Predicting long-term mortality with first week post-operative data after Coronary Artery Bypass Grafting using Machine Learning models

José Castela Forte J.N.ALVES.CASTELA.CARDOSO.FORTE@STUDENT.RUG.NL
Faculty of Medical Sciences
University of Groningen
Groningen, The Netherlands

Marco A. Wiering M.A.WIERING@RUG.NL
Institute of Artificial Intelligence and Cognitive Engineering
Faculty of Science and Engineering, University of Groningen
Groningen, The Netherlands

Hjalmar R. Bouma H.R.BOUMA@UMCG.NL
Departments of Internal Medicine and Clinical Pharmacy and Pharmacology
University Medical Centre Groningen, University of Groningen
Groningen, The Netherlands

Fred de Geus A.F.DE.GEUS@UMCG.NL
Department of Anesthesiology
University Medical Centre Groningen, University of Groningen
Groningen, The Netherlands

Anne H. Epema A.H.EPEMA@UMCG.NL
Department of Anesthesiology
University Medical Centre Groningen, University of Groningen
Groningen, The Netherlands

Abstract

Coronary Artery Bypass Graft (CABG) surgery is the most common cardiac operation and its complications are associated with increased long-term mortality rates. Although many factors are known to be linked to this, much remains to be understood about their exact influence on outcome. In this study we used Machine Learning (ML) algorithms to predict long-term mortality in CABG patients using data from routinely measured clinical parameters from a large cohort of CABG patients (n=5868). We compared the accuracy of 5 different ML models with traditional Cox and Logistic Regression, and report the most important variables in the best performing models. In the validation dataset, the Gradient Boosted Machine (GBM) algorithm was the most accurate (AUROC curve [95%CI] of 0.767 [0.739-0.796]), proving to be superior to traditional Cox and logistic regression ($p < 0.01$) for long-term mortality prediction. Measures of variable importance for outcome prediction extracted from the GBM and Random Forest models partly reflected what is known in the literature, but interestingly also highlighted other unexpectedly relevant parameters. In conclusion, we found ML algorithm-based models to be more accurate than traditional Logistic Regression in predicting long-term mortality after CABG. Finally, these models may pro-

vide essential input to assist the development of intelligent decision support systems for clinical use.

Keywords: CABG, Mortality, Prediction, Machine Learning, Decision Support Systems

1. Introduction

Coronary artery bypass graft surgery (CABG) is the most frequently performed cardiac surgical operation in patients with coronary artery disease. Different factors are known to be associated with both in-hospital and long-term mortality (Aranki and Aroesty, 2017). Acute kidney injury (AKI) is one of the most common complications after CABG operations and present in 15-30% of the operated patients (Loef et al., 2009). Earlier studies have shown that even small changes in renal function during the first post-operative week can predict long-term mortality (Loef et al., 2009; Lassnigg et al., 2005; Loef et al., 2005). Similarly, age, hemoglobin concentration and comorbid conditions have been associated with post-operative kidney injury (Loef et al., 2005; Westenbrink et al., 2011; Shahian et al., 2012). The mechanisms behind this increased mortality rate after CABG operations are poorly understood, and it is not clear how different patient- and procedure-related factors contribute to long-term mortality in CABG patients (Lassnigg et al., 2005; Loef et al., 2005; Westenbrink et al., 2011; Shahian et al., 2012). Therefore, refining long-term mortality models to identify individual patients at increased risk is necessary to assist in the development of intelligent decision support systems.

As such, we propose a novel approach to construct predictive models using more sophisticated data analysis techniques of patient data routinely obtained before, during and after cardiac surgery. Starting as a well-known method in gene analysis, Machine Learning (ML) has spread to multiple other fields of Medicine for predictive model development, variable importance analysis, and patient risk-stratification (Churpek et al., 2016; Deo, 2015; Allyn et al., 2017; Diaz-Uriarte and Alvarez de Andrés, 2006; Eslam et al., 2016). A recent paper describes how, based on findings achieved with ML techniques, 5 large medical centres successfully implemented changes in a vital postoperative procedure (Wolf et al., 2016). This highlights the great potential of ML-based, clinically-oriented research to identify and prioritize specific areas in which to improve care (Wolf et al., 2016). Despite this, there is scarce literature on the use of ML in the field of cardiac surgery (Allyn et al., 2017; Eslam et al., 2016; Wolf et al., 2016; Legrand et al., 2013). We did not find any papers describing the prediction of long-term, individual patient outcomes after CABG operation, one of the most common major in-hospital interventional procedures, with around 250.000 CABG procedures yearly in the United States alone (Epstein et al., 2011; NHS, 2017).

In this study, different ML algorithms are used to develop multiple models applied to prospectively collected data from a large patient cohort study to predict long-term mortality in patients who underwent CABG. We compared the accuracy of these ML models on a separate validation data set by means of ROC (Receiver Operated Characteristic) curves against each other and to the traditional gold standard Cox regression model. Our analysis demonstrated that ML-based models proved to be superior to the

traditional statistical method. Furthermore, we determined which variables from our large set of routinely measured clinical parameters are important predictors of patient outcome, showing that many parameters to which little attention is paid in practice, such as urea at post-operative day 4, have an unexpectedly high predictive power.

2. Cohort

The Cardiothoracic Anesthesiology Registry (CAROLA) comprises extensive data of all adult patients who underwent first-time valve surgery, CPB-assisted CABG, or both between 1997 and 2015 in the University Medical Centre Groningen. The total number of patients is 11994. Their mortality data, as of August 5th 2015, were collected from the Dutch Municipal Personal Records Database, which contains actual, reliable data of all citizens within the Netherlands.

The original cohort dataset has a total of 239 variables, including perioperative parameters, hemodynamic function, respiratory function, renal function, liver function, and blood values from samples collected during the operation, and at least once daily during their stay in the Intensive Care Unit (ICU), and subsequently at the ward at days 1, 3, and 5 (at least), and at discharge.

2.1 Cohort selection

We excluded those who underwent valve surgery and combined valve and CABG, and focused only on the patients who underwent CABG (n=5868). Compared to the full cohort, the subset of patients who underwent CABG has a slightly lower 5-year mortality rate (14.2% vs. 16.6%).

2.2 Data extraction

To make our models as complete as possible, and assure that variables which seem unimportant or even irrelevant for clinicians were not excluded prematurely, our initial dataset included the greatest possible number of variables. Due to early discharge or transfer of some patients to other hospitals, there was a variable pattern of missing data, for which reason we excluded all variables of the original 239 with 50% or more missing data, the highest acceptable percentage found in similar medical literature (Churpek et al., 2016). The missing data in the remaining variables was imputed using the "cart" method of the R package "mice" (Buuren et al., 2006; Buuren and Groothuis-Oudshoorn, 2011; Shah et al., 2014; Ridgeway, 2017). This package uses a method, Multivariate Imputation by Chained Equations (MICE), which is more accurate than single imputation procedures because it imputes missing values based on both the observed values for an individual and the relations observed in the data for other patients (Azur et al., 2011). In addition, we only included the predictor variables with a minimum proportion of usable cases $\geq$0.25, in an attempt to optimise the imputation. We validated our imputation by proving the similarity between the distribution of the residuals of the regression of "Creatinine at 1 week after surgery" (which previously had 37.3% missing data) on its propensity score for both the original and imputed data (Buuren and Groothuis-Oudshoorn, 2011) (Fig. 3, Appendix 1).

The final dataset without missing data consisted of 72 predictor variables and 3 outcome variables, of which 42 were numeric, 30 integers, and 4 factors. Tables 1 and 2 in Appendix 2 show the variables grouped by system with the data descriptives, and the raw dataset from R. The outcome variable for all patients was 5-year mortality, collected from the Dutch Municipal Personal Records Database. Because different models require the outcome variable to be coded differently, we coded mortality as a factor ("alive" or "deceased"), as an integer ("0" representing alive, and "1" deceased), and as numeric (the duration of follow-up in days, for the Cox Regression).

2.3 Feature selection

For logistic regression, RandomForest, and Gradient Boosted Machine pre-processing is not required (Kuhn, 2016). For Support Vector Machines and k-Nearest Neighbours, however, data must be pre-processed, which we did by centring the input data (subtracting the mean) and scaling (dividing by the standard deviation), using preProcess in the "caret" package (Hechenbichler and Schliep, 2004; Hsu et al., 2016; Field et al., 2012; Kuhn, 2016; Schliep and Hechenbichler, 2016). In addition, dummy variables were created for all categorical variables in SVM and wkNN (Schliep and Hechenbichler, 2016; C. Brown, 2012).

3. Methods

All analyses were performed using R version 3.3.2 (The R Foundation for Statistical Computing; Vienna, Austria) (R Development Core Team - R Foundation for Statistical Computing, 2008). Data in tables and in the results are expressed as mean (95% confidence interval), and, if categorical, with the percentage in parentheses. The accuracy of the models is reported as area under the ROC curve (AUROC), with a 95% confidence interval. Due to the closeness of the spread of the estimated accuracy of multiple models, we did a nonparametric test for the difference in areas under the curve as proposed by DeLong et al. (1988) and adapted in the "pROC" package for paired and unpaired ROC curves (DeLong et al., 1988; Robin et al., 2011). All curves were tested for correlation, with trained Logistic Regression (LR), SVM, and wkNN being found to be correlated. A p-value$<$0.05 was accepted as a statistically significant difference, and the difference in AUROC between the three correlated curves used the DeLong method for paired curves in "pROC" (Robin et al., 2011).

3.1 Model descriptions

We split the original dataset into training and testing/validation sets with 60% and 40% of the data, respectively. Each Machine Learning model, as well as one of the logistic regression models, was trained using 10-fold cross-validation with 10 repeats, so as to determine the optimal values of all modifiable parameters to maximize the model's AUROC. As described below, post-hoc sensitivity analysis was done to refine the parameters beyond what the tuning functions in R allow (Thabane et al., 2013). The Cox model was not trained, but a test set was also generated to obtain the AUROC for comparison.

3.1.1 COX, UNTRAINED, AND TRAINED LOGISTIC REGRESSION

We used a Cox Regression based on previous works from our team, and a logistic regression model, as benchmark to compare our ML models to. In the Cox regression, AKI adjusted for age, sex, and duration of perfusion was used as predictor. Subsequently this model was fitted with a lasso penalty, calibrated with 10-fold cross-validation at time=1825 days (our 5-year mortality endpoint), and externally validated with a Time-Dependent ROC curve using the data in the validation dataset (Xiao et al., 2016).

As a bridge to our ML approaches, two binomial logistic regression models (LR) were built using all predictor variables as covariates and the cases in our training dataset. The first generalized linear model built and fitted to the training data was not further improved ("untrained LR"), and was directly tested against the validation dataset. The second ("trained LR"), in fact a form of ML, was trained using the train function of the "caret" package before being tested (I. Brown and Mues, 2012).

3.1.2 SUPPORT VECTOR MACHINE

A Support Vector Machine (SVM) is a class of supervised learning algorithms mostly used in classification. It classifies data points into two different classes by taking these points in a multidimensional space and constructing the hyperplane that best differentiates between the two (Hastie et al., 2009). Furthermore, the use of kernels allows the projection of input data to a higher-dimensional space if necessary, which can increase the efficacy of the trained models.

Due to the great variation in range of our numeric predictor variables, we scaled and centred the numeric variables to prevent attributes with greater numeric ranges dominating those with smaller ranges (Field et al., 2012). The need for coding the categorical variables into dummy variables further changed the dataset, which, for SVM, had 80 predictor variables with 4 dummy variables for "AKI", 2 for "Sex", and 4 for "ICU categories".

The SVM model was tuned during training for three hyper-parameters: sigma (σ), cost (C), and weight (Kuhn, 2016). Sigma, like gamma (γ), defines "smoothing" when the Radial Basis Function (RBF) kernel is used; that is, how far the influence of a single training example reaches. An excessively big value would constrain the model by making the model linear and thus preventing it from capturing the complexity of the data (Hastie et al., 2009). The cost term (C) controls the misclassification tolerance by forcing the SVM towards a harder margin with a larger value, or allowing a smoother decision boundary and an increased probability of misclassification. The weight parameter is adjusted to compensate for the possible effect of imbalance between the majority class for the outcome "Mortality" in the training set ("alive" = 3021), and the minority class ("deceased" = 500). For our model, a weight of 0.166 was given to "alive". In this way, the algorithm is penalised more for misclassifying cases of the minority class than cases of the majority class during training (Batuwita and Palade, 2012; Akbani et al., 2004).

To allow the hyperplane boundary between classes to be non-linear, a Radial Basis Function (RBF) kernel was chosen, and the best values for various pairs of exponen-

tially growing C and sigma values were tried during training, with the best being determined by means of a "grid-search" using cross-validation (Hsu et al., 2016; Hastie et al., 2009). The initial parameter-value grid ranged from 2^{-7} to 2^{7} for both C and σ. The best value for the cost parameter was C= 20 for all values of σ. As such, with C= 20, another grid with σ ranging from 2^{-15} to 2^{-7} was tried, which yielded the best value for sigma at σ=2^{-9}. The smaller the sigma, the lower the specificity: a sigma of 2^{-20}, tried after the grid run to control for the decreasing trend found in specificity, yielded an almost null specificity. For higher values of sigma, the differences in AUROC were not as considerable and depended more on the value of C. Another adjustment of the grid showed that both C=20 and C=21 were similar for σ=2^{-9}, with C=20 being marginally better. When σ <2^{-9}, for any value of C, the AUROC consistently decreased and the specificity approached 0.

In the end, we tested the best fitting SVM model against the test set, with parameters σ=2^{-9}, C=20, weighted classes with "alive"=0.166, and an RBF kernel.

3.1.3 WEIGHTED K-NEAREST-NEIGHBOURS

Weighted k-nearest neighbours (wkNN) is an application of the common nearest neighbour technique, a non-parametric method for classification where a new observation is classified into a certain class based on proximity in the training set (Hechenbichler and Schliep, 2004). In the case of k-nearest neighbours, the k most similar cases are selected to predict the outcome of a new observation, which is classified into the class of the majority of the nearest "neighbouring" observations. Weighted kNN differs from, and improves, the classical method in that it weighs the nearest neighbours according to their similarity (i.e. distance) to the new observation, which is expected to reduce misclassification errors (Hechenbichler and Schliep, 2004; Schliep and Hechenbichler, 2016). On these grounds, the distances on which the search for the nearest neighbours is based have to be transformed into similarity measures (Schliep and Hechenbichler, 2016). Therefore, as with SVM, the data were centred and scaled.

The wkNN model was tuned during training for two hyper-parameters, distance (d) and k, with multiple kernels. Distance in this algorithm corresponds to the p in the Minkowski distance, given by

$$d(x, y) = (\sum_{(i=1)}^{n} |x_i - y_i|^p)^{(1/p)}$$

and determines the range within which observations are considered nearest neighbours, and k is the number of nearest neighbours to be considered when labelling new observations (Schliep and Hechenbichler, 2016). If k=1, only the nearest neighbour influences the predicted label of the unseen example. Conversely, a higher k increases the likelihood of a new case receiving the label of the most frequent class within the training set (Schliep and Hechenbichler, 2016).

The best values of distance and k were determined using cross-validated training and by setting the maximum possible value for k at 500, and trying multiple values for distance. There is discussion as to which order of k to choose. Following Maier et al. (2009), we tested several values for k, conducting post-hoc sensitivity analysis

with confusion matrices for the desired balance in prediction (Maier et al., 2009). A larger k should be more reliable, but our training showed a lower k actually led to less misclassification. The accuracy increased with an increasing value of k at first, but peaked at k=104, thereafter fluctuating without reaching a more optimal value. The most accurate model in training had parameters d=1, k=104, and a "triweight" kernel, which is a Gaussian function that attributes higher weights to data closer to an observation than any other kernel.

3.1.4 RANDOM FOREST

A Random Forest (RF) is an ensemble-based technique proposed by Breiman (2001) that attempts to minimize the limitations of classical decision trees by building multiple trees, each from a random subset of the original training data, considering only a random number of predictor variables at each split, and aggregating their results (Liaw and Wiener, 2002; Breiman, 2001; Boulesteix et al., 2012).

As a learning method, it is more robust to overfitting than normal decision trees, and shows good predictive performance even with considerable noise (Hastie et al., 2009; Biau, 2012). Computationally, it runs efficiently on large samples with a large number of input variables, and can accommodate different data scales, and categorical and continuous variables (Chen et al., 2004). It also allows for variables to be assessed and ranked with respect to their ability to predict the outcome variable, a feature which is later used in this paper due to its potential applicability in clinical practice (Boulesteix et al., 2012).

We built a model with specifications for parameters mtry, ntree, and class weights. ntree defines the number of trees built by the Random Forest algorithm, mtry defines the number of candidate variables randomly selected and tried at each split of a tree, and class weights attributes a prior probability to a certain class (Diaz-Uriarte and Alvarez de Andrés, 2006; Chen et al., 2004; Kuhn, 2016). To train the algorithm, we built parameter-value grids with mtry=8 (the squared root of the number of predictors), and ntree between 100 and 6000. A weight of 0.166 was assigned to the majority class "alive", placing a higher misclassification cost on the minority class.

For mtry=8, the highest AUROC was achieved with 6000 trees. The improvement in AUROC between lower and higher numbers of trees was mostly marginal and plateaued quickly, increasing only decimally after n=1000. We then computed the cross-validated prediction performance of these models and obtained the 20 most important variables (Fig. 2).

3.1.5 GRADIENT BOOSTED MACHINE

In the Gradient Boosting Machine (GBM) algorithm, new models (base-learners) are consecutively fitted to the training data set in order to provide a more accurate estimate of the outcome variable (Friedman, 2001). This is done by combining decision trees, and increasingly weighting the "difficult to predict" events to a greater degree. Using k-fold cross-validation, GBM will fit k models in order to compute the cross validation error estimate, followed by an additional, final model using all of the

data (Ridgeway, 2012). GBM is considered a strong model with potentially interesting clinically-oriented properties (Natekin and Knoll, 2013).

The GBM model was tuned during training for four hyper-parameters: n.trees (the number of trees (iterations) to be generated by the algorithm), n.minobsinnode (the minimum number of observations in the terminal nodes of a tree), interaction depth (defines the number of terminal nodes or leaves of a tree), and shrinkage (Kuhn, 2016). This parameter depends inversely on the number of iterations, and controls the learning rate of the algorithm by controlling the rate at which the boosting algorithm descends the error surface (Churpek et al., 2016; Ridgeway, 2012; Natekin and Knoll, 2013).

Using the stochastic gradient boosting method, we defined multiple parameter-value grids with the above-mentioned parameters, running consecutive post-hoc sensitivity analysis with 10-fold cross-validation to determine the best values for shrinkage between 0.01 and 0.0001, interaction depth between 1 and 8, and for n.trees up to a maximum of 15000 trees, and setting nminobsinnode=2. The most accurate model in training had shrinkage=0.001, ntrees=10600, and interaction depth=8.

3.2 Analysing variable importance

We obtained and plotted the relative importance of the most important predictor variables for the best RF and GBM models. Variable importance represents the impact of each feature on the accuracy of a model, and is calculated by excluding or permuting that variable and measuring how much this decreases the model's accuracy. The greater the decrease in accuracy, the greater the degree of association between the variable and the classification result (Khalilia et al., 2011).

3.3 Results

The accuracy of all models in predicting 5-year mortality after CABG was assessed by testing against the validation dataset, with results reported as AUROC (95%CI).

Cox Regression, the most commonly used survival analysis tool in Medicine, was used as a baseline for comparison and proved the least accurate of all models with a time-dependent AUROC of 0.644 at 5 years follow-up.

The AUROCs of all LR and ML models are shown in Figure 1. All ML based and LR models performed better than the Cox regression. The GBM (AUROC 0.767 [0.739–0.795]), RF (0.760 [0.734–0.793]) and SVM (0.759 [0.731–0.788]) clearly outperformed the rest (Fig. 1). The wkNN model performed the least well (0.712 [0.680–0.744]), falling below both untrained and trained LR models, with 0.719 [0.687–0.750] and 0.737 [0.709–0.768], respectively.

The table of two-sided tests for statistical significance shows that the RF and GBM models are significantly more accurate than untrained Logistic Regression ($p<0.05$ for both cases) (Table 3 Appendix 3). To obtain further measures of performance and confusion matrices for the best ML models, we defined four cut-offs (0.50, 0.28, 0.21, and 0.14) based on the actual mortality rate (5 year mortality rate: 14.2%), for the probability of dying of each individual patient to be counted as a negative event, so that all patients with a $>50\%$, $>28\%$, $>21\%$, and $>14\%$ predicted probability of dying

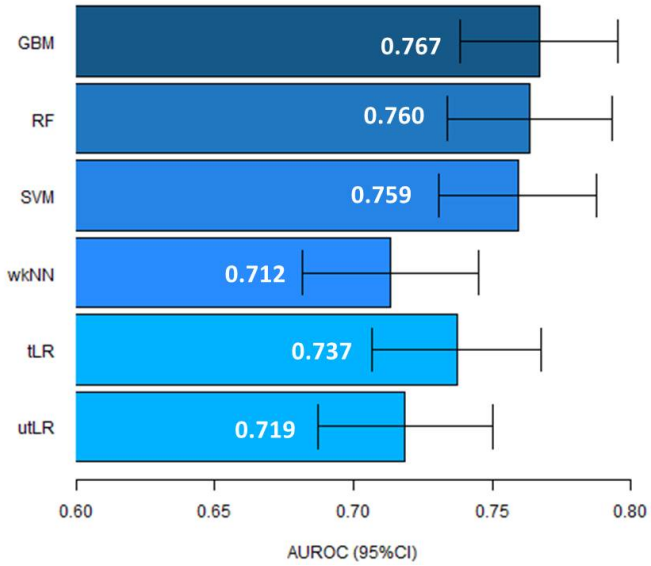

Figure 1: Area under the receiver operator characteristic (AUROC) curves with 95% CIs for all models, for 5-year Mortality prediction in the validation dataset.

were counted as such. As an example, a cut-off of 0.21 (an additional 50% risk of dying) results in a sensitivity of 83.5 and a specificity of 57.7, which offers a sufficiently reliable estimation of true negatives in a clinical setting (Table 1, and Fig. 4 Appendix 3).

Lastly, the top 20 predictors for the GBM and RF models are shown in Figure 2. GBM highlights the importance of a much wider array of parameters, but there is still some overlap between the models, with clinically established predictors like serum creatinine after surgery and age ranking highly in both. Interestingly, another kidney function related-parameter, serum urea at day 4 after operation, was the most important predictor in both.

4. Discussion

To our knowledge, this is the first study to describe the use of ML methods to predict long-term mortality in patients who underwent CABG. Here, we demonstrate the superiority of models developed with ML algorithms over traditional Logistic Regression for long-term mortality prediction after CABG operations. These findings are in line with the predictive capacity of ML models in other fields of Medicine (Churpek et al., 2016; Allyn et al., 2017; Taylor, 2016). Also, the superiority of GBM and RF in a classification problem like this, with pronouncedly imbalanced data, reflects the findings of previous studies, where both performed better than k-nearest neighbours (I. Brown and Mues, 2012).

Cut-off for RF model	0.14	0.21	0.28	0.50
Sensitivity (%)	65.7	83.5	91.4	98.9
Specificity (%)	71.5	57.7	44.7	16.2

Table 1: Sensitivity and specificity values for different cut-offs for the Random Forest model showing 0.21 provides the most clinically relevant balance between sensitivity and specificity

Furthermore, we obtained the list of variables used in the RF and GBM models ranked by their relative importance in predicting log-term mortality after CABG. Some of the top-ranked predictor variables relate to kidney injury and confirm earlier studies (Loef et al., 2009; Loef et al., 2005; Lassnigg et al., 2005). However, to our surprise, the most important predictor variable in both GBM and RF was urea measured at day 4 after surgery. Though related to kidney function, urea measurements had never clearly been identified as independent predictors of long-term mortality after CABG.

Previous studies identified a higher baseline urea level as best single predictor of mortality in patients admitted to a hospital with heart failure, regardless of the level of renal dysfunction, leading the authors to suggest that it should be considered a valid stand alone risk-stratification tool for physicians to closely monitor (Fonarow et al., 2005; Filippatos et al., 2007). Likewise, our findings based on ML approaches highlight a similar role for urea in the post-operative period for CABG patients.

Moreover, our exploratory data analysis also shows that predicting long-term mortality in large datasets of patients with complex clinical situations using Machine Learning-based prediction models can be done at least as successfully as the prediction of short-term mortality after cardiac surgery reported by other authors (Allyn et al., 2017).

The possibility of translating some of the complexity in the information provided by these models in a clinically useful way encourages further research into applications for intelligent clinical decision support systems. Especially when different models suggest unexpected associations between some predictor variables and outcome, a critical appraisal of the clinical utility of these parameters is certainly justified.

All efforts of attempting to predict long-term mortality ultimately aim at providing patients with the necessary care to lower this risk. ML algorithms provide “raw” probabilities for mortality for each individual patient, and having individual predicted mortality probabilities allows caregivers to better define the cut-offs for intervention within a specific cohort. In our cohort, patients had a 0.14 a priori likelihood of dying (corresponding to an overall 14.2% mortality rate). To translate our results into a clinically meaningful estimation of mortality risk for a patient in the early post-operative phase, consider a patient with a 50% increase in predicted mortality risk. Our study shows that applying this cut-off value (0.21) to the “raw” individual patient probabilities provided by the RF model achieves an important, balanced combination of specificity (58%) and sensitivity (84%). Such a combination limits the over-diagnosis associated with a cut-off equal to the general cohort mortality rate of 0.14, and reduces the number of false negatives (that is, of patients who died while being predicted not to) when compared to higher cut-off values.

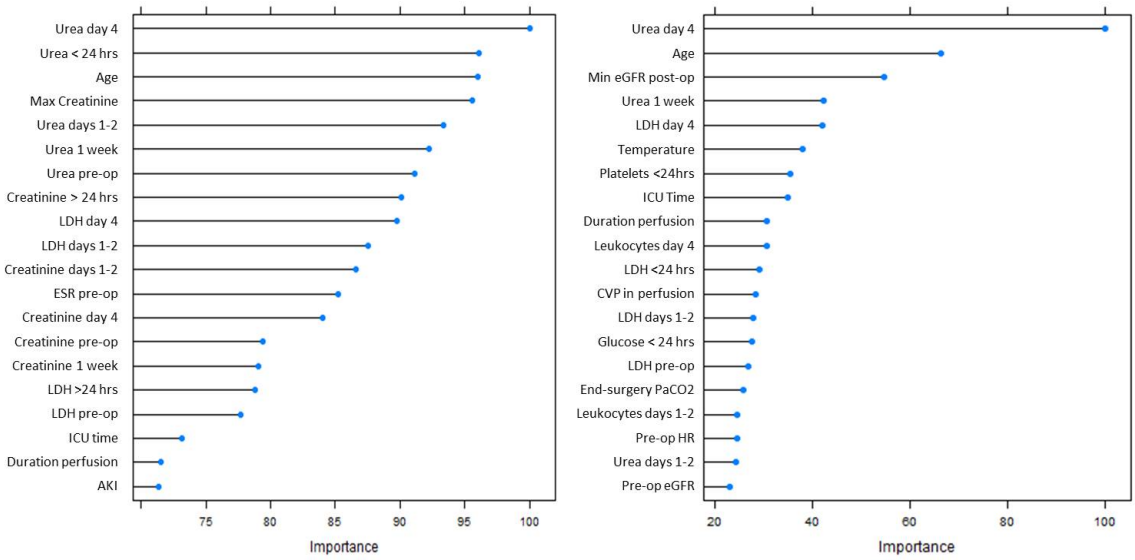

Figure 2: Variable importance in the Random Forest and GBM models, scaled to a maximum of 100.

In conclusion, ML-based models are a more accurate method of predicting long-term mortality in individual patients undergoing CABG, especially when using RF and GBM. There is great potential in the use of measurements of variable importance obtained from these algorithms to discover unclear associations between and to better understand the influence of certain clinical parameters and long-term mortality, and thus help guide the development of therapeutic strategies to optimize long-term outcome. Finally, the results of this study demonstrate that data which are already routinely available during and after CABG operations can be used to develop ML models to predict long-term mortality. As such, ML-based models may form the basis to build intelligent decision support systems for clinical use.

References

[1] S Aranki and JM Aroesty. Coronary artery bypass graft surgery: long-term clinical outcomes, 2017. URL: https://http://www.uptodate.com/contents/coronary-artery-bypass-graft-surgery-long-term-clinical-outcomes?source=search_result&search=Coronary+artery+bypass+graft+surgery%5C%3A+Long-term+clinical+outcomes&selectedTitle=1~150.

[2] BG Loef, AH Epema, G Navis, T Ebels, and CA Stegeman. Postoperative renal dysfunction and preoperative left ventricular dysfunction predispose patients to increased long-term mortality after coronary artery bypass graft surgery. *Br J Anaesth*, 102(6):749–755, 2009.

[3] A Lassnigg, D Schmidlin, M Mouhieddine, LM Bachmann, W Druml, P Bauer, and M Hiesmayr. Minimal changes of serum creatinine predict prognosis in pa-

tients after cardiothoracic surgery: a prospective cohort study. *J Am Soc Nephrol*, 15(6):1597–1605, 2005.

[4] BG Loef, AH Epema, TD Smilde, RH Henning, T Ebels, G Navis, and CA Stegeman. Immediate postoperative renal function deterioration in cardiac surgical patients predicts in-hospital mortality and long-term survival. *J Am Soc Nephrol*, 16(1):195–200, 2005.

[5] BD Westenbrink, L Kleijn, RA de Boer, JG Tijssen, WJ Warnica, R Baillot, JL Rouleau, and WH van Gilst. Sustained postoperative anaemia is associated with an impaired outcome after coronary artery bypass graft surgery: insights from the imagine trial. *Heart*, 97(19):1590–1596, 2011.

[6] DM Shahian, SM O'Brien, S Sheng, FL Grover, JE Mayer, and JP Jacobs. Predictors of long-term survival after coronary artery bypass grafting surgery: results from the society of thoracic surgeons adult cardiac surgery database (the ASCERT study). *Circulation*, 125(12):1491–1500, 2012.

[7] MM Churpek, TC Yuen, C Winslow, DO Meltzer, MW Kattan, and DP Edelson. Multicenter comparison of machine learning methods and conventional regression for predicting clinical deterioration on the wards. *Crit Care Med*, 44(2):368–374, 2016.

[8] RC Deo. Machine learning in medicine. *Circulation*, 132(20):1920–1930, 2015.

[9] J Allyn, N Allou, P Augustin, I Philip, O Martinet, and M Belghiti. A comparison of a machine learning model with euroscore ii in predicting mortality after elective cardiac surgery: a decision curve analysis. *PLoS One*, 12(1), 2017.

[10] R Diaz-Uriarte and S Alvarez de Andrés. Gene selection and classification of microarray data using random forest. *BMC Bioinformatics*, 7(3), 2006.

[11] M Eslam, AM Hashem, M Romero-Gomez, T Berg, GJ Dore, and A Mangia. Fibrogene: a gene-based model for staging liver fibrosis. *J Hepatol*, 64(2):390–398, 2016.

[12] MJ Wolf, EK Lee, SC Nicolson, GD Pearson, MK Witte, J Huckaby, M Gaies, LS Shekerdemian, and WT Mahle. Rationale and methodology of a collaborative learning project in congenital cardiac care. *Am Heart J*, 174:129–137, 2016.

[13] M Legrand, R Pirracchio, A Rosa, ML Petersen, M Van der Laan, JN Fabiani, MP Fernandez-Gerlinger, and I Podglajen. Incidence, risk factors and prediction of post-operative acute kidney injury following cardiac surgery for active infective endocarditis: an observational study. *Crit Care*, 17(5), 2013.

[14] AJ Epstein, D Polsky, F Yang, L Yang, and PW Groeneveld. Coronary revascularization trends in the united states: 2001–2008. *JAMA*, 305(17):1769–1776, 2011.

[15] NHS. Coronary artery bypass graft - nhs choices, 2017. URL: `http://www.nhs.uk/conditions/Coronary-artery-bypass/Pages/Introduction.aspx`.

[16] S van Buuren, HC Boshuizen, and SA Reijneveld. Toward targeted hypertension screening guidelines. *Med Decis Making*, 26:145–153, 2006.

[17] S van Buuren and K Groothuis-Oudshoorn. Mice: multivariate imputation by chained equations in r. *Journal of Statistical Software*, 45(3):1–67, 2011.

[18] AD Shah, JW Bartlett, J Carpenter, O Nicholas, and H Hemingway. Comparison of random forest and parametric imputation models for imputing missing data using mice: a caliber study. *Am J Epidemiol*, 179(6):764–774, 2014.

[19] MJ Azur, EA Stuart, C Frangakis, and PJ Leaf. Multiple imputation by chained equations: what is it and how does it work? *Int J Methods Psychiatr Res*, 20(1):40–49, 2011.

[20] JL Schafer and JW Graham. Missing data: our view of the state of the art. *Psychological Methods*, 7:147–177, 2002.

[21] K Hechenbichler and K Schliep. Weighted k-nearest-neighbor techniques and ordinal classification, 2004. URL: `http://www.stat.uni-muenchen.de/sfb386/papers/dsp/paper399.ps`.

[22] C-W Hsu, C-C Chang, and C-J Lin. A practical guide to support vector classification, 2016. URL: `http://www.csie.ntu.edu.tw/~cjlin/papers/guide/guide.pdf`.

[23] A Field, J Miles, and Z Field. *Discovering Statistics Using R*. SAGE Publications, California, 2012, page 871.

[24] L Thabane, L Mbuagbaw, S Zhang, Z Samaan, M Marcucci, and C Ye. A tutorial on sensitivity analyses in clinical trials: the what, why, when and how. *BMC Medical Research Methodology*, 13:92, 2013.

[25] T Hastie, R Tibshirani, and JH Friedman. *The Elements of Statistical Learning*. New York, NY, 2009.

[26] O Chapelle and A Zien. Semi-supervised classification by low density separation. *Proceedings of the Tenth International Workshop on Artificial Intelligence and Statistics (AISTATS 2005)*:57–64, 2005.

[27] R Batuwita and V Palade. Class imbalance learning methods for support vector machines. In H He and Y Ma, editors, *Imbalanced Learning: Foundations, Algorithms, and Applications*, part 5, pages 83–96. John Wiley Sons, New Jersey, 2012.

[28] R Akbani, S Kwek, and N Japkowicz. Applying support vector machines to imbalanced datasets. *Proceedings of the 15th European Conference on Machine Learning*:39–50, 2004.

[29] M Maier, M Hein, and U von Luxburg. Optimal construction of k-nearest-neighbor graphs for identifying noisy clusters. *Theor Comput Sci*, 410:1749–1764, 2009.

[30] A Liaw and M Wiener. Classification and regression by randomforest. *R News*, 2(3):18–22, 2002.

[31] L Breiman. Random forests. *Machine Learning*, 45(1):5–32, 2001.

[32] AL Boulesteix, S Janitza, J Kruppa, and IR Konig. Overview of random forest methodology and practical guidance with emphasis on computational biology and bioinformatics. *WIREs Data Mining Knowl Discov*, 2(6):493–507, 2012.

[33] G Biau. Analysis of a random forests model. *Journal of Machine Learning Research*, 13:1063–1095, 2012.

[34] C Chen, A Liaw, and L Breiman. Using random forest to learn imbalanced data, 2004.

[35] JH Friedman. Greedy function approximation: a gradient boosting machine. *Annals of Statistics*, 29(5):1189–1232, 2001.

[36] G Ridgeway. Generalized boosted models: a guide to the gbm package, 2012. URL: https://pdfs.semanticscholar.org/a3f6/d964ac323b87d2de3434b23444cb774a216e.pdf.

[37] A Natekin and A Knoll. Gradient boosting machines, a tutorial. *Front Neurorobot*, 7, 2013.

[38] M Khalilia, S Chakraborty, and M Popescu. Predicting disease risks from highly imbalanced data using random forest. *BMC Medical Informatics and Decision Making*, 11, 2011.

[39] ER DeLong, DM DeLong, and DL Clarke-Pearson. Comparing the areas under two or more correlated receiver operating characteristic curves: a nonparametric approach. *Biometrics*, 44(3):837–845, 1988.

[40] RA Taylor. Prediction of in-hospital mortality in emergency department patients with sepsis: a local big data-driven, machine learning approach. *Acad Emerg Med*, 23(3):269–278, 2016.

[41] I Brown and C Mues. An experimental comparison of classification algorithms for imbalanced credit scoring data sets. *Expert Systems with Applications*, 39(3):3446–3453, 2012.

[42] GC Fonarow, KF Adams, and WT Abraham. Risk stratification for in-hospital mortality in acutely decompensated heart failure: classification and regression tree analysis. *JAMA*, 293(5):572–580, 2005.

[43] G Filippatos, J Rossi, DM Lloyd-Jones, WG Stough, J Ouyang, and DD Shin. Prognostic value of blood urea nitrogen in patients hospitalized with worsening heart failure: insights from the acute and chronic therapeutic impact of a vasopressin antagonist in chronic heart failure (activ in chf) study. *J Card Fail*, 13(5), 2007.

[44] M Kuhn. Caret: classification and regression training - r package version 6.0-73, 2016. URL: https://CRAN.R-project.org/package=caret.

[45] K Schliep and K Hechenbichler. Kknn: weighted k-nearest neighbors - r package version 1.3.1, 2016. URL: https://CRAN.R-project.org/package=kknn.

[46] N Xiao, Q Xu, and M-Z Li. Hdnom: building nomograms for penalized cox models with high-dimensional survival data, 2016. URL: http://biorxiv.org/content/early/2016/08/23/065524.

[47] C Brown. Dummies: create dummy/indicator variables flexibly and efficiently - package version 1.5.6, 2012. URL: https://CRAN.R-project.org/package=dummies.

[48] G Ridgeway. Gbm: generalized boosted regression models - r package version 2.1.3, 2017. URL: https://CRAN.R-project.org/package=gbm.

[49] X Robin, N Turck, A Hainard, N Tiberti, H Lisacek, JC Sanchez, and M Müller. Proc: an open-source package for r and s+ to analyze and compare roc curves. *BMC Bioinformatics*, 12, 2011.

[50] Austria R Development Core Team - R Foundation for Statistical Computing Vienna. R: a language and environment for statistical computing, 2008. URL: http://www.R-project.org.

Appendices

Appendix I – Imputing the missing data

MICE can handle missing data both at random (MAR) and not at random[17]. Since we assume the missing data from our original dataset were MAR, the spread of the residuals for any imputed variable should be similar for observed and imputed data[17]. Figure 3 shows the residuals of the regression of "Creatinine at 1 week after surgery" (Kreat1) on its propensity score (the average over the imputations). It's clear that the distributions of observed and imputed data are a good fit, and the imputation is good.

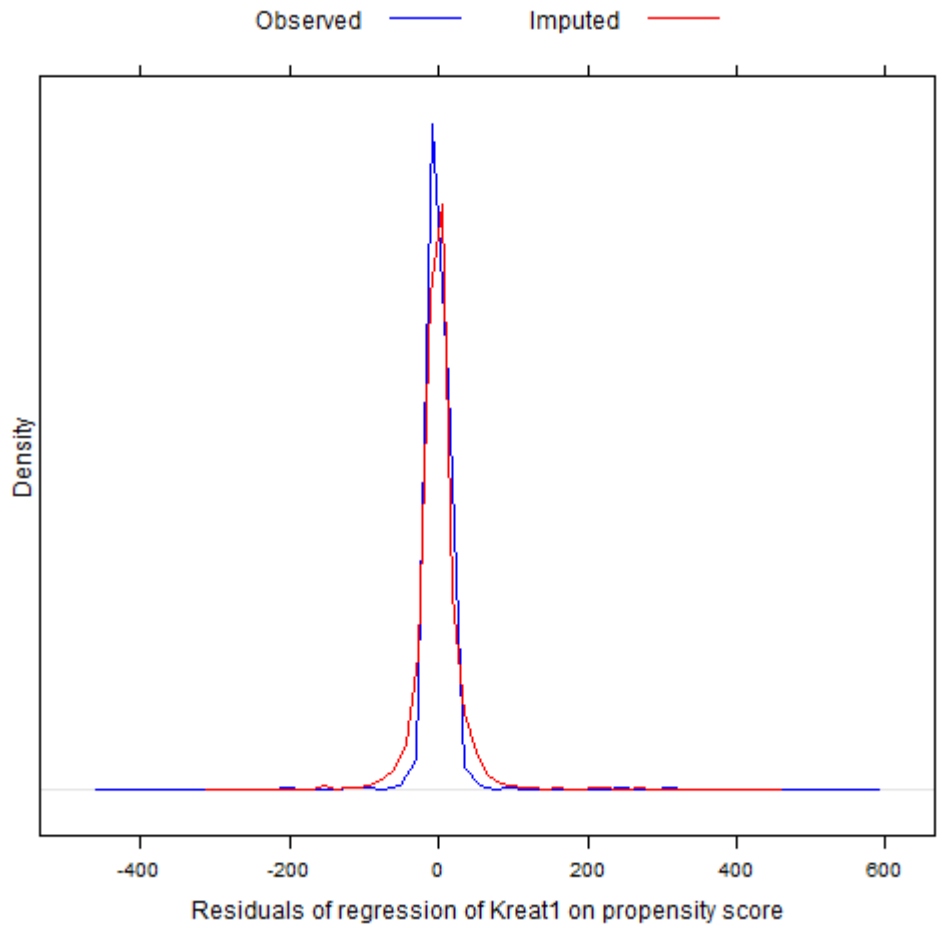

Figure 3. Residuals of regression of Creatinine at week 1 on propensity score

Appendix II – Data tables

Table 1. Descriptives of all the variables in the dataset.

Patient characteristics	Age (years)	65.7 (65.5-66.0)
	Male gender	4471 (76.2%)
Perioperative parameters	Minimum Body Temperature (°C)	31.79 (31.74-31.84)
	Maximum Flow	4.33 (4.28-4.38)
	Duration of perfusion (min)	102.8 (101.7-103.9)
	Duration of cardiac arrest (min)	0.140 (0.074 - 0.217)
	Duration of aortic cross-clamp (min)	58.67 (57.96-59.38)
	ICU time (days)	58.8 (54.3-63.4)
	ICU category	
	0 = Less than 19 days	1500 (25.6%)
	1 = 19-22 days	1397 (23.8%)
	2 = 22-39 days	1671 (28.5%)
	3 = More than 39 days	1300 (22.1%)
Blood values	ESR before surgery (mm/hour)	21.5 (21.0-22.00)
	LDH before surgery (U/L)	211.0 (208.2-213.7)
	LDH within 24 hours after surgery	364.8 (351.8-317.9)
	LDH 1-2 days after surgery	360.6 (351.7-368.4)
	LDH at day 4 after surgery	328.0 (309.8-346.3)
	Haemoglobin before surgery (mmol/L)	8.20 (8.14-8.26)
	Haemoglobin within 24 hours after surgery	6.39 (6.37-8.41)
	Haemoglobin at 1-2 days after surgery	6.26 (6.24-6.28)
	Haemoglobin at day 4 after surgery	6.48 (6.44-6.52)
	Haemoglobin at 1 week after surgery	8.42 (8.39-8.45)
	Leucocytes before surgery ($x10^9$/L)	8.14 (8.06-8.22)
	Leucocytes within 24 hours after surgery	8.14 (8.06-8.22)
	Leucocytes 1-2 days after surgery	17.0 (16.9-17.1)
	Leucocytes at day 4 after surgery	13.5 (9.5-17.6)
	Leucocytes at 1 week after surgery	7.95 (7.88-8.02)
	Thrombocytes before surgery ($x10^9$/L)	239 (237-241)
	Thrombocytes within 24 hours after surgery	165 (163-166)
	Thrombocytes at 1 week after surgery	247 (245-249)
	Neutrophils within 24 hours after surgery ($x10^9$/L)	12.1 (12.0-12.2)
	Lymphocytes within 24 hours after surgery ($x10^9$/L)	1.12 (1.07-1.18)
	Blood glucose within 24 hours after surgery (mmol/L)	10.9 (10.9-11.0)
Haemodynamic parameters	Heart rate before surgery (beats/min)	63.9 (63.5-64.2)
	Heart rate during perfusion	63.3 (61.9-64.7)
	Systolic BP before surgery (mmHg)	112.3 (111.5-113.2)
	Systolic BP during perfusion	62.4 (61.9-63.0)
	Diastolic BP before surgery (mmHg)	64.3 (63.5-65.1)
	Diastolic BP during perfusion	56.4 (56.00-56.9)
	CVP before surgery (mmHg)	12.7 (11.9-13.5)

	CVP during perfusion	6.6 (6.4-6.8)
	O_2 saturation before surgery (%)	98.1 (98.1-98.2)
	O_2 saturation during perfusion	99.0 (99.0-99.1)
	O_2 saturation at the end of surgery	97.9 (97.8-98.0)
	PaO_2 before surgery (kPa)	21.4 (21.0-21.7)
	PaO_2 during perfusion	26.7 (26.5-27.0)
	PaO_2 at the end of surgery	18.0 (17.8-18.4)
	$PaCO_2$ before surgery (kPa)	5.02 (5.01-5.04)
	$PaCO_2$ during perfusion	5.05 (5.04-5.07)
	$PaCO_2$ at the end of surgery	4.87 (4.85-4.88)
Renal function parameters	Creatinine before surgery (μmol/l)	102.2 (100.5-104.0)
	Creatinine within 24 hours after surgery	93.0 (91.3-94.6)
	Creatinine at days 1-2 after surgery	101.3 (99.7-102.9)
	Creatinine at day 4 after surgery	97.8 (96.1-99.3)
	Creatinine at 1 week after surgery	103.1 (101.3-105.0)
	Absolute difference in creatinine	9.3 (8.3-10.3)
	Maximum creatinine	111.5 (109.6-113.5)
	Relative difference in creatinine	1.13 (1.10-1.16)
	Post-operative AKI grade	
	0	5242 (89.3%)
	1	503 (8.6%)
	2	57 (1.0%)
	3	66 (1.1%)
	eGFR before surgery (ml/min/1.73m^2)	80.1 (78.9-81.3)
	Minimum eGFR after surgery	75.0 (74.2-75.7)
	Absolute difference in eGFR	-5.57 [(-6.92) – (-4.21)]
	Relative difference in eGFR	0.952 (0.947-0.956)
	Urea before surgery (mmol/L)	7.42 (7.26-7.59)
	Urea within 24 hours after surgery	12.3 (11.3-13.3)
	Urea at 1-2 days after surgery	11.5 (10.7-12.3)
	Urea at day 4 after surgery	9.88 (8.93-10.82)
	Urea at 1 week after surgery	7.10 (7.01-7.19)
Liver function parameters	ALAT before surgery (U/L)	40.3 (39.3-41.3)
	ALAT within 24 hours after surgery	45.0 (39.9-50.1)
	ALAT at 1-2 days after surgery	44.6 (39.2-50.1)
	ASAT before surgery (U/L)	34.8 (33.9-34.8)
	ASAT within 24 hours after surgery	76.3 (70.1-82.5)
	ASAT at 1-2 days after surgery	68.5 (61.3-75.7)
	ASAT at day 4 after surgery	54.2 (45.5-63.0)
Outcome variable	5-year Mortality	833 (14.2%)
	Follow-up time (days)	3173 (3127-3219)

ESR=Erythrocyte Sedimentation Rate, ALAT=Alanine Aminotransferase, ASAT=Aspartate Aminotransferase, LDH=Lactate Dehydrogenase, eGFR=Estimated Glomerular Filtration Rate, CVP= Central venous pressure, AKI= Acute Kidney Injury. Data are presented as mean (95%CI) in the case of continuous variables, or as the number of cases and percentage (%) in the case of categorical variables.

Appendix III – Confusion matrices for the predictions of individual models and DeLong significance test

Table 3. Two-sided p-values for comparison between AUROCs of all models

	utLR	**tLR**	**SVM**	**wkNN**	**RF**	**GBM**
utLR		0.41	0.063	0.82	**0.044***	**0.026****
tLR	0.41		0.012$^{\$\$}$	0.13	0.23	0.16
SVM	0.063	0.012$^{\$\$}$		0.0017$^{\$\$}$	0.85	0.70
wkNN	0.82	0.13	0.0017$^{\$\$}$		0.025**	0.014**
RF	**0.044****	0.23	0.85	0.025**		0.86
GBM	**0.026****	0.16	0.70	0.014**	0.86	

utLR = untrained Logistic Regression, tLR = trained Logistic Regression, SVM = Support Vector Machine, wkNN = Weighted k-Nearest Neighbours, RF = Random Forest, GBM = Gradient Boosted Machine

** denotes statistical significance for uncorrelated curves ($p < 0.05$)

$^{\$\$}$ denotes statistical significance for correlated curves ($p < 0.05$)

Figure 4. Confusion matrices of the tLR, SVM, RF, and GBM models

	Reference (actual patient outcomes)	
Prediction by GBM with cut-off of 0.50	Alive	Deceased
Alive	1936	309
Deceased	78	24
Prediction by GBM with cut-off of 0.28	Alive	Deceased
Alive	1799	280
Deceased	215	53
Prediction by GBM with cut-off of 0.21	Alive	Deceased
Alive	1718	268
Deceased	296	65
Prediction by GBM with cut-off of 0.14	Alive	Deceased
Alive	1538	234
Deceased	476	99

Table 4.1.1-4.1.4 – Confusion matrices for the trained Gradient Boosted Machine model

	Reference (actual patient outcomes)	
Prediction by SVM with cut-off of 0.50	Alive	Deceased
Alive	1988	286
Deceased	26	47
Prediction by SVM with cut-off of 0.28	Alive	Deceased
Alive	1853	207
Deceased	161	126
Prediction by SVM with cut-off of 0.21	Alive	Deceased
Alive	1723	173
Deceased	291	160

Prediction by SVM with cut-off of 0.14	Alive	Deceased
Alive	1412	108
Deceased	602	225

Table 4.2.1-4.2.4 – Confusion matrices for the Support Vector Machine model

	Reference (actual patient outcomes)	
Prediction by tLR with cut-off of 0.50	Alive	Deceased
Alive	1972	281
Deceased	42	52
Prediction by tLR with cut-off of 0.28	Alive	Deceased
Alive	1861	214
Deceased	153	119
Prediction by tLR with cut-off of 0.21	Alive	Deceased
Alive	1723	173
Deceased	291	160
Prediction by tLR with cut-off of 0.14	Alive	Deceased
Alive	1457	119
Deceased	557	214

Table 4.3.1-4.3.4 – Confusion matrices for the trained Logistic Regression model

	Reference (actual patient outcomes)	
Prediction by RF with cut-off of 0.50	Alive	Deceased
Alive	1992	279
Deceased	22	54
Prediction by RF with cut-off of 0.28	Alive	Deceased
Alive	1841	184
Deceased	173	149
Prediction by RF with cut-off of 0.21	Alive	Deceased
Alive	1682	141
Deceased	332	192
Prediction by RF with cut-off of 0.14	Alive	Deceased
Alive	1324	95
Deceased	690	238

Table 4.4.1-4.4.4 – Confusion matrices for the Random Forest model