Appendix 1

Classification algorithms, hyperparameter tuning, and methods of summarizing time series data used in machine learning for prediction of postoperative complications among patients undergoing thoracic surgery using one-lung ventilation.

Random forests

Random forests are used commonly in both classification and regression (1). They are often used for supervised classification techniques, which are a combination of decision tree predictors (2). Random forests are a collection of unpruned classification and regression decision trees, which combines the decision of individual decisions trees using bagging and feature randomness. An individual decision tree makes a decision by proceeding from the root to a lead, and the entire random forest makes a decision by majority vote.

An example of a decision tree is depicted in *Figure S1*. Internal nodes of a decision tree are a test and leaf nodes are the ultimate outcome of evaluating several of these tests. Testing begins at the root node, which is the uppermost node on the tree. As in *Figure S1*, if a patient has a diffusing capacity of carbon monoxide of 100, then the decision tree will follow the next internal node, which evaluates whether a fraction of inspired oxygen is greater than or equal to 70. The decision tree will continue to proceed through each node until the terminal verdict is reached.

In training, each tree in a random forest relies on a subset of the features of an instance sample independently. As such, random forests connect randomized decision trees and provides an average prediction from them (1). They construct n individual decision trees where n is a value that the user can adjust (i.e. hyperparameter), and each decision trees makes a class prediction. Then, the random forest considers all of the results of decision trees, and the class with the maximum number of correct predictions becomes the final prediction of the model. Random forests work efficiently when the number of cases is much less than the number of features, which is in keeping with the characteristics in our study. Random forests also require all individuals to have the same number of features.

https://dx.doi.org/10.21037/jtd-2025-1-2513

Figure S1 Flow chart depicting an example of a decision tree used to characterize postoperative complications from perioperative variables. CCI, Charlson Comorbidity Index; DLCO, diffusing capacity for carbon monoxide; EtCO2, end-tidal carbon dioxide; FEV1, forced expiratory volume in one second; FiO2, fraction of inspired oxygen; HR, heart rate. This figure was reused/adapted with the permission from Ul Aftab's Master's degree thesis: Aftab, Atif Ul. Predicting Perioperative Outcomes from Surgical Data during One Lung Ventilation. 2020 (3).

https://dx.doi.org/10.21037/jtd-2025-1-2513

Support vector machines

Support vector machines are commonly used for classification (4). It has been shown to achieve satisfactory results in medical diagnostics, optical character identification, and forecasting among other areas (5). This algorithm is a supervised learning technique that requires labeled training sets (6,7). It is also efficient for large datasets and those with diverse variables, such as within biological data. As with random forests, support vector machines require that all data to have the same number of features. For a given set of inputs, support vector machines attempt to find a decision surface for separating two classes. Data points that are closest to the decision surface are called support vectors. In the most basic case, the decision surface is a hyperplane. Support vector machines maximize the boundary between the separating hyperplane and instances, then finds an optimal solution among the multitude of possible solutions. New instances are classified by the side of the decision boundary that the new instance lies on.

An example of this is shown in *Figure S2*. This is a two-dimensional case. As such, the instances have two features and are plotted as points in the two-dimensional plane. The objective of the support vector machine is to find an optimal line that divides the dataset into two different classes. After finding the decision boundary, the support vector machine identifies the near points to the decision boundary. These points are the support vectors. Then, the support vector machine measures the distance between the line and the support vectors. It attempts to maximize the distance between them to characterize the optimal hyperplane.

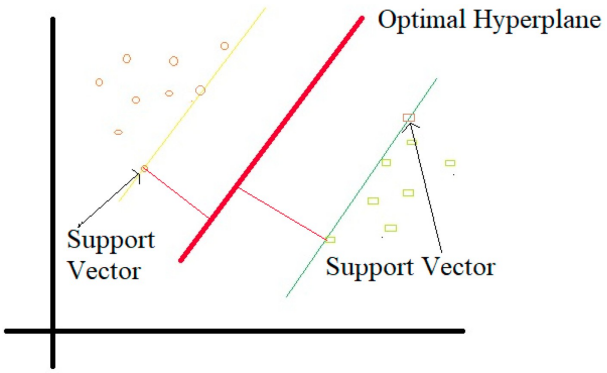

Figure S2 Example of a two-dimensional support vector machine. This figure was reused/adapted with the permission from Ul Aftab's Master's degree thesis: Aftab, Atif Ul. Predicting Perioperative Outcomes from Surgical Data during One Lung Ventilation. 2020 (3).

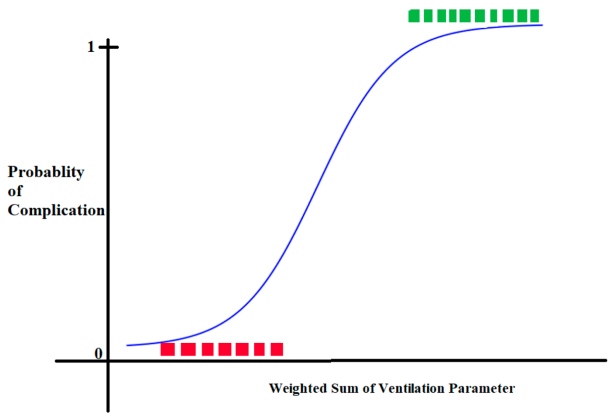

Figure S3 Example of a logistic regression function between the weighted sum of ventilation parameters and the probability of experiencing a complication. This figure was reused/adapted with the permission from Ul Aftab's Master's degree thesis: Aftab, Atif Ul. Predicting Perioperative Outcomes from Surgical Data during One Lung Ventilation. 2020 (3).

Logistic regression

Logistic regression is a predictive analysis used when the outcome class is binary. It builds models based on the logistic function and describes the connection between binary classification and feature variables. Outcomes are assessed based on previous knowledge or information by identifying the most critical relationship between the different circumstances and their consequences (8). As an example, we will consider a model where one must identify the postoperative outcome of a surgery for a given set of ventilation parameters as input features. Here, the outcome of interest is the occurrence of complications (yes/no). Logistic regression will first calculate the weighted sum of the ventilation parameters. Once this is complete, the regression analysis will input the weighted sum into a sigmoid function, which is the logistic function. This will yield a probability value. The probability value is then converted into a binary outcome by use of a threshold. For a new instance, the probability of that instance is calculated and then assigned to a class. The formula for the sigmoid function is:

$$sigmoid(x) = \frac{1}{1+e^{-x}}$$

This function is depicted in *Figure S3*. The x-axis represents the weighted sum of the ventilation parameters. The y-axis represents the probability of a complication.

https://dx.doi.org/10.21037/jtd-2025-1-2513

Plotting the sigmoid function results in an S-curve. If the logistic regression encounters an outlier that does not fit a particular data group, it will be processed using the logistic function.

Hyperparameter tuning

In machine learning, the value employed to regulate the training process of a model is called the hyperparameter. Hyperparameter tuning identifies optimized hyperparameters for a specific model, as different machine learning models have different hyperparameters. To find a machine learning model with the best accuracy, it is essential to supply the optimum value of hyperparameters to the model. Our study used Grid Search Cross Validation for hyperparameter tuning (9,10). Grid Search Cross Validation finds the optimal hyperparameters by using cross-validation and exhaustively checking all the possible values of those hyperparameters in that model.

Summarizing time series data

For each of the patients, the percentage of time of the surgery duration above a given threshold for different intraoperative variables was delineated. These variables were: heart rate, fraction of inspired oxygen, mean arterial blood pressure, arterial systolic blood pressure, positive end-expiratory pressure, peak inspiratory pressure, and end-tidal carbon dioxide. Given pre-determined threshold, T, the percentage of time that a feature was above or below T was identified. T for heart rate in beats per minute were >120, >110, >100, and <60. T for fraction of inspired oxygen was >70%. T for mean arterial blood pressure was <60mmHg. T for systolic arterial blood pressure was <80mmHg. T for positive end-expiratory pressure was >10mmHg. T for peak inspiratory pressure was >30. T for end-tidal carbon dioxide was >45mmHg and <25mmHg.

The processed data was represented by a value between 0 and 1. A variable would achieve 1 if it met a given T for the entire surgery. Conversely, a variable would achieve 0 if it did not meet a given T for the entire surgery. Fraction values between 0 to 1 were attributed to each variable representative of the percentage of time within T.

References

1. Biau G, Scornet E. A random forest guided tour. TEST: An Official Journal of the Spanish Society of Statistics and Operations Research 2016;25:197-227.
2. Breiman L. Random Forests. Machine Learning 2001;45:5-32.
3. Ul Aftab A. Predicting Perioperative Outcomes from Surgical Data during One Lung Ventilation. University of Manitoba, 2020. Available online: https://mspace.lib.umanitoba.ca/server/api/core/bitstreams/fc22b1b2-59e5-4768-8bbd-a1735c39d777/content
4. Hsu CW, Chang CC, Lin CJ. A Practical Guide to Support Vector Classification. Department of Computer Science, National Taiwan University; 2003.
5. Auria L, Moro RA. Support Vector Machines (SVM) as a Technique for Solvency Analysis. IDEAS Working Paper Series from RePEc; 2008.
6. Boser BE, Guyon IM, Vapnik VN. A training algorithm for optimal margin classifiers. Proceedings of the Fifth Annual Workshop on Computational Learning Theory, New York, NY, USA: Association for Computing Machinery; 1992:144-52.
7. Noble WS. What is a support vector machine? Nat Biotechnol 2006;24:1565-7.
8. Tolles J, Meurer WJ. Logistic Regression: Relating Patient Characteristics to Outcomes. JAMA 2016;316:533-4.
9. Pedregosa F, Varoquaux G, Gramfort A, et al. Scikit-learn: Machine Learning in Python. Journal of Machine Learning Research 2011;12:2825-30.
10. Buitinck L, Louppe G, Blondel M, et al. API design for machine learning software: experiences from the scikit-learn project. European Conference on Machine Learning and Principles and Practices of Knowledge Discovery in Databases, Prague, Czech Republic; 2013.

https://dx.doi.org/10.21037/jtd-2025-1-2513