_BICARBONATE_2ND_DAY   -1.234377189112169
Attrib MIN_BICARBONATE_3RD_DAY   -2.019857456648258
Attrib OUT_1ST_DAY   0.32612887126923973
Attrib OUT_2ND_DAY   -0.08258903653350672
Attrib OUT_3RD_DAY   -3.357979227039707

Sigmoid Node 6
Inputs   Weights
Threshold   0.9270946216020787
Attrib AGE   7.694336768599923

Attrib MAX_BILI_1ST_DAY 3.9030387496881174
Attrib MAX_BILI_2ND_DAY 5.3605626334838625
Attrib MAX_BILI_3RD_DAY 4.952341400544528
Attrib MAX_HR_1ST_DAY 2.8390815615822493
Attrib MAX_HR_2ND_DAY -5.492102120405208
Attrib MAX_HR_3RD_DAY -10.091754565227909
Attrib MAX_SODIUM_1ST_DAY 10.203092557942913
Attrib MAX_SODIUM_2ND_DAY 3.2615245070455394
Attrib MAX_SODIUM_3RD_DAY -1.3899037669197918
Attrib MIN_SYSBP_1ST_DAY 3.000729334515342
Attrib MIN_SYSBP_2ND_DAY -0.22196330861603672
Attrib MIN_SYSBP_3RD_DAY -3.879450888768234
Attrib MIN_WBC_1ST_DAY -3.630232256737846
Attrib MIN_WBC_2ND_DAY 2.952554388848692
Attrib MIN_WBC_3RD_DAY 1.513570165994642
Attrib MIN_POTASSIUM_1ST_DAY 6.265658855621252
Attrib MIN_POTASSIUM_2ND_DAY 0.7858940281615161
Attrib MIN_POTASSIUM_3RD_DAY -5.654165773398329
Attrib MAX_BUN_1ST_DAY 2.6939711161288624
Attrib MAX_BUN_2ND_DAY 3.431430617968402
Attrib MAX_BUN_3RD_DAY 6.7986272259779055
Attrib MIN_GCS_1ST_DAY -2.237686206645055
Attrib MAX_TEMP_1ST_DAY -1.2907084903479398
Attrib MAX_TEMP_2ND_DAY 6.84187118521024
Attrib MAX_TEMP_3RD_DAY 1.2962834922177762
Attrib MIN_BICARBONATE_1ST_DAY -0.21332518392336686
Attrib MIN_BICARBONATE_2ND_DAY 3.7687615434594646
Attrib MIN_BICARBONATE_3RD_DAY -1.7829312400710553
Attrib OUT_1ST_DAY -2.5417540683681312
Attrib OUT_2ND_DAY -3.1215149520067746
Attrib OUT_3RD_DAY 2.7043090226389

Sigmoid Node 7
Inputs Weights
Threshold -4.083345952431022
Attrib AGE 1.6339415000582338
Attrib MAX_BILI_1ST_DAY 3.0440086379285916
Attrib MAX_BILI_2ND_DAY 2.974452641838227
Attrib MAX_BILI_3RD_DAY 3.4101594176484147
Attrib MAX_HR_1ST_DAY -1.022573336366865
Attrib MAX_HR_2ND_DAY -1.1589208711339076
Attrib MAX_HR_3RD_DAY -1.1143934637739505
Attrib MAX_SODIUM_1ST_DAY 0.07578518674896377
Attrib MAX_SODIUM_2ND_DAY 7.773432732989296
Attrib MAX_SODIUM_3RD_DAY 3.074482511406869
Attrib MIN_SYSBP_1ST_DAY 0.7478725430049922

Attrib MIN_SYSBP_2ND_DAY   1.7779413957649253
Attrib MIN_SYSBP_3RD_DAY   4.678430215300333
Attrib MIN_WBC_1ST_DAY   -7.621477630039714
Attrib MIN_WBC_2ND_DAY   -3.934158037002979
Attrib MIN_WBC_3RD_DAY   0.5714704568371883
Attrib MIN_POTASSIUM_1ST_DAY   1.2199476384515149
Attrib MIN_POTASSIUM_2ND_DAY   0.5693650523672749
Attrib MIN_POTASSIUM_3RD_DAY   9.81007615812578
Attrib MAX_BUN_1ST_DAY   -3.845675922578348
Attrib MAX_BUN_2ND_DAY   1.1722350147592946
Attrib MAX_BUN_3RD_DAY   0.5599135331799119
Attrib MIN_GCS_1ST_DAY   -0.10780167446461286
Attrib MAX_TEMP_1ST_DAY   3.3757036912884
Attrib MAX_TEMP_2ND_DAY   0.864475006886609
Attrib MAX_TEMP_3RD_DAY   3.3604497740116477
Attrib MIN_BICARBONATE_1ST_DAY   0.5963080719541503
Attrib MIN_BICARBONATE_2ND_DAY   4.9109312396237215
Attrib MIN_BICARBONATE_3RD_DAY   4.562077402442496
Attrib OUT_1ST_DAY   1.5734368877202973
Attrib OUT_2ND_DAY   -6.0353860436988
Attrib OUT_3RD_DAY   4.31642643290264

Sigmoid Node 8
Inputs   Weights
Threshold   0.07344978570336876
Attrib AGE   -3.2193275191023067
Attrib MAX_BILI_1ST_DAY   0.51448437257876
Attrib MAX_BILI_2ND_DAY   0.2408378306113538
Attrib MAX_BILI_3RD_DAY   0.3910976806134419
Attrib MAX_HR_1ST_DAY   -4.070255271782322
Attrib MAX_HR_2ND_DAY   -0.13493073520115578
Attrib MAX_HR_3RD_DAY   -2.9906623226435953
Attrib MAX_SODIUM_1ST_DAY   -2.5166691238936223
Attrib MAX_SODIUM_2ND_DAY   0.6079527186359707
Attrib MAX_SODIUM_3RD_DAY   2.528394372281778
Attrib MIN_SYSBP_1ST_DAY   -1.6305753389250208
Attrib MIN_SYSBP_2ND_DAY   2.2671578484791888
Attrib MIN_SYSBP_3RD_DAY   -0.09172502826504167
Attrib MIN_WBC_1ST_DAY   0.7059195699613782
Attrib MIN_WBC_2ND_DAY   1.1198096308432177
Attrib MIN_WBC_3RD_DAY   1.2611281914504486
Attrib MIN_POTASSIUM_1ST_DAY   -0.1270371680241
Attrib MIN_POTASSIUM_2ND_DAY   -2.641303052250215
Attrib MIN_POTASSIUM_3RD_DAY   2.2662826875327777
Attrib MAX_BUN_1ST_DAY   -0.8138254902631887
Attrib MAX_BUN_2ND_DAY   -1.4391947810389722
Attrib MAX_BUN_3RD_DAY   -2.1598255592487896
Attrib MIN_GCS_1ST_DAY   -4.044437946556673

Attrib MAX_TEMP_1ST_DAY -3.1734206153303464
Attrib MAX_TEMP_2ND_DAY 1.7829711360178002
Attrib MAX_TEMP_3RD_DAY -2.315955131363183
Attrib MIN_BICARBONATE_1ST_DAY -3.34319668811683
Attrib MIN_BICARBONATE_2ND_DAY 0.16424382205270546
Attrib MIN_BICARBONATE_3RD_DAY 0.2742030719436974
Attrib OUT_1ST_DAY 3.306429346217753
Attrib OUT_2ND_DAY 1.6527017190468745
Attrib OUT_3RD_DAY 0.4746185738230785

Sigmoid Node 9
Inputs Weights
Threshold 1.7379450279066948
Attrib AGE -2.83652063929421
Attrib MAX_BILI_1ST_DAY -1.4976177306325955
Attrib MAX_BILI_2ND_DAY -1.2746363459940178
Attrib MAX_BILI_3RD_DAY -0.44805699839942487
Attrib MAX_HR_1ST_DAY 1.4728890001764685
Attrib MAX_HR_2ND_DAY 2.6542620202032947
Attrib MAX_HR_3RD_DAY 2.0388900410712605
Attrib MAX_SODIUM_1ST_DAY -2.9006545031548034
Attrib MAX_SODIUM_2ND_DAY 1.2671634871956865
Attrib MAX_SODIUM_3RD_DAY -1.3461230864348783
Attrib MIN_SYSBP_1ST_DAY -1.3514242312760636
Attrib MIN_SYSBP_2ND_DAY 6.2187341027332455
Attrib MIN_SYSBP_3RD_DAY 13.118465679433758
Attrib MIN_WBC_1ST_DAY 0.5962355390297256
Attrib MIN_WBC_2ND_DAY 2.488086576000562
Attrib MIN_WBC_3RD_DAY -1.4316853735283128
Attrib MIN_POTASSIUM_1ST_DAY -2.1150041449608494
Attrib MIN_POTASSIUM_2ND_DAY 8.828417825153107
Attrib MIN_POTASSIUM_3RD_DAY 9.163967536838785
Attrib MAX_BUN_1ST_DAY 5.454957675611211
Attrib MAX_BUN_2ND_DAY -3.4944719082554165
Attrib MAX_BUN_3RD_DAY -11.573071465224345
Attrib MIN_GCS_1ST_DAY -6.142448582712379
Attrib MAX_TEMP_1ST_DAY -4.6108336767697
Attrib MAX_TEMP_2ND_DAY 12.793628897124266
Attrib MAX_TEMP_3RD_DAY 3.9406371076337856
Attrib MIN_BICARBONATE_1ST_DAY 6.600626641346628
Attrib MIN_BICARBONATE_2ND_DAY 6.630326818947827
Attrib MIN_BICARBONATE_3RD_DAY 5.155536160158344
Attrib OUT_1ST_DAY 6.36646629364538
Attrib OUT_2ND_DAY 8.709680653292809
Attrib OUT_3RD_DAY 0.07750051817479571

Sigmoid Node 10
Inputs Weights

Threshold 0.7945904748545054
Attrib AGE 0.11545903418767693
Attrib MAX_BILI_1ST_DAY 0.5864583167646756
Attrib MAX_BILI_2ND_DAY 0.8191002502855731
Attrib MAX_BILI_3RD_DAY 0.2076085940843961
Attrib MAX_HR_1ST_DAY 0.8658243361678672
Attrib MAX_HR_2ND_DAY 3.2274435000605943
Attrib MAX_HR_3RD_DAY 0.02281349994215846
Attrib MAX_SODIUM_1ST_DAY -4.277219463253005
Attrib MAX_SODIUM_2ND_DAY -4.563991936268182
Attrib MAX_SODIUM_3RD_DAY 0.1901545875997017
Attrib MIN_SYSBP_1ST_DAY 1.4030600181759911
Attrib MIN_SYSBP_2ND_DAY -3.7594430865439428
Attrib MIN_SYSBP_3RD_DAY -7.726775639080874
Attrib MIN_WBC_1ST_DAY 0.6134911401404434
Attrib MIN_WBC_2ND_DAY 5.252962615989575
Attrib MIN_WBC_3RD_DAY 5.092193921052401
Attrib MIN_POTASSIUM_1ST_DAY 9.378839004710532
Attrib MIN_POTASSIUM_2ND_DAY 3.3423100959377923
Attrib MIN_POTASSIUM_3RD_DAY 7.787107758249437
Attrib MAX_BUN_1ST_DAY -1.0214374020697248
Attrib MAX_BUN_2ND_DAY -3.0516786921594874
Attrib MAX_BUN_3RD_DAY 0.1469994965201023
Attrib MIN_GCS_1ST_DAY 0.3159780657660722
Attrib MAX_TEMP_1ST_DAY 5.476850943531028
Attrib MAX_TEMP_2ND_DAY -3.54444452574292
Attrib MAX_TEMP_3RD_DAY 10.94920704749574
Attrib MIN_BICARBONATE_1ST_DAY 2.1130633454136887
Attrib MIN_BICARBONATE_2ND_DAY -1.2406834965190052
Attrib MIN_BICARBONATE_3RD_DAY -2.4782460785805625
Attrib OUT_1ST_DAY -1.7294143077158242
Attrib OUT_2ND_DAY 12.06261231608906
Attrib OUT_3RD_DAY -7.669216226491172

Sigmoid Node 11
Inputs Weights
Threshold 0.743893383005845
Attrib AGE -0.5262259110284028
Attrib MAX_BILI_1ST_DAY -0.7327996131734283
Attrib MAX_BILI_2ND_DAY 1.398291423462784
Attrib MAX_BILI_3RD_DAY 0.9514728969829901
Attrib MAX_HR_1ST_DAY 5.394712448126475
Attrib MAX_HR_2ND_DAY 3.5797148028187578
Attrib MAX_HR_3RD_DAY -4.456276942406125
Attrib MAX_SODIUM_1ST_DAY -2.306114647440039
Attrib MAX_SODIUM_2ND_DAY 1.9903237106933103
Attrib MAX_SODIUM_3RD_DAY -1.9206184126364196
Attrib MIN_SYSBP_1ST_DAY 8.364225951302236

Attrib MIN_SYSBP_2ND_DAY   -0.3811687242311922
Attrib MIN_SYSBP_3RD_DAY   -5.95390210429209
Attrib MIN_WBC_1ST_DAY   5.775557778473224
Attrib MIN_WBC_2ND_DAY   6.15318527214792
Attrib MIN_WBC_3RD_DAY   5.505511083399141
Attrib MIN_POTASSIUM_1ST_DAY   0.21701817094948953
Attrib MIN_POTASSIUM_2ND_DAY   3.4784934510139367
Attrib MIN_POTASSIUM_3RD_DAY   -0.005678970048578292
Attrib MAX_BUN_1ST_DAY   -0.9497196059690504
Attrib MAX_BUN_2ND_DAY   0.38149296677141376
Attrib MAX_BUN_3RD_DAY   4.020534700153218
Attrib MIN_GCS_1ST_DAY   1.5265754588619864
Attrib MAX_TEMP_1ST_DAY   -2.895966936958027
Attrib MAX_TEMP_2ND_DAY   1.7638780747861522
Attrib MAX_TEMP_3RD_DAY   1.93657502353917
Attrib MIN_BICARBONATE_1ST_DAY   4.36802997516722
Attrib MIN_BICARBONATE_2ND_DAY   -3.153177567119412
Attrib MIN_BICARBONATE_3RD_DAY   3.1395706624082886
Attrib OUT_1ST_DAY   -5.650495929796473
Attrib OUT_2ND_DAY   -1.7318588022055903
Attrib OUT_3RD_DAY   -2.943428330999003

Sigmoid Node 12
Inputs   Weights
Threshold   -3.864475987346374
Attrib AGE   -3.3167321174802518
Attrib MAX_BILI_1ST_DAY   -1.4120112116471468
Attrib MAX_BILI_2ND_DAY   -2.931264231327299
Attrib MAX_BILI_3RD_DAY   -3.142319355973683
Attrib MAX_HR_1ST_DAY   -3.234432668526858
Attrib MAX_HR_2ND_DAY   8.819732190636646
Attrib MAX_HR_3RD_DAY   -10.975003680790545
Attrib MAX_SODIUM_1ST_DAY   2.339351390638358
Attrib MAX_SODIUM_2ND_DAY   -2.0482642587859607
Attrib MAX_SODIUM_3RD_DAY   -1.5184676168182303
Attrib MIN_SYSBP_1ST_DAY   4.039609150661839
Attrib MIN_SYSBP_2ND_DAY   -12.105678718242041
Attrib MIN_SYSBP_3RD_DAY   0.6577868907534105
Attrib MIN_WBC_1ST_DAY   -0.8023773348620934
Attrib MIN_WBC_2ND_DAY   1.0892532559399186
Attrib MIN_WBC_3RD_DAY   -0.022180578723652655
Attrib MIN_POTASSIUM_1ST_DAY   -1.8989571908630005
Attrib MIN_POTASSIUM_2ND_DAY   2.3404589408878875
Attrib MIN_POTASSIUM_3RD_DAY   -3.1058315620038544
Attrib MAX_BUN_1ST_DAY   0.630423863547478
Attrib MAX_BUN_2ND_DAY   -0.4739959438174737
Attrib MAX_BUN_3RD_DAY   -12.288632493227517
Attrib MIN_GCS_1ST_DAY   2.8968357551267245

Attrib MAX_TEMP_1ST_DAY 2.1329660637644765
Attrib MAX_TEMP_2ND_DAY 0.19301486166451767
Attrib MAX_TEMP_3RD_DAY -0.17153228814720395
Attrib MIN_BICARBONATE_1ST_DAY -5.363172378738092
Attrib MIN_BICARBONATE_2ND_DAY 9.330451775713254
Attrib MIN_BICARBONATE_3RD_DAY 7.208765143519193
Attrib OUT_1ST_DAY 8.610069234515294
Attrib OUT_2ND_DAY 8.529217211244376
Attrib OUT_3RD_DAY 0.9050119278433751

Sigmoid Node 13
Inputs Weights
Threshold 0.19505279279833537
Attrib AGE -22.48629565756519
Attrib MAX_BILI_1ST_DAY -0.966512150657686
Attrib MAX_BILI_2ND_DAY -2.9322888311365767
Attrib MAX_BILI_3RD_DAY -4.893318745433939
Attrib MAX_HR_1ST_DAY 6.26506542570398
Attrib MAX_HR_2ND_DAY -9.835919315842576
Attrib MAX_HR_3RD_DAY -4.056109224324377
Attrib MAX_SODIUM_1ST_DAY 3.456992630178913
Attrib MAX_SODIUM_2ND_DAY 1.4042844692657592
Attrib MAX_SODIUM_3RD_DAY 0.9446321176128758
Attrib MIN_SYSBP_1ST_DAY 12.490392345935858
Attrib MIN_SYSBP_2ND_DAY -0.4507477648991308
Attrib MIN_SYSBP_3RD_DAY 3.881218465589853
Attrib MIN_WBC_1ST_DAY -4.989501432205054
Attrib MIN_WBC_2ND_DAY 4.1428171353291425
Attrib MIN_WBC_3RD_DAY 2.4750580392710266
Attrib MIN_POTASSIUM_1ST_DAY -0.10044637957351747
Attrib MIN_POTASSIUM_2ND_DAY -7.450014539279326
Attrib MIN_POTASSIUM_3RD_DAY -1.9687599847412203
Attrib MAX_BUN_1ST_DAY 2.2971589236690506
Attrib MAX_BUN_2ND_DAY 1.627899992328025
Attrib MAX_BUN_3RD_DAY 5.644658792378511
Attrib MIN_GCS_1ST_DAY -1.9847599065646713
Attrib MAX_TEMP_1ST_DAY 1.100278060550258
Attrib MAX_TEMP_2ND_DAY 2.9099008091130383
Attrib MAX_TEMP_3RD_DAY -6.50497152434343
Attrib MIN_BICARBONATE_1ST_DAY 7.662516388872762
Attrib MIN_BICARBONATE_2ND_DAY -0.46691381656004116
Attrib MIN_BICARBONATE_3RD_DAY 7.431684873941655
Attrib OUT_1ST_DAY 3.6450233034387485
Attrib OUT_2ND_DAY 4.417427344018491
Attrib OUT_3RD_DAY 12.67156453520751

Sigmoid Node 14
Inputs Weights

Threshold   -4.397730436121573
Attrib AGE   -2.0053112496464114
Attrib MAX_BILI_1ST_DAY   0.15229933832875675
Attrib MAX_BILI_2ND_DAY   1.2916530189132545
Attrib MAX_BILI_3RD_DAY   2.3432845044040973
Attrib MAX_HR_1ST_DAY   -1.5202021521992386
Attrib MAX_HR_2ND_DAY   -6.191839123660413
Attrib MAX_HR_3RD_DAY   -0.6924475668242038
Attrib MAX_SODIUM_1ST_DAY   -0.9673099303948176
Attrib MAX_SODIUM_2ND_DAY   -5.479475287797898
Attrib MAX_SODIUM_3RD_DAY   2.474006869330379
Attrib MIN_SYSBP_1ST_DAY   2.0489935966843182
Attrib MIN_SYSBP_2ND_DAY   2.1421613633836314
Attrib MIN_SYSBP_3RD_DAY   4.192687295775002
Attrib MIN_WBC_1ST_DAY   3.492295288611128
Attrib MIN_WBC_2ND_DAY   3.834291927565411
Attrib MIN_WBC_3RD_DAY   5.993387185548316
Attrib MIN_POTASSIUM_1ST_DAY   1.904682815254465
Attrib MIN_POTASSIUM_2ND_DAY   -5.356804182202292
Attrib MIN_POTASSIUM_3RD_DAY   -11.829233747114696
Attrib MAX_BUN_1ST_DAY   -0.20682046371672427
Attrib MAX_BUN_2ND_DAY   0.47865083308989803
Attrib MAX_BUN_3RD_DAY   -0.8039526803941232
Attrib MIN_GCS_1ST_DAY   -1.3181341428457687
Attrib MAX_TEMP_1ST_DAY   3.552960740722941
Attrib MAX_TEMP_2ND_DAY   -3.4175607967903674
Attrib MAX_TEMP_3RD_DAY   2.3422777558190906
Attrib MIN_BICARBONATE_1ST_DAY   -0.8659006608198601
Attrib MIN_BICARBONATE_2ND_DAY   1.13202271669212
Attrib MIN_BICARBONATE_3RD_DAY   4.242713043207359
Attrib OUT_1ST_DAY   -0.3445331879849189
Attrib OUT_2ND_DAY   -5.364410631834074
Attrib OUT_3RD_DAY   -1.596004824991153

Sigmoid Node 15
Inputs   Weights
Threshold   -3.0555500929920147
Attrib AGE   -1.0386670975916399
Attrib MAX_BILI_1ST_DAY   1.5687575364755895
Attrib MAX_BILI_2ND_DAY   1.9062166972003383
Attrib MAX_BILI_3RD_DAY   2.1891272579026664
Attrib MAX_HR_1ST_DAY   -2.5723227443836616
Attrib MAX_HR_2ND_DAY   6.241223647979989
Attrib MAX_HR_3RD_DAY   3.3877629974084082
Attrib MAX_SODIUM_1ST_DAY   -0.8714467539478228
Attrib MAX_SODIUM_2ND_DAY   7.190837991161446
Attrib MAX_SODIUM_3RD_DAY   0.6775016724152704
Attrib MIN_SYSBP_1ST_DAY   2.525887937609905

Attrib MIN_SYSBP_2ND_DAY -5.004484183303543
Attrib MIN_SYSBP_3RD_DAY -2.5984928459263252
Attrib MIN_WBC_1ST_DAY -2.613170281818693
Attrib MIN_WBC_2ND_DAY -3.3615263493259806
Attrib MIN_WBC_3RD_DAY -1.4817461925590583
Attrib MIN_POTASSIUM_1ST_DAY -2.97436352754108
Attrib MIN_POTASSIUM_2ND_DAY -5.91884392565617
Attrib MIN_POTASSIUM_3RD_DAY 2.727732494194443
Attrib MAX_BUN_1ST_DAY 0.07559313016918975
Attrib MAX_BUN_2ND_DAY -0.19127000783522682
Attrib MAX_BUN_3RD_DAY -4.346969189433085
Attrib MIN_GCS_1ST_DAY 2.2741585325518274
Attrib MAX_TEMP_1ST_DAY -0.6443788266461213
Attrib MAX_TEMP_2ND_DAY -0.07090565893123027
Attrib MAX_TEMP_3RD_DAY 3.5640570484651466
Attrib MIN_BICARBONATE_1ST_DAY -0.38186769925103536
Attrib MIN_BICARBONATE_2ND_DAY -0.2095549324753679
Attrib MIN_BICARBONATE_3RD_DAY 2.244717405468632
Attrib OUT_1ST_DAY 0.8310666505545408
Attrib OUT_2ND_DAY 1.267946977098172
Attrib OUT_3RD_DAY 4.117365371105995

Sigmoid Node 16
Inputs Weights
Threshold -1.3310885510240347
Attrib AGE -3.545237774056753
Attrib MAX_BILI_1ST_DAY 0.46774257730800944
Attrib MAX_BILI_2ND_DAY 0.8267520187290993
Attrib MAX_BILI_3RD_DAY 1.1107852626520378
Attrib MAX_HR_1ST_DAY -7.989666265732035
Attrib MAX_HR_2ND_DAY -4.4693024764659155
Attrib MAX_HR_3RD_DAY 0.510078433926595
Attrib MAX_SODIUM_1ST_DAY -6.045996724962367
Attrib MAX_SODIUM_2ND_DAY 4.122228109561406
Attrib MAX_SODIUM_3RD_DAY 3.5658313781673385
Attrib MIN_SYSBP_1ST_DAY 2.1464190394222618
Attrib MIN_SYSBP_2ND_DAY -6.447697844924475
Attrib MIN_SYSBP_3RD_DAY 0.24853731716824204
Attrib MIN_WBC_1ST_DAY 2.376186277867898
Attrib MIN_WBC_2ND_DAY 1.1793273573528515
Attrib MIN_WBC_3RD_DAY 2.841077378793076
Attrib MIN_POTASSIUM_1ST_DAY 0.7806974504068874
Attrib MIN_POTASSIUM_2ND_DAY 1.3172919816851012
Attrib MIN_POTASSIUM_3RD_DAY 2.075438839306262
Attrib MAX_BUN_1ST_DAY -4.4675657498152965
Attrib MAX_BUN_2ND_DAY -2.6146650617365483
Attrib MAX_BUN_3RD_DAY 0.32372602256006267
Attrib MIN_GCS_1ST_DAY 7.666631763978966

Attrib MAX_TEMP_1ST_DAY   -1.2911558182498644
Attrib MAX_TEMP_2ND_DAY   -1.0320662673130303
Attrib MAX_TEMP_3RD_DAY   2.8779967488617095
Attrib MIN_BICARBONATE_1ST_DAY   0.010751741944686597
Attrib MIN_BICARBONATE_2ND_DAY   -0.5858268512748303
Attrib MIN_BICARBONATE_3RD_DAY   -0.15503040341701382
Attrib OUT_1ST_DAY   1.8844041917299545
Attrib OUT_2ND_DAY   2.5497868044249166
Attrib OUT_3RD_DAY   4.053708133912279

Sigmoid Node 17
Inputs   Weights
Threshold   -2.8890516108460487
Attrib AGE   11.679266832455
Attrib MAX_BILI_1ST_DAY   0.932404572010839
Attrib MAX_BILI_2ND_DAY   1.6870171241496021
Attrib MAX_BILI_3RD_DAY   1.681046157074337
Attrib MAX_HR_1ST_DAY   -1.3282347659159066
Attrib MAX_HR_2ND_DAY   0.05110325346289713
Attrib MAX_HR_3RD_DAY   3.462064219996326
Attrib MAX_SODIUM_1ST_DAY   -1.7541639263504574
Attrib MAX_SODIUM_2ND_DAY   -6.1519833734520795
Attrib MAX_SODIUM_3RD_DAY   -1.7684987264777094
Attrib MIN_SYSBP_1ST_DAY   3.6296304405212036
Attrib MIN_SYSBP_2ND_DAY   -3.248094742092814
Attrib MIN_SYSBP_3RD_DAY   -0.8618991659361398
Attrib MIN_WBC_1ST_DAY   3.4510375642670676
Attrib MIN_WBC_2ND_DAY   1.1353986568049905
Attrib MIN_WBC_3RD_DAY   -1.0523364977938654
Attrib MIN_POTASSIUM_1ST_DAY   8.69634932663346
Attrib MIN_POTASSIUM_2ND_DAY   -3.1404994147837493
Attrib MIN_POTASSIUM_3RD_DAY   2.855895572590571
Attrib MAX_BUN_1ST_DAY   3.476435738445983
Attrib MAX_BUN_2ND_DAY   1.7755585664158617
Attrib MAX_BUN_3RD_DAY   0.6552837199372622
Attrib MIN_GCS_1ST_DAY   -3.581793840333377
Attrib MAX_TEMP_1ST_DAY   11.832376299820314
Attrib MAX_TEMP_2ND_DAY   -1.6309736635270746
Attrib MAX_TEMP_3RD_DAY   -6.08397904476542
Attrib MIN_BICARBONATE_1ST_DAY   3.753822450791
Attrib MIN_BICARBONATE_2ND_DAY   -3.889394358252617
Attrib MIN_BICARBONATE_3RD_DAY   -0.6362321162004614
Attrib OUT_1ST_DAY   -3.145005050467075
Attrib OUT_2ND_DAY   -9.921516280694009
Attrib OUT_3RD_DAY   -2.141049272381556

Sigmoid Node 18
Inputs   Weights

Threshold -0.5966223091749483
Attrib AGE -0.5442012913448329
Attrib MAX_BILI_1ST_DAY 2.2770184474866872
Attrib MAX_BILI_2ND_DAY 3.889761653504749
Attrib MAX_BILI_3RD_DAY 4.416917862495088
Attrib MAX_HR_1ST_DAY -4.475428636538316
Attrib MAX_HR_2ND_DAY -0.08256696685541495
Attrib MAX_HR_3RD_DAY 1.9000312506974542
Attrib MAX_SODIUM_1ST_DAY 2.7384410883572103
Attrib MAX_SODIUM_2ND_DAY -0.05123240525546232
Attrib MAX_SODIUM_3RD_DAY -4.362777269503207
Attrib MIN_SYSBP_1ST_DAY 5.052617399492712
Attrib MIN_SYSBP_2ND_DAY -4.4592312043114
Attrib MIN_SYSBP_3RD_DAY -3.979051378070056
Attrib MIN_WBC_1ST_DAY -7.202265996404142
Attrib MIN_WBC_2ND_DAY -2.301635338638274
Attrib MIN_WBC_3RD_DAY 1.8968011515211245
Attrib MIN_POTASSIUM_1ST_DAY 4.028088797469194
Attrib MIN_POTASSIUM_2ND_DAY -1.2214108362376375
Attrib MIN_POTASSIUM_3RD_DAY -0.879960861489472
Attrib MAX_BUN_1ST_DAY 0.9051908777041918
Attrib MAX_BUN_2ND_DAY 2.5936226242648455
Attrib MAX_BUN_3RD_DAY 2.7133548315921012
Attrib MIN_GCS_1ST_DAY -1.8762488346013038
Attrib MAX_TEMP_1ST_DAY -5.150025483474964
Attrib MAX_TEMP_2ND_DAY -0.7459544380881137
Attrib MAX_TEMP_3RD_DAY 5.000186963681234
Attrib MIN_BICARBONATE_1ST_DAY 5.733220756790168
Attrib MIN_BICARBONATE_2ND_DAY 8.094207090611427
Attrib MIN_BICARBONATE_3RD_DAY 2.0618487572063047
Attrib OUT_1ST_DAY 1.4303469428246023
Attrib OUT_2ND_DAY 1.065783095140918
Attrib OUT_3RD_DAY -2.371829710708604

Correctly Classified Instances 642 65.5771 %
Incorrectly Classified Instances 337 34.4229 %
Kappa statistic 0.1928
Mean absolute error 0.3428
Root mean squared error 0.5512
Relative absolute error 79.0545 %
Root relative squared error 118.3819 %
Total Number of Instances 979

TP Rate	FP Rate	Precision	Recall	F-Measure	ROC Area	Class
0.762	0.572	0.741	0.762	0.751	0.633	N
0.428	0.238	0.455	0.428	0.441	0.633	Y

Appendix E. Simple CART to Predict Hospital Mortality using Tenfold Cross-Validation (Weka 3.5.7)

CART Decision Tree: N(668.0/311.0)
Number of Leaf Nodes: 1
Size of the Tree: 1

Correctly Classified Instances	661	67.5179 %
Incorrectly Classified Instances	318	32.4821 %
Kappa statistic	0.0915	
Mean absolute error	0.4154	
Root mean squared error	0.4753	
Relative absolute error	95.7864 %	
Root relative squared error	102.0876 %	
Total Number of Instances	979	

TP Rate	FP Rate	Precision	Recall	F-Measure	ROC Area	Class
0.915	0.839	0.701	0.915	0.794	0.562	N
0.161	0.085	0.467	0.161	0.239	0.562	Y