

Data Science Program

Capstone Report - Spring 2024

Predicting Blood Transfusions for Coronary Artery Bypass Graft Patient

Jenny Hsiao-Tien Tsai,

Jichong Wu,

Puneet Gupta

Supervised by

Amir Jafari

**Abstract**

**Contents**

[1 Introduction 4](#_Toc162647345)

[2 Problem Statement & Project Objectives 4](#_Toc162647346)

[3 Related Work 4](#_Toc162647347)

[4 Solution and Methodology 5](#_Toc162647348)

[5 Results and Discussion 6](#_Toc162647349)

[5.1 Model selection and tuning 6](#_Toc162647350)

[5.2 Results and interpretation 17](#_Toc162647351)

[6 Discussion - understanding what features have higher impacts on model prediction 18](#_Toc162647352)

[7 Conclusion 24](#_Toc162647353)

[8 References 24](#_Toc162647354)

[9 Appendix 25](#_Toc162647355)

**List of Tables**

[**Table 1** Summary of Iteration #1 Setup 8](#_Toc162647552)

[**Table 2** Model Results from Iteration #1 8](#_Toc162647553)

[**Table 3** Summary of Iteration #2 Setup 9](#_Toc162647554)

[**Table 4** Model Results from Iteration #2 9](#_Toc162647555)

[**Table 5** Summary of Iteration #3 Setup 12](#_Toc162647556)

[**Table 6** Model Results from Iteration #3 12](#_Toc162647557)

[**Table 7** Summary of Iteration #4 Setup 13](#_Toc162647558)

[**Table 8** Model Results from Iteration #4 13](#_Toc162647559)

[**Table 9** Summary of Iteration #5 Setup 14](#_Toc162647560)

[**Table 10** Model Results from Iteration #5 14](#_Toc162647561)

[**Table 11** Summary of Iteration #6 Setup and results 15](#_Toc162647562)

[**Table 12** Summary of Iteration #7 Setup 16](#_Toc162647563)

[**Table 13** Summary of Iteration #7 Setup and results 16](#_Toc162647564)

[**Table 14** Summary of Iteration #8 Setup 17](#_Toc162647565)

[**Table 15** Summary of Iteration #8 Setup and results 17](#_Toc162647566)

[**Table 16** Best Performing Models – Gradient Boosting vs Random Forest vs XGBoost from Iteration #3 18](#_Toc162647567)

[**Table 17** Top 2 Models Comparison in All Iterations 18](#_Toc162647568)

[**Table 18** TOP 20 Important Features and Their Impacts on Gradient Boosting Model Prediction 21](#_Toc162647569)

**List of Figures**

[**Figure 1** Research Strategy of the Project 8](#_Toc162647583)

[**Figure 2** ROC Plot with 10 Selected Models from Iteration #1 10](#_Toc162647584)

[**Figure 3** ROC Plot with 10 Selected Models from Iteration #2 11](#_Toc162647585)

[**Figure 4** Top Correlation Feature Pairs in Iteration #3 12](#_Toc162647586)

[**Figure 5** Variance Inflation Factor (VIF) Values of All Features (n = 41) in Iteration #3 12](#_Toc162647587)

[**Figure 6** ROC Plot with 10 Selected Models from Iteration #3 14](#_Toc162647588)

[**Figure 7** ROC Plot with 10 Selected Models from Iteration #4 15](#_Toc162647589)

[**Figure 8** ROC Plot with 10 Selected Models from Iteration #5 16](#_Toc162647590)

[**Figure 9** ROC Plot with 10 Selected Models from Iteration #7 18](#_Toc162647591)

[**Figure 10** ROC Plot with 10 Selected Models from Iteration #8 19](#_Toc162647592)

[**Figure 11** Top 2 Models Comparison in All Iterations 20](#_Toc162647593)

[**Figure 12** Feature Importance TOP 20 from Gradient Boosting in Iteration #3 vs Iteration #7 21](#_Toc162647594)

[**Figure 13** Beeswarm Plot of Important Feature Relationships from Gradient Boosting from Iteration #7 22](#_Toc162647595)

[**Figure 14** Relationship Between DISCANCR and DIABETES and Their Impact on Prediction 23](#_Toc162647596)

[**Figure 15** Relationship Between PRINR and PRBUN and Their Impact on Prediction 23](#_Toc162647597)

[**Figure 16** Relationship Between OPTIME and DIALYSIS and Their Impact on Prediction 23](#_Toc162647598)

[**Figure 17** Relationship Between PRALBUM and DISCANCR and Their Impact on Prediction 24](#_Toc162647599)

[**Figure 18** Relationship Between PRSGOT and STEROID and Their Impact on Prediction 24](#_Toc162647600)

[**Figure 19** Relationship Between DYSPNEA and STEROID and Their Impact on Prediction 24](#_Toc162647601)

[**Figure 20** Relationship Between HXCOPD and TRANSFUS and Their Impact on Prediction 25](#_Toc162647602)

[**Figure 21** Relationship Between RACE\_NEW and PRHCT and Their Impact on Prediction 25](#_Toc162647603)

[**Figure 22** Relationship Between PRBUN and STEROID and Their Impact on Prediction 25](#_Toc162647604)

[**Figure 23** Relationship Between PRSODM and DISCANCR and Their Impact on Prediction 26](#_Toc162647605)

# Introduction

Coronary Artery Bypass Graft (CABG) is a common cardiac surgery but continues to have many associated risks, including major bleeding which might need blood transfusion. Previous research has shown that blood transfusion during CABG surgery is associated with an increased risk for mortality after surgery. Specially, post-operative blood transfusion after CABG is associated with higher odds of readmission and heart failure within 30-days.

To lower the risk of mortality after surgery, there is a need to develop models that preoperatively predict which patients will need an intra-operative or post-operative blood transfusion. This will not only help to improve patient selection and patient education, but also physician preoperative awareness and perioperative guidelines for CABG patients. Therefore, the goal of this project is to explore different approaches and find the models that can best make predictions, including feature selection/engineering, classical statistical models, and neural networks.

# Problem Statement & Project Objectives

The objectives of the project are three-fold. The first objective is to develop models that can best predict whether a CABG patient will need blood transfusions. Second, we also look to experiment with various data science techniques to be applied in our models in order to achieve best performance, including feature selection, feature engineering, and synthetic data generation. Lastly, we aim to build a full set of modules and functions to be reused in the future beyond the current project. The modularized codes include but not limited to data preprocessing, feature selection, feature engineering, and modeling.

# Related Work

Research have been conducted to investigate factors that can help to predict major bleeding (Gao, et al., 2022) and the need for red blood cell transfusion after cardiac surgery (Li, et al., 2024). In one of the studies (Tschoellitsch, Bock, Mahecic, Hofmann, & Meier, 2022) that is most relevant to the current project, the researchers employed machine learning models to predict perioperative allogeneic blood transfusion for cardiac patients. The best model (Random Forest) showed good performance (RUC ranged from .76 - .86), however, the study has several limitations. For example, the data was from a single adult cardiac surgery center in Austria with a relatively small sample size (N = 3782), thus the results may not be generalizable to other samples with different demographics or nationalities. Moreover, the studies only predicted allogeneic blood transfusion (i.e., transfusion of more than 10 units of packed red blood cells (pRBC)), while blood transfusion regardless of volume has been associated with many known risks. Lastly, the study only tested the basic machine learning models (e.g., tree-based models), and it is likely that the performance can be significantly improved using more advanced techniques and deep neutral networks.

To address this research gap, the current project will use the national medical database in the U.S. with a large sample size of over 8,000 data points. Additionally, we will predict blood transfusion regardless of volume. Lastly, we will experiment with various approaches in order to optimize the performance, including feature selection, feature engineering, synthetic data generation, and deep neural networks.

# Solution and Methodology

*Data Source and Data Preprocessing*

The data was downloaded from the Participant Use Data File (PUF) on the American College of Surgeons National Surgical Quality Improvement Program ([ACS NSQIP](https://www.facs.org/quality-programs/data-and-registries/acs-nsqip/)). In this project, we focus on the data from 2018 to 2022, which has a total of 8,587 observations and around 294 variables across five datasets.

Add a table to summarize our data?

Add a table to display our feature dictionary?

After data preprocessing, including basic cleanup, imputation (mean for numeric variables and most frequent values for categorical variables), standardization, and encoding, the dataset with 41 features identified as most relevant to the current study was served as our baseline data. The target variable is Occurrences Bleeding Transfusions, which is a binary variable predicting whether the patient needs blood transfusion after surgery. The target can be further categorized into intraoperative vs. postoperative vs. no transfusion, therefore can be transformed into a 3-class variable when needed. With different analysis strategies, these features will be entered into our models to predict the target variable, and we will compare the performance with each other as well as with the benchmarks from previous research.

Add a table to display the 41 selected features

*Add a section on EDA?*

*Analysis Strategy*

***Figure 1*** shows the analysis strategy for the current project. After data preprocessing, we first entered the data into eight models, and used the results as our baseline benchmark. Next, we experimented with a different technique and method, and then entered the modified data into our models. In each iteration, we compared the new model performance with the baseline results and may start again from any of previous steps. For example, new data will be added to compare the prediction results when they become available. Or new feature engineering methods will be applied so we run the same data again starting from the third column (“methods).

**Figure 1** Research Strategy of the Project

A diagram of a process

Description automatically generated

# Results and Discussion

## Model selection and tuning

Ten typical and representative supervised learning classification models are selected for this project and will be run through at each iteration every time there is an adjustment or improvement in the method, or new or extended data become available. The selected models are Decision Tree (criterion = “gini” and “entropy”), SVM (kernel = “linear” and “rbf”), Gaussian Naive Bayes, Logistic Regression, Gradient Boosting, XGBoost, KNN, and Random Forest (top important features = 20).

The target variable (the predicted variable, dependent variable) is “OTHBLEED” in the original dataset, or Occurrences Bleeding Transfusions. We group the values “Transfusions/Intraop/Postop” together and map as “1”, and “No Complication” was converted to “0”. We set random state as 100, testing size 25%, k-folds 10, and hold them constant in all models for comparison. The parameters of model construction and prediction results and evaluations for each run are documented in this section below.

*Iteration #1*

As the first step, we include all possible features with a missing data percentage larger than 50%. We then drop “NOTHBLEED” and “DOTHBLEED” due to high collinearity with the target. NOTHBLEED, number of bleeding transfusions occurrences, is highly correlated with the target OTHBLEED (occurrences bleeding transfusions) and has a Pearson correlation coefficient of -0.99. Similarly, DOTHBLEED, days from operation until bleeding transfusions complication, has a -0.81 Pearson correlation coefficient with the target. This leaves the data to be a 4953 by 127 dimension.

Simple linear imputation is applied to fill in the missing data in order for some models to run without errors. However, data is not standardized in this first round of modeling. ***Table 1*** summarizes the setup of iteration #1.

Eight typical classification models were selected to train the data, including Decision Tree, SVM, Gaussian Naive Bayes, Logistic Regression, Gradient Boosting, XGBoost, KNN, and Random Forest. Different parameters and algorithm were also compared within Decision Tree (gini vs entropy) and SVM (linear vs rbf). ***Table 2*** indicates that Random Forest and Gradient Boosting have the better results across several evaluation metrics (accuracy score, root mean square error, F1 score, and ROC-AUC score). ***Figure 2*** combines the ROC plots for all the models which verify the conclusion above on top performing models.

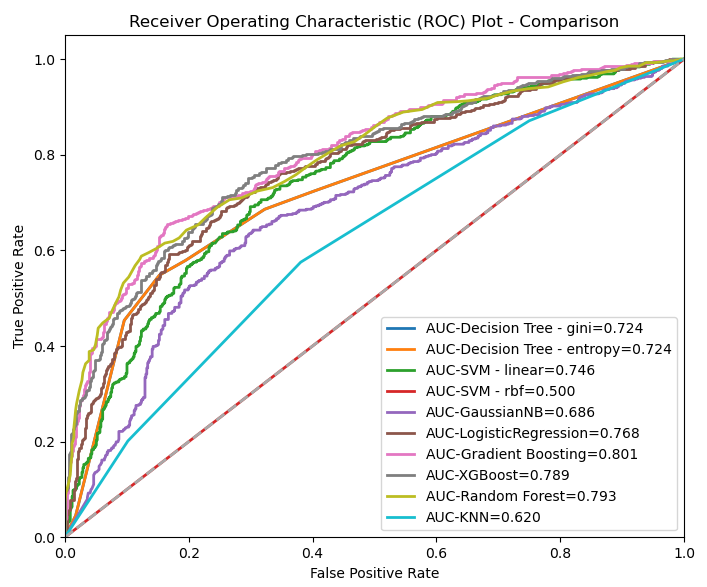
**Table 1** Summary of Iteration #1 Setup

|  |  |
| --- | --- |
| Data year | 2018-2020 |
| Observations | 4953 |
| Features included | 126 |
| Features manually dropped based on expert judgement | NOTHBLEED  DOTHBLEED |
| Data preprocessing methods applied | Simple imputations |
| Final dataset | [CABG\_2018\_2020\_baseline.csv](https://github.com/jennytsai32/Capstone/blob/master/code/main_code/processed_data/2018_2020/CABG_2018_2020_baseline.csv) |

**Table 2** Model Results from Iteration #1

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Model Name | Parameters | Accuracy | RMSE | F1 (macro avg) | ROC-AUC |
| Decision Tree – gini | max\_depth=3  min\_samples\_leaf=5 | 68.12 | 0.56 | 0.68 | 0.72 |
| Decision Tree – entropy | max\_depth=3  min\_samples\_leaf=5 | 68.12 | 0.56 | 0.68 | 0.72 |
| SVM – linear | C=1.0, gamma=auto | 69.49 | 0.55 | 0.69 | 0.75 |
| SVM – rbf | C=1.0, gamma=0.2 | 55.53 | 0.67 | 0.36 | 0.50 |
| Gaussian Naive Bayes |  | 52.95 | 0.69 | 0.50 | 0.69 |
| Logistic Regression |  | 71.75 | 0.53 | 0.71 | 0.77 |
| Gradient Boosting | n\_estimators=300  learning\_rate=0.05 | 73.93 | 0.51 | 0.73 | 0.80 |
| XGBoost | n\_estimators=100  eta=0.3 | 72.56 | 0.52 | 0.72 | 0.79 |
| KNN | n\_neighbor=3 | 59.97 | 0.63 | 0.60 | 0.62 |
| Random Forest | n\_estimators=300  feature\_importances=20 | 73.93 | 0.51 | 0.73 | 0.79 |

**Figure 2** ROC Plot with 10 Selected Models from Iteration #1



*Iteration #2*

Based on the first round of modeling with little data manipulation and interventions, we take it a step further to simply include more data from year 2021-2022 (***Table 3***). Results indicate very slight improvements in almost all models, with Gradient Boosting still being the best in all evaluation metrics followed by Random Forest not too far behind (***Table 4*, *Figure 3***).

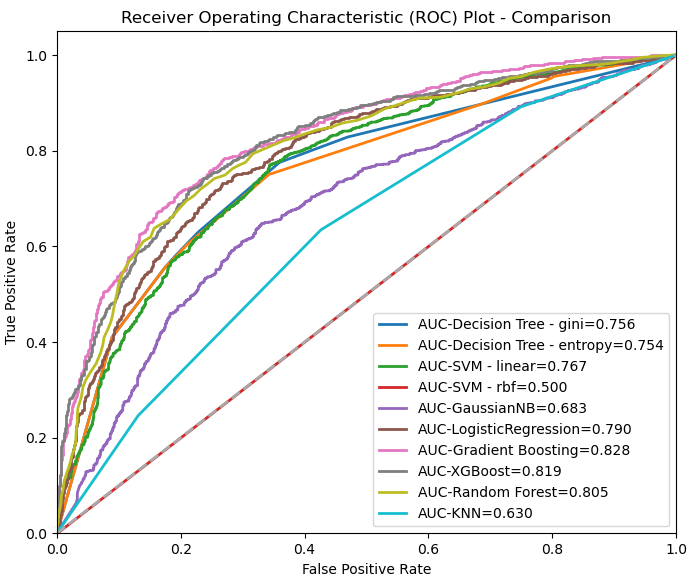
**Table 3** Summary of Iteration #2 Setup

|  |  |
| --- | --- |
| Data year | 2018-2022 |
| Observations | 8587 |
| Features included | 126 |
| Final dataset | [CABG\_2018\_2022\_baseline.csv](https://github.com/jennytsai32/Capstone/blob/master/code/main_code/processed_data/2018_2022/CABG_2018_2022_baseline.csv) |

**Table 4** Model Results from Iteration #2

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Model Name | Parameters | Accuracy | RMSE | F1 (macro avg) | ROC-AUC |
| Decision Tree – gini | max\_depth=3  min\_samples\_leaf=5 | 70.70 | 0.54 | 0.71 | 0.76 |
| Decision Tree – entropy | max\_depth=3  min\_samples\_leaf=5 | 70.33 | 0.54 | 0.70 | 0.75 |
| SVM – linear | C=1.0, gamma=auto | 70.28 | 0.55 | 0.70 | 0.77 |
| SVM – rbf | C=1.0, gamma=0.2 | 50.77 | 0.70 | 0.34 | 0.50 |
| Gaussian Naive Bayes |  | 57.29 | 0.65 | 0.53 | 0.68 |
| Logistic Regression |  | 72.99 | 0.52 | 0.73 | 0.79 |
| Gradient Boosting | n\_estimators=300  learning\_rate=0.05 | 75.41 | 0.50 | 0.75 | 0.83 |
| XGBoost | n\_estimators=100  eta=0.3 | 75.13 | 0.50 | 0.75 | 0.82 |
| KNN | n\_neighbor=3 | 60.36 | 0.63 | 0.60 | 0.63 |
| Random Forest | n\_estimators=300  feature\_importances=20 | 74.29 | 0.51 | 0.74 | 0.81 |

**Figure 3** ROC Plot with 10 Selected Models from Iteration #2



*Iteration #3:*

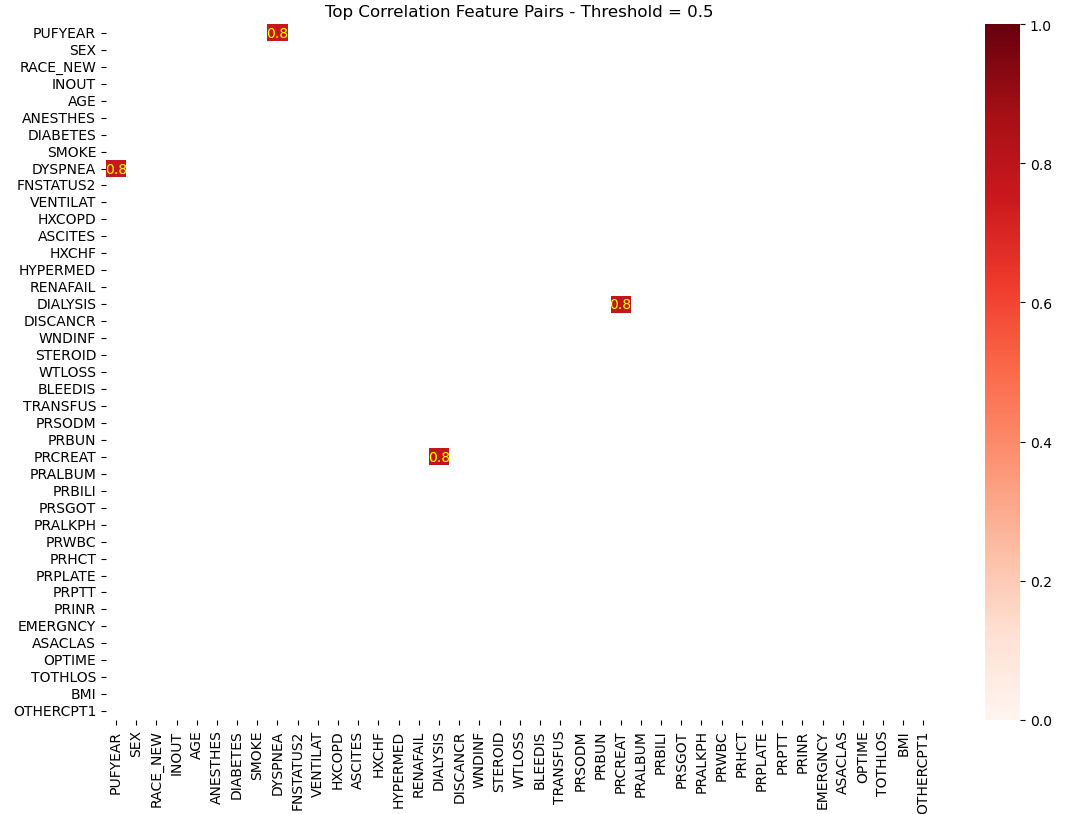
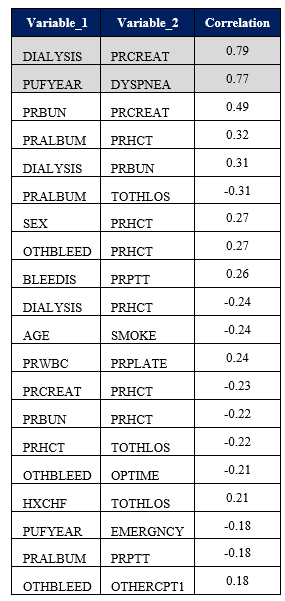
Next, we focus on cleaning up the data with more advanced data pre-processing methods including imputation, encoding, standardization to improve trainings of the model. Cross validation method is also introduced with a 10 fold parameter to enhance the mean accuracy. Further, 41 out of the 126 features from previous steps were manually screened and selected. Those obviously irrelevant features or those have little relationship with the target variable were removed based on common sense in medical and data science.

Correlation is checked for all features to identify highly correlated features which may affect model performance. “DLALYSIS” and “PRCREAT”, and “PUFYEAR” and “DYSPNEA” have the highest Pearson correlation coefficient and larger than 0.5 (***Figure 4***, left panel is a heatmap and right panel is the table of paired features).

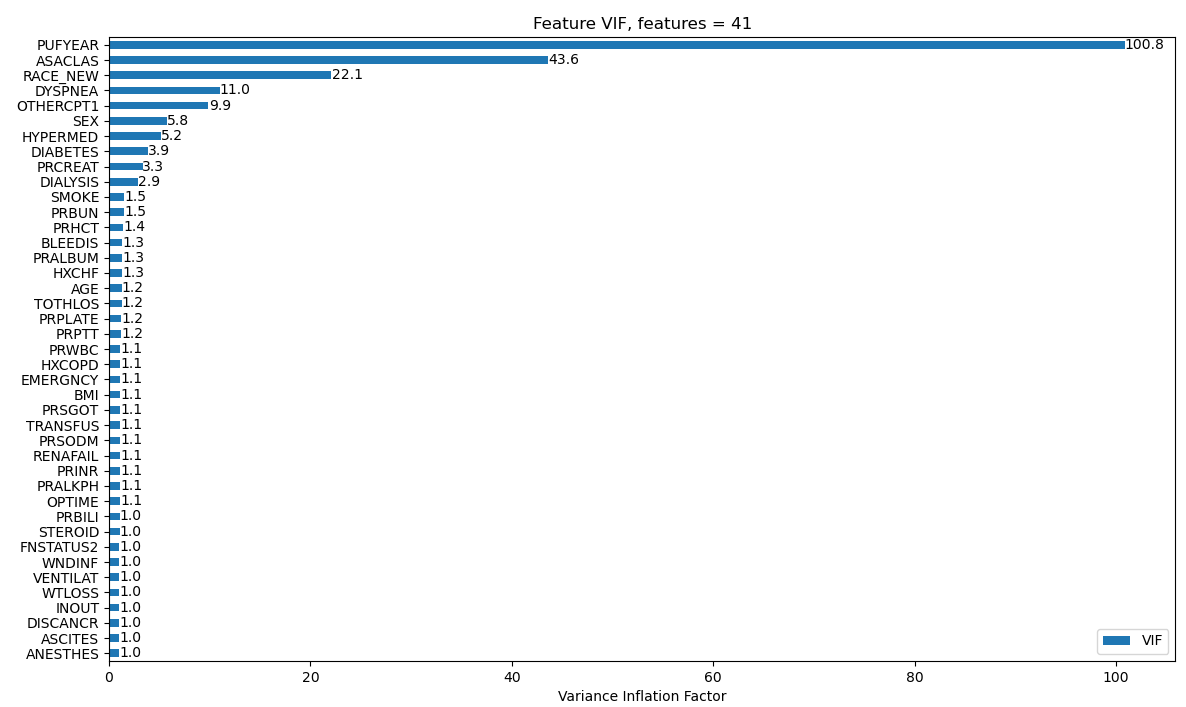
“HEIGHT”, “WEIGHT” are dropped due to multicollinearity issue with “BMI” (kept). Similarly, “ETHNICITY\_HISPANIC” was dropped which is highly correlated with and a subset of feature “RACE\_NEW” (kept). There are also features that may cause multicollinearity after the variance inflation factor (VIF) check but we decide to keep given their importance as a measure in understanding impacts on transfusion needs and decisions. As shown in ***Figure 5*** for example, “ASACLAS” (ASA classification, VIF = 43.6), “RACE\_NEW” (VIF = 22.1), “OTHERCPT1” (Other Procedure, VIF = 9.9), and (SEX (VIF = 5.8). The data year, “PUFYEAR” has the highest VIF with a value of 100.8, we run all ten models with and without “PUFYEAR” to double check if recent or older data has a relationship with blood transfusion. The results are close to being identical between the two runs. Therefore, it is decided to drop “PUFYEAR” since it won’t matter much and given its high VIF value.

***Table 6*** summarizes the setup for model construction iteration 3, ***Table 6*** and ***Figure 6*** summarizes the results. Same as previous, Random Forest performs the best followed closely by Gradient Boosting. These 40 selected features now composite the baseline of our model iterations, before applying more advanced data processing and modeling methods in the following steps.

**Figure 4** Top Correlation Feature Pairs in Iteration #3

**Figure 5** Variance Inflation Factor (VIF) Values of All Features (n = 41) in Iteration #3



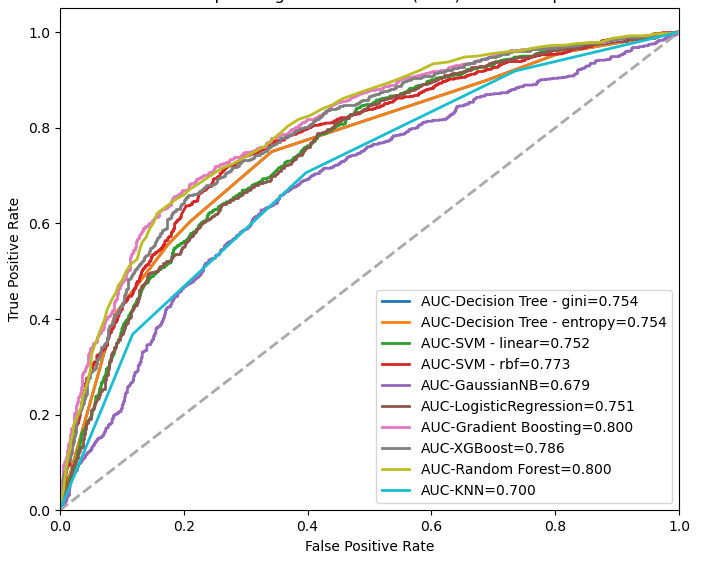
**Table 5** Summary of Iteration #3 Setup

|  |  |
| --- | --- |
| Data year | 2018-2022 |
| Observations | 8587 |
| Features included | 40 |
| Features manually dropped based on expert judgement | HEIGHT  WEIGHT  ETHNICITY\_HISPANIC  PUFYEAR |
| Features kept based on expert judgement | ASACLAS  RACE\_NEW  SEX  OTHERCPT1 |
| Data preprocessing methods applied | Standardization  Encoding  Cross validation (10-folds) |
| Final dataset | [CABG\_5yr\_preselect41.csv](https://github.com/jennytsai32/Capstone/blob/master/code/main_code/processed_data/2018_2022/CABG_5yr_preselect41.csv) |

**Table 6** Model Results from Iteration #3

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Model Name | Parameters | Mean Accuracy  (10 folds) | RMSE | F1  (macro avg) | ROC-AUC |
| Decision Tree – gini | max\_depth=3  min\_samples\_leaf=5 | 69.70 | 0.54 | 0.70 | 0.75 |
| Decision Tree – entropy | max\_depth=3  min\_samples\_leaf=5 | 69.65 | 0.54 | 0.70 | 0.75 |
| SVM – linear | C=1.0, gamma=auto | 68.31 | 0.57 | 0.68 | 0.75 |
| SVM – rbf | C=1.0, gamma=0.2 | 71.52 | 0.54 | 0.71 | 0.77 |
| Gaussian Naive Bayes |  | 54.23 | 0.68 | 0.48 | 0.68 |
| Logistic Regression |  | 68.28 | 0.57 | 0.68 | 0.75 |
| Gradient Boosting | n\_estimators=300  learning\_rate=0.05 | 73.49 | 0.52 | 0.73 | 0.80 |
| XGBoost | n\_estimators=100  eta=0.3 | 72.62 | 0.53 | 0.72 | 0.79 |
| KNN | n\_neighbor=3 | 66.32 | 0.59 | 0.65 | 0.70 |
| Random Forest | n\_estimators=300  feature\_importances=20 | 73.98 | 0.52 | 0.73 | 0.80 |

**Figure 6** ROC Plot with 10 Selected Models from Iteration #3



*Iteration #4:*

Principal Component Analysis (PCA) is a dimensionality reduction technique used to transform high-dimensional data into a lower-dimensional representation, preserving the most important information. It is commonly used to tackle multicollinearity and improves dimension. To have a complete comparison with all popular feature selection methods, we conduct PCA and the new dataset after PCA transformation reduced the feature dimension by one.

***Table 7*** summarizes iteration #4 setup and ***Table 8*** and ***Figure 7*** includes the results comparison. Random Forest is the best performing model in this iteration (, however its results are significantly below iteration #3 and the same applies to all other models. We will then stop using PCA in future model constructions.

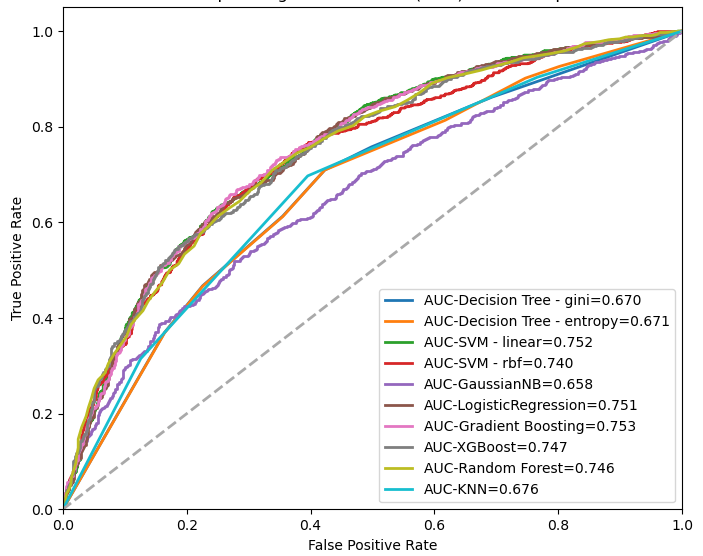
**Table 7** Summary of Iteration #4 Setup

|  |  |
| --- | --- |
| Data year | 2018-2022 |
| Observations | 8587 |
| Features included | 39 |
| Data preprocessing methods applied | PCA |
| Final dataset | [CABG\_5yr\_PCA\_39feature.csv](https://github.com/jennytsai32/Capstone/blob/master/code/main_code/processed_data/2018_2022/CABG_5yr_PCA_39feature.csv) |

**Table 8** Model Results from Iteration #4

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Model Name | Parameters | Mean Accuracy (10 folds) | RMSE | F1  (macro avg) | ROC-AUC |
| Decision Tree - gini | max\_depth=3, min\_samples\_leaf=5 | 63.03 | 0.60 | 0.64 | 0.67 |
| Decision Tree - entropy | max\_depth=3, min\_samples\_leaf=5 | 62.89 | 0.60 | 0.64 | 0.67 |
| SVM - linear | C=1.0, gamma=auto | 68.31 | 0.57 | 0.68 | 0.75 |
| SVM - rbf | C=1.0, gamma=0.2 | 68.04 | 0.57 | 0.67 | 0.74 |
| GaussianNB |  | 56.46 | 0.66 | 0.53 | 0.66 |
| LogisticRegression |  | 68.27 | 0.57 | 0.68 | 0.75 |
| Gradient Boosting | n\_estimators=300, learning\_rate=0.05 | 68.90 | 0.56 | 0.69 | 0.75 |
| XGBoost | n\_estimators=100, eta=0.3 | 68.78 | 0.57 | 0.68 | 0.75 |
| KNN | n\_neighbors=3 | 63.71 | 0.59 | 0.657 | 0.68 |
| Random Forest | n\_estimators=100, features\_importances=20 | 69.10 | 0.57 | 0.68 | 0.75 |

**Figure 7** ROC Plot with 10 Selected Models from Iteration #4

**

*Iteration #5:*

The key feature for this experiment is to use AutoFeat package to transform the original features. It automates feature engineering and selection and fit a linear prediction model. In this iteration, top 20 important features derived from previous iterations were selected to use with AutoFeat library. ***Table 9*** summaries the key information and ***Table 10*** and ***Figure 8*** displays the results. Gradient Boosting is the best performing model but it doesn’t beat the top models from previous iterations.

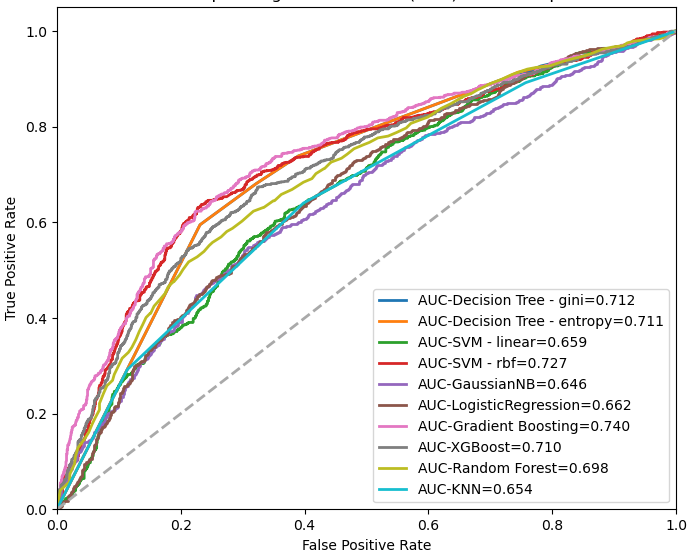
**Table 9** Summary of Iteration #5 Setup

|  |  |
| --- | --- |
| Data year | 2018-2022 |
| Observations | 8587 |
| Features included | 20 |
| Data preprocessing methods applied | AutoFeat |
| Final dataset | [CABG\_autofeat\_top20.csv](https://github.com/jennytsai32/Capstone/blob/master/code/main_code/processed_data/2018_2022/CABG_autofeat_top20.csv) |

**Table 10** Model Results from Iteration #5

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Model Name | Parameters | Mean Accuracy (10 folds) | RMSE | F1-score (macro avg) | ROC-AUC |
| Decision Tree - gini | max\_depth=3, min\_samples\_leaf=5 | 67.45 | 0.56 | 0.68 | 0.71 |
| Decision Tree - entropy | max\_depth=3, min\_samples\_leaf=5 | 67.47 | 0.56 | 0.68 | 0.71 |
| SVM - linear | C=1.0, gamma=auto | 60.57 | 0.63 | 0.60 | 0.66 |
| SVM - rbf | C=1.0, gamma=0.2 | 68.99 | 0.55 | 0.69 | 0.73 |
| GaussianNB |  | 52.55 | 0.69 | 0.44 | 0.65 |
| LogisticRegression |  | 61.50 | 0.62 | 0.62 | 0.66 |
| Gradient Boosting | n\_estimators=300, learning\_rate=0.05 | 68.88 | 0.55 | 0.69 | 0.74 |
| XGBoost | n\_estimators=100, eta=0.3 | 67.07 | 0.57 | 0.67 | 0.71 |
| KNN | n\_neighbors=3 | 62.00 | 0.62 | 0.62 | 0.65 |
| Random Forest | n\_estimators=100, features\_importances=20 | 64.61 | 0.59 | 0.65 | 0.70 |

**Figure 8** ROC Plot with 10 Selected Models from Iteration #5



*Iteration #6:*

TPOT is an automated machine learning tool that optimizes machine learning pipelines using genetic programming. TPOT automatically explores thousands of possible pipelines to find the best results data. This experiment tests the TPOT method and concludes that best model is ExtraTrees (***Table 11***). However, it is slightly under performed by the best models from iteration #3 – Random Forest and Gradient Boosting.

**Table 11** Summary of Iteration #6 Setup and results

|  |  |
| --- | --- |
| Data year | 2018-2022 |
| Observations | 8587 |
| Features included | 40 |
| Data preprocessing methods applied | TPOT |
| Final dataset | [CABG\_5yr\_preselect41.csv](https://github.com/jennytsai32/Capstone/blob/master/code/main_code/processed_data/2018_2022/CABG_5yr_preselect41.csv) |
| Model parameters | n\_estimators = 100  generations = 5  population\_size = 20  verbosity = 2 |
| Results | Best pipeline: ExtraTreesClassifier with the following model parameters (booststrap=True, criterion=entropy, max\_features=1.0, min\_samples\_leaf=1, min\_samples\_split=9, n\_estimators=100)  Accuracy = 72.52 |

*Iteration #7:*

Synthetic data generation is another effective feature engineering method and is used in this iteration. Again, 40 features that lead to the best models so far in iteration #3 are all included in this experiment. DataSynthesizer library that is based on Bayesian networks algorithm is used to re-generate a new dataset with the size of 1000 x 40 that feeds into our 10 models (***Table 12***).

This time, all models produce much higher results and the best is still between Gradient Boosting and Random Forest. Gradient Boosting model with synthetic data generation method significantly brings the mean accuracy to above 90%, with an error of 0.31, an F1 score of 0.8, and an ROC-AUC score of 0.93 (***Table 14*** and ***Figure 9***). This is our best results so far.

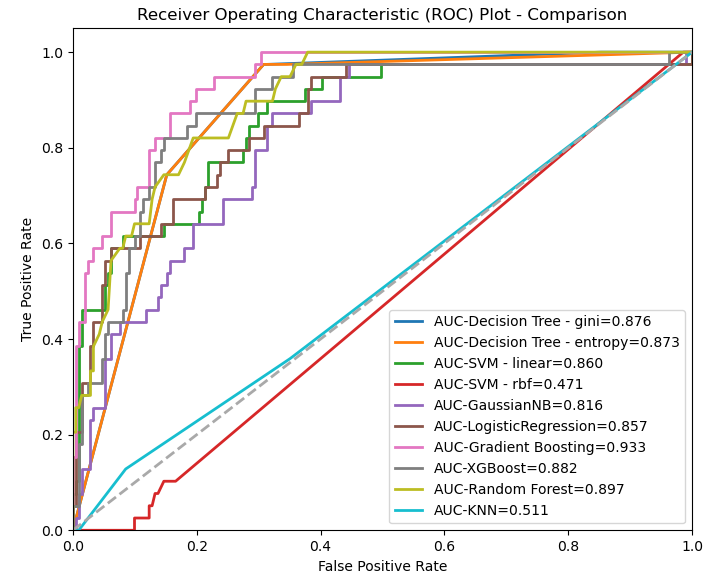
**Table 12** Summary of Iteration #7 Setup

|  |  |
| --- | --- |
| Data year | 2018-2022 |
| Observations | 1000 |
| Features included | 40 |
| Data preprocessing methods applied | Synthetic data generation – Bayesian networks |
| Final dataset | [CABG\_synthetic\_Bayesian.csv](https://github.com/jennytsai32/Capstone/blob/master/code/main_code/processed_data/2018_2022/CABG_synthetic_Bayesian.csv) |

**Table 13** Summary of Iteration #7 Setup and results

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Model Name | Parameters | Mean Accuracy  (10 folds) | RMSE | F1  (macro avg) | ROC-AUC |
| Decision Tree - gini | max\_depth=3, min\_samples\_leaf=5 | 86.90 | 0.41 | 0.74 | 0.88 |
| Decision Tree - entropy | max\_depth=3, min\_samples\_leaf=5 | 86.90 | 0.41 | 0.74 | 0.87 |
| SVM - linear | C=1.0, gamma=auto | 87.60 | 0.36 | 0.76 | 0.86 |
| SVM - rbf | C=1.0, gamma=0.2 | 83.90 | 0.39 | 0.46 | 0.47 |
| GaussianNB |  | 79.60 | 0.53 | 0.62 | 0.82 |
| LogisticRegression |  | 86.60 | 0.34 | 0.77 | 0.86 |
| Gradient Boosting | n\_estimators=300, learning\_rate=0.05 | 90.80 | 0.31 | 0.80 | 0.93 |
| XGBoost | n\_estimators=100, eta=0.3 | 88.80 | 0.38 | 0.72 | 0.88 |
| KNN | n\_neighbors=3 | 81.20 | 0.46 | 0.52 | 0.51 |
| Random Forest | n\_estimators=100, features\_importances=20 | 86.60 | 0.35 | 0.71 | 0.90 |

**Figure 9** ROC Plot with 10 Selected Models from Iteration #7



*Iteration #8:*

Lastly, we include even more data from 2015-2017 and test if older data will even improve the current results even more. This brings the total observations to 13,534 from over 8,000 (***Table 14***), but it does not significantly improve the modeling results – the best model is still Gradient Boosting followed closely by Random Forest (***Table 15*** and ***Figure 10***).

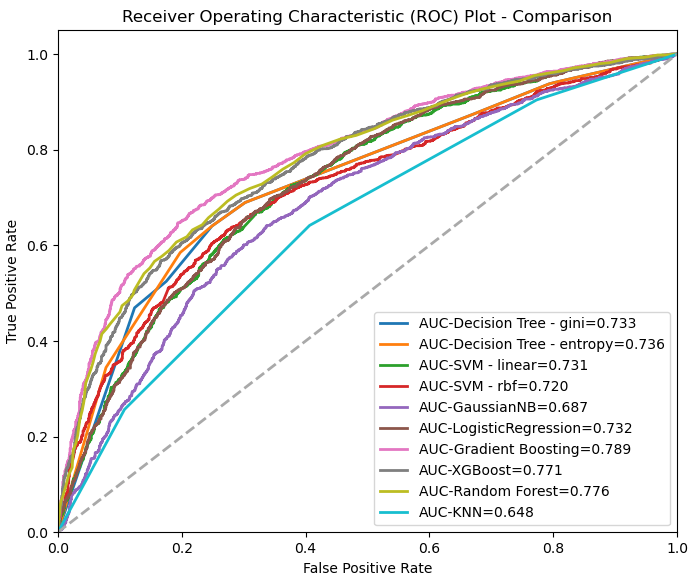
**Table 14** Summary of Iteration #8 Setup

|  |  |
| --- | --- |
| Data year | 2015-2022 |
| Observations | 13534 |
| Features included | 40 |
| Final dataset | [CABG\_8yr\_preselect41.csv](https://github.com/jennytsai32/Capstone/blob/master/code/main_code/processed_data/2015_2022/CABG_8yr_preselect41.csv) |

**Table 15** Summary of Iteration #8 Setup and results

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Model Name | Parameters | Mean Accuracy (10 folds) | RMSE | F1  (macro avg) | ROC-AUC |
| Decision Tree - gini | max\_depth=3, min\_samples\_leaf=5 | 69.27 | 0.55 | 0.69 | 0.73 |
| Decision Tree - entropy | max\_depth=3, min\_samples\_leaf=5 | 69.28 | 0.55 | 0.69 | 0.74 |
| SVM - linear | C=1.0, gamma=auto | 66.91 | 0.57 | 0.67 | 0.73 |
| SVM - rbf | C=1.0, gamma=0.2 | 66.43 | 0.58 | 0.64 | 0.72 |
| GaussianNB |  | 54.99 | 0.66 | 0.53 | 0.69 |
| LogisticRegression |  | 66.90 | 0.57 | 0.68 | 0.73 |
| Gradient Boosting | n\_estimators=300, learning\_rate=0.05 | 72.33 | 0.52 | 0.73 | 0.79 |
| XGBoost | n\_estimators=100, eta=0.3 | 70.97 | 0.54 | 0.70 | 0.77 |
| KNN | n\_neighbors=3 | 61.00 | 0.62 | 0.62 | 0.65 |
| Random Forest | n\_estimators=100, features\_importances=20 | 72.23 | 0.54 | 0.70 | 0.78 |

**Figure 10** ROC Plot with 10 Selected Models from Iteration #8



## Results and interpretation

By all data science and machine learning methods in our experiments, Gradient Boosting, Random Forest, and XGBoost are consistently leading the performance metrics in all iterations with Gradient Boosting being slightly better. The best model construction is from iteration #3 with data covered from 2018-2022, proper data cleaning methods applied, and 40 features included. In this run, Gradient Boosting performs the best. While it has a slightly lower average accuracy of 73.49 compared with 73.98 from Random Forest, it’s better in all other important model evaluation metrics in RMSE, F1-score, and the ROC-AUC score (***Table 16***), which are typically weighted more heavily than accuracy in modeling evaluation.

**Table 16** Best Performing Models – Gradient Boosting vs Random Forest vs XGBoost from Iteration #3

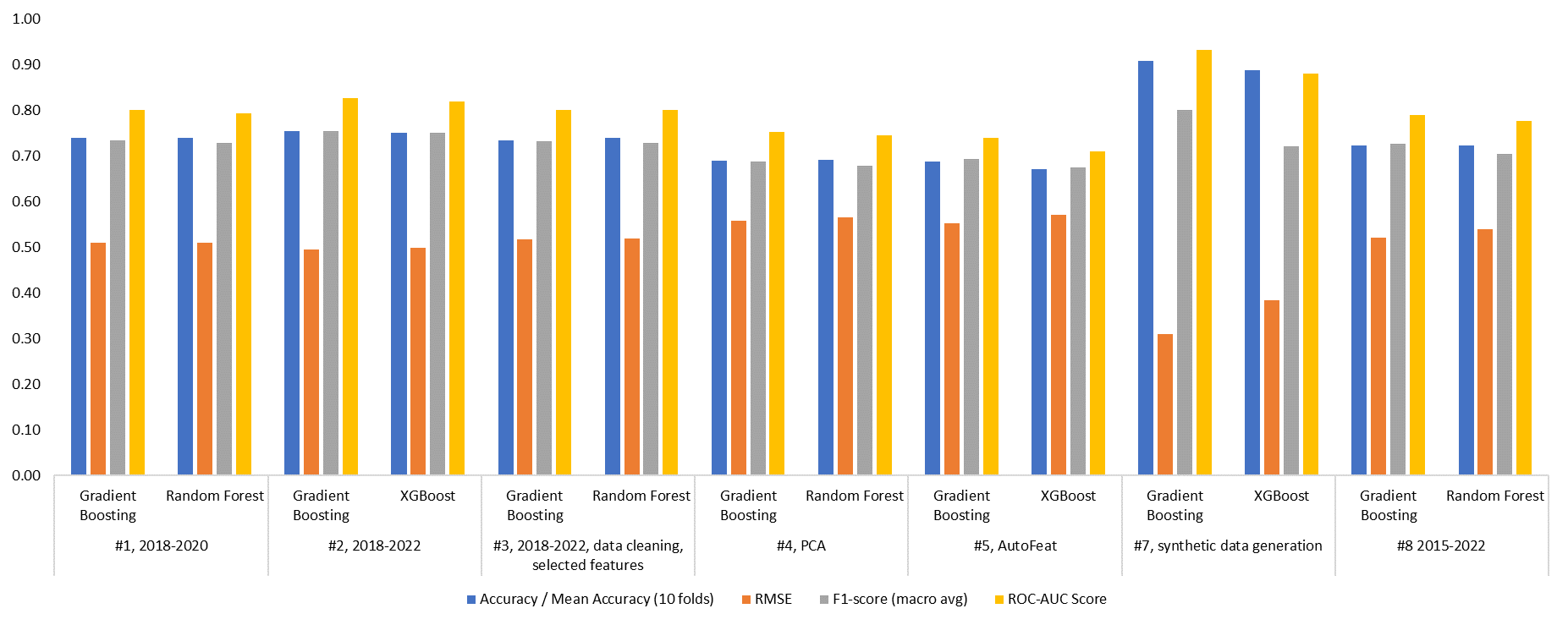
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Model Name | Mean Accuracy (10 folds) | RMSE | F1-score  (macro avg) | ROC-AUC score |
| Gradient Boosting | 73.49469 | 0.51751 | 0.73204 | 0.80018 |
| Random Forest | 73.98384 | 0.52020 | 0.72878 | 0.80013 |
| XGBoost | 72.62148 | 0.53128 | 0.71766 | 0.78577 |

Focusing on just Gradient Boosting and other top 2 models across all iterations, model accuracy ranges from 67.07 - 90.80, root-mean-square error (RMSE) 0.31 – 0.57, F1-score 0.67 – 0.80, and ROC-AUC score has an upper bound of 0.93 and a lower bound of 0.71 (***Table 17***and***Figure 11***). Our results are consistent with our literature review in which others previous work has a range of ROC-AUC from 0.76 – 0.86. The synthetic data generation method enables significant improvements and bring the ROC-AUC scores in our results to 0.93 as the highest (Gradient Boosting from iteration #3).

**Table 17** Top 2 Models Comparison in All Iterations

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Iteration** | **Top 2 Models** | **Accuracy / Mean Accuracy (10 folds)** | **RMSE** | **F1-score (macro avg)** | **ROC-AUC Score** |
| #1. 2018-2020 | Gradient Boosting | 73.93059 | 0.51058 | 0.73430 | 0.80146 |
| Random Forest | 73.93059 | 0.51058 | 0.72932 | 0.79298 |
| #2. 2018-2022 | Gradient Boosting | 75.40755 | 0.49591 | 0.75391 | 0.82770 |
| XGBoost | 75.12809 | 0.49872 | 0.75123 | 0.81872 |
| #3. 2018-2022  data cleaning  selected features | Gradient Boosting | 73.49469 | 0.51751 | 0.73204 | 0.80018 |
| Random Forest | 73.98384 | 0.52020 | 0.72878 | 0.80013 |
| #4. PCA | Gradient Boosting | 68.90664 | 0.55821 | 0.68838 | 0.75256 |
| Random Forest | 69.10431 | 0.56567 | 0.67935 | 0.74613 |
| #5. AutoFeat | Gradient Boosting | 68.88298 | 0.55276 | 0.69395 | 0.73976 |
| XGBoost | 67.06653 | 0.57059 | 0.67438 | 0.71022 |
| #7. synthetic data generation | Gradient Boosting | 90.80000 | 0.30984 | 0.80066 | 0.93341 |
| XGBoost | 88.80000 | 0.38471 | 0.72188 | 0.88152 |
| #8. 2015-2022 | Gradient Boosting | 72.32897 | 0.52056 | 0.72637 | 0.78931 |
| Random Forest | 72.22554 | 0.53924 | 0.70484 | 0.77600 |

**Figure 11** Top 2 Models Comparison in All Iterations



# Discussion - understanding what features have higher impacts on model prediction

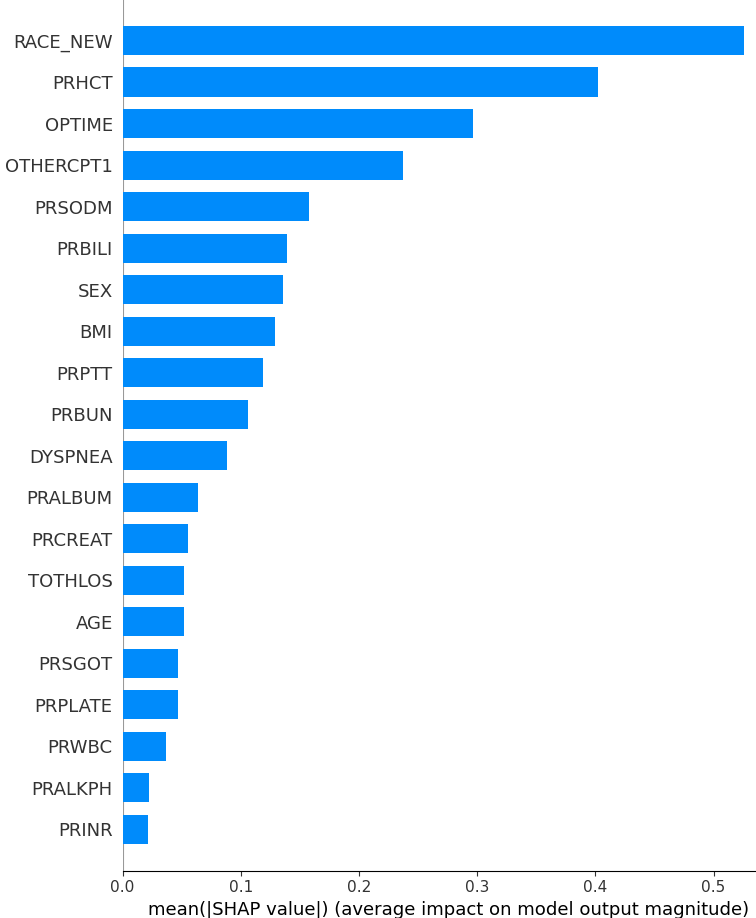
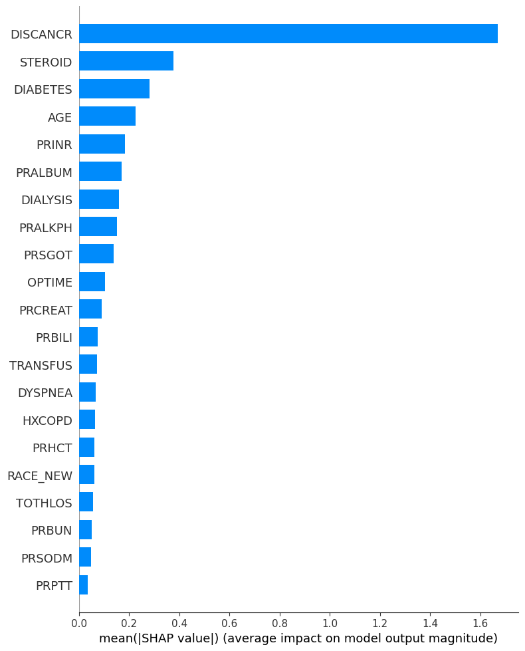
***Figure 12*** lists the ranking of the most importance 20 features from the best performing model – Gradient Boosting. Panel b on the right is what will be focused on in the following discussions since it has best results, but we do note that the feature data is transformed after synthetic generation. Therefore, we do not want to completely rule out any interesting observations on feature contributions to model prediction from iteration #3, which is the second best and the feature data are closer to their original values. We decide to document the feature importance and impacts analysis from panel a from iteration #3 in the Appendix as a background reference.

Back to panel b in ***Figure 12***, by ranking, DISCANCER[[1]](#footnote-1), STEROID[[2]](#footnote-2), DIABETES[[3]](#footnote-3), AGE, and PRALBUM[[4]](#footnote-4) are the top five important features that have made significant contributions to our Gradient Boosting model prediction from iteration #7 using the synthetic data generation method. Among which, DISCANCER has a significantly higher impact on predictions than other features.

**Figure 12** Feature Importance TOP 20 from Gradient Boosting in Iteration #3 vs Iteration #7

b. Feature importance ranking in Iteration #7 (synthetic data generation)

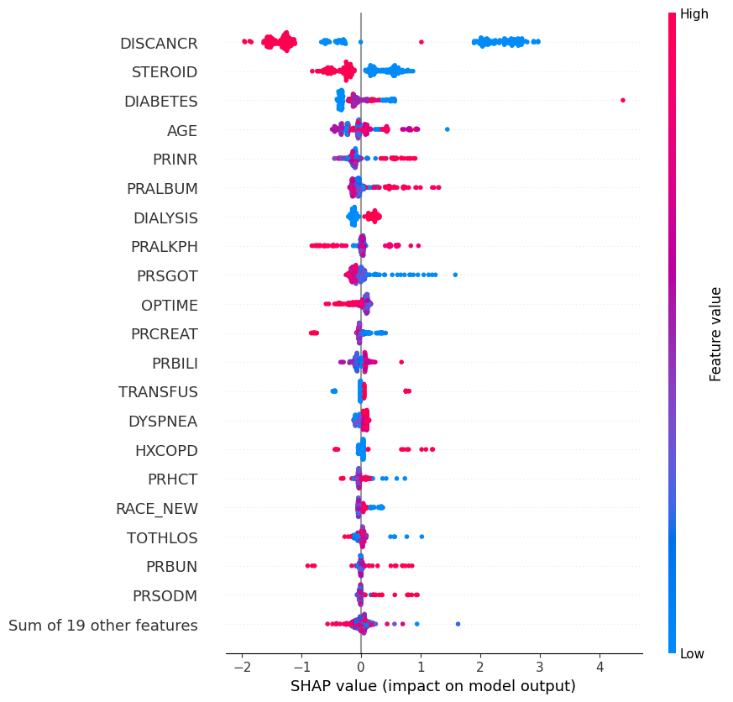
a. Feature importance ranking in Iteration #3

Beeswarm plots can be used to highlight these important feature relationships. It indicates the relationship between feature values (high is red, blue is low) and its contribution to prediction classes (below zero or on the left falls into zero class or no transfusion, above zero or on the right belongs to one or need transfusion), as shown in***Figure 13***. For example, the top 1 feature that impact the predicting results the most – DISCANCR. Those high density red dots on the left meaning many high values of DISCANCER (which is 1 since it’s a categorical data) are contributing to high probability of predicting a zero class, or no transfusion. A negative feature impact on prediction class is observed.

Similarly, STEROID, PRALKPH, OPTIME all show strong relationship of a native impact on prediction – the higher their values, the more they contribute to high probability of the zero class, the more likely no transfusion is needed. On the contrary, PRSGOT, PRCREAT, PRHCT, and RACE\_NEW indicate a strong positive impact on prediction, that is the higher values of these features, the more likely transfusion is needed. ***Table 18*** explained the relationship and impacts on target class for each of the top 20 features. Yellow highlighted features indicate a stronger impact in model prediction.

**Figure 13** Beeswarm Plot of Important Feature Relationships from Gradient Boosting from Iteration #7



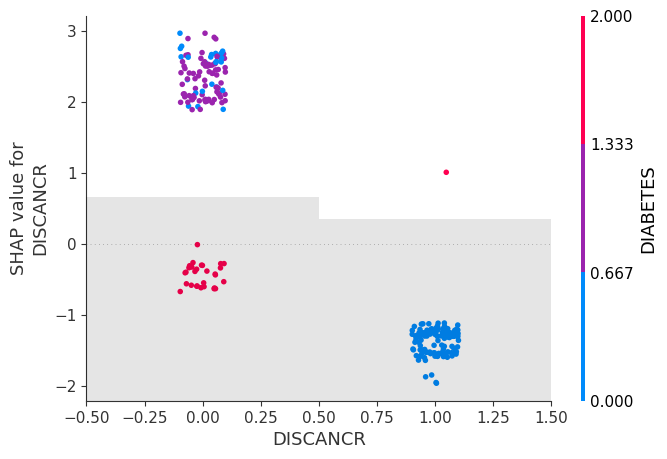
**Table 18** TOP 20 Important Features and Their Impacts on Gradient Boosting Model Prediction



Lastly, we focus on several selected features to study its impact on prediction together with their most highly related feature (***Figure 14*** *–* ***Figure 23****).*

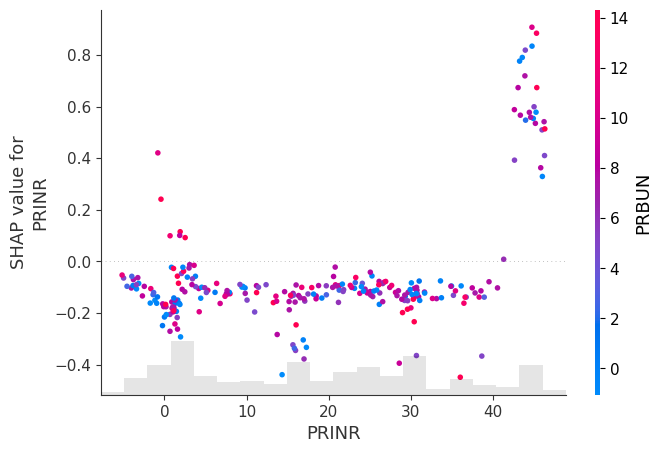
**Figure 14** Relationship Between DISCANCR and DIABETES and Their Impact on Prediction

When DISCANCR (Disseminated cancer) = 0 and DIABETES = 1 (MODERATE EXERTION), these observations tend to have a higher positive SHAP value meaning they are more likely needed for blood transfusion.



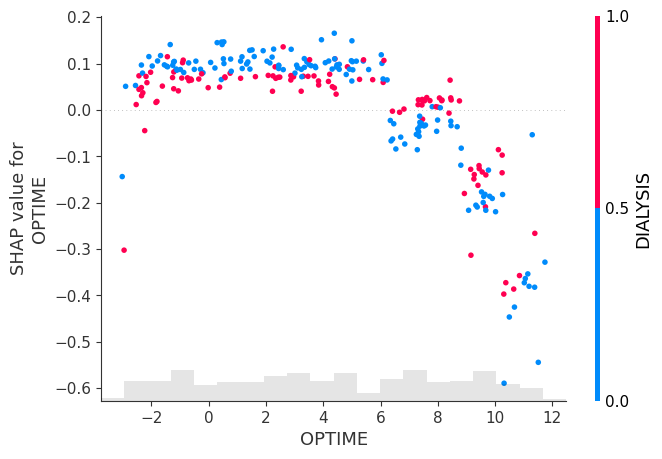
**Figure 15** Relationship Between PRINR and PRBUN and Their Impact on Prediction

A small sample of observations indicate that when PRINR (Days from INR Preoperative Labs to Operation) values are high, the higher value its most related feature PRBUN (Days from BUN Preoperative Labs to Operation) is, the more the more likely these observations require blood transfusion.



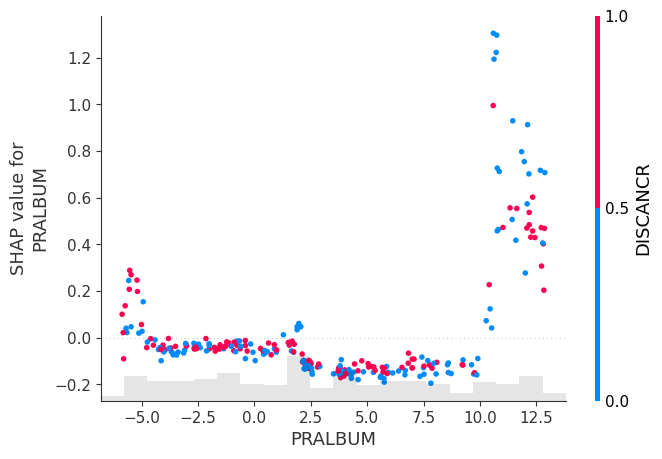
**Figure 16** Relationship Between OPTIME and DIALYSIS and Their Impact on Prediction

A strong negative relationship is observed: OPTIME (Total operation time) is negatively associated with blood transfusion, the longer OPTIME, the less likely blood transfusion is needed. Secondly, If DIALYSIS (Currently on dialysis (pre-op)) is “No”, it’s more likely that no blood transfusion is more likely to be needed.



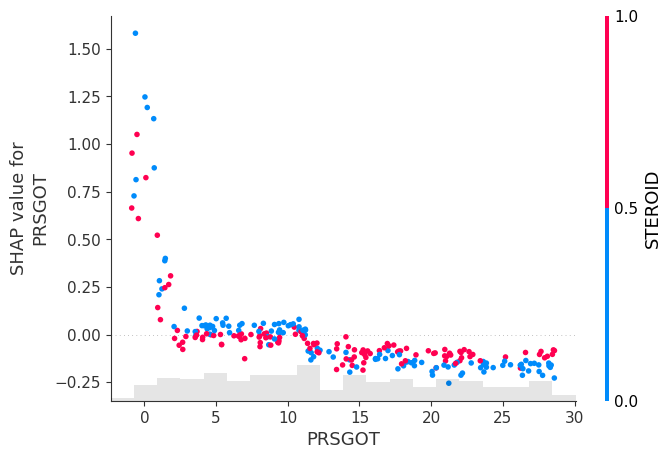
**Figure 17** Relationship Between PRALBUM and DISCANCR and Their Impact on Prediction

Some observations indicate that when PRALBUM (Pre-operative serum albumin) values are high, blood transfusion tends to be needed. And for those observations, DISCANCR (disseminated cancer) = 0 is more likely to cause transfusion than DISCANCR = 1.



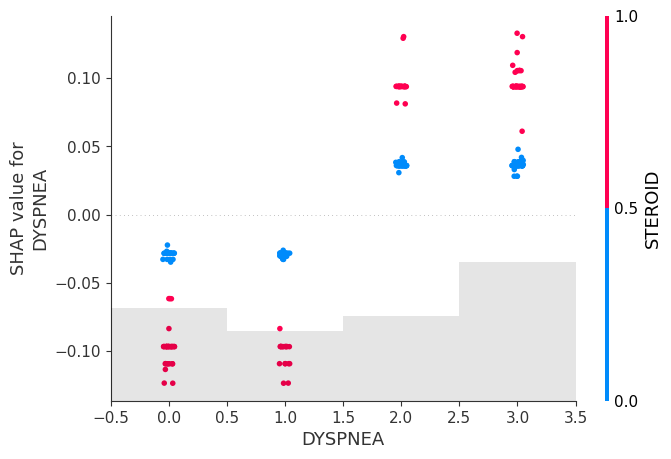
**Figure 18** Relationship Between PRSGOT and STEROID and Their Impact on Prediction

A small sample of observations indicate that when PRSGOT (Days from SGOT Preoperative Labs to Operation) values are low, blood transfusion is more likely to be needed.



**Figure 19** Relationship Between DYSPNEA and STEROID and Their Impact on Prediction

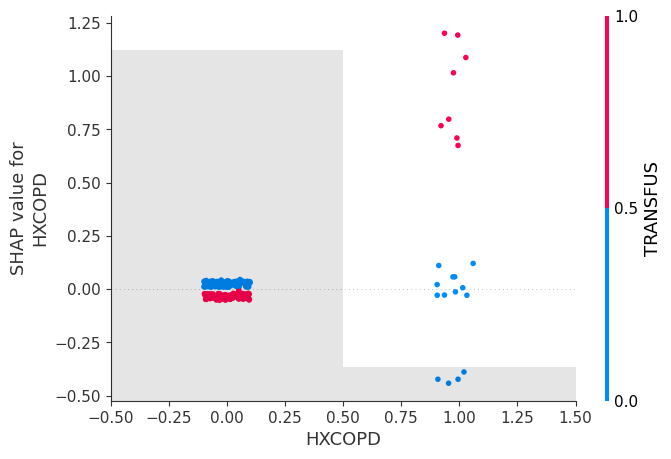
When DYSPNEA occurs, it is more likely that blood transfusion tends to be needed. On top of that, if STEROID (Immunosuppressive Therapy) is “Yes”, transfusion is even more likely.



**Figure 20** Relationship Between HXCOPD and TRANSFUS and Their Impact on Prediction

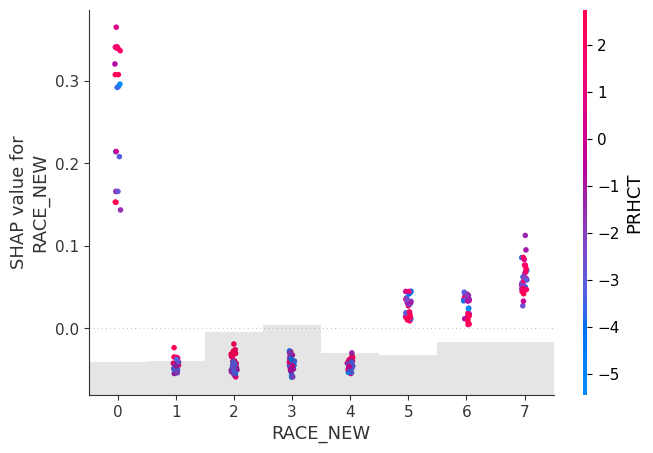
A small sample of observations indicate that HXCOPD (History of severe COPD) may lead to higher chance of blood transfusion, and if TRANSFUS (Preop Transfusion of >= 1 unit of whole/packed RBCs in 72 hours prior to surgery) = “Yes”, it’s even more likely (those red dots in the upper right quadrant.

The opposite has a much stronger pattern: no HXCOPD plus no TRANSFUS have little impact on transfusion.



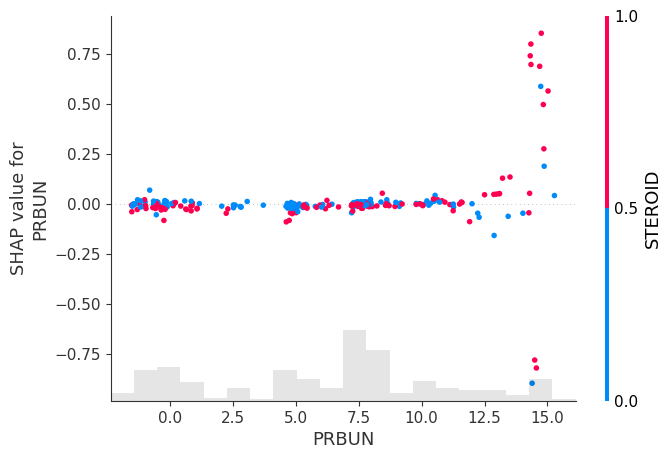
**Figure 21** Relationship Between RACE\_NEW and PRHCT and Their Impact on Prediction

A small sample of observations with RACE\_NEW = 0 (American Indian or Alaska Native) present higher chance for blood transfusion.



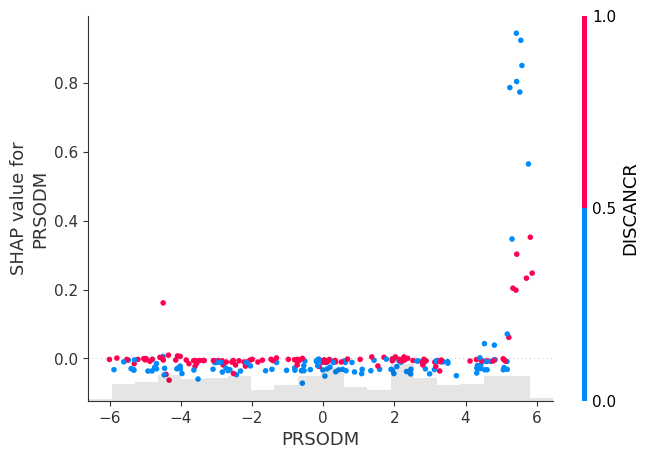
**Figure 22** Relationship Between PRBUN and STEROID and Their Impact on Prediction

A small sample of observations indicate that high value of PRBUN (Days from BUN Preoperative Labs to Operation) may have a higher chance leading to blood transfusion; among those, STEROID (Immunosuppressive Therapy) = “Yes” seems to have an even higher likelihood.



**Figure 23** Relationship Between PRSODM and DISCANCR and Their Impact on Prediction

A small sample of observations indicate higher PRSODM (Pre-operative serum sodium) may have a higher chance leading to blood transfusion, especially for those DISCANCR (Disseminated cancer) = “No”.



The following research activities can be conducted to confirm our research findings and continue to improve model prediction performance.

* Use automated libraries or other methods to repeat some data processing and feature engeering methods we used with our own code and compare the results. For example, Caret for data cleaning, and Featurewiz, Featuretools for feature engineering; GANs for synthetic data generation.
* Break down the target label into 3 classes – same as the original data (Transfusions; Intraop/Postop; No Complication) – instead of the current two (Yes and No) and see if this would improve model predictions.
* Use deep learning and neural networks methods to build more advanced models to further improve the performance.

# Conclusion

1. Gradient Boosting, Random Forest, XGBoost present constant model performance across all methods and iterations in the order of ranking.
2. It appears adding significantly more recent (2021-2022) or older data (2015-2017) may bring little improvement in model performance.
3. Properly applying data processing techniques and feature selection and engineering methods will help improve modeling results.
4. Synthetic data generation method (DataSynthesizer using Bayesian networks) may significantly improve model performance. Together with #3 above, our model evaluation metrics have superior to previous modeling work (ROC-AUC ranged from 0.76 - 0.86) with best model results of an accuracy score of 90.80, RMSE 0.31, F1-score 0.80, and ROC-AUC comes to 0.93.
5. DISCANCR, STEROID, DIABETES, AGE, and PRINR are the top five most important features that have a larger impact on predicting blood transfusions. DISCANCR also presents a far more important influence, whose feature importance values are four times higher than the second place STEROID.
6. DISCANCR and STEROID have a negative contribution relationship to the prediction target classes, while DIABETES, AGE, and PRINR have a positive impact. This means that for example, the higher value of DISCANCR (1), the more it contributes to a higher probability of predicting of the lower level class for the target (blood transfusion = 0), meaning the less likely blood transfusion would occur. The other example is AGE, and it has a positive impact on prediction target classes. The higher AGE value, the more it contributes to a higher probability of predicting of the higher level class for the target (blood transfusion = 1), meaning higher likelihood for blood transfusion.
7. Page 23-25 examined selected pairs among the top 20 features and their impacts on model prediction. A strong negative relationship is observed between OPTIME (total operation time) and blood transfusion: the longer the operation, the less likely blood transfusion occurs. In addition, for those having long operation time samples, if DIALYSIS = 0 (Currently on dialysis (pre-op) is “No”), the more likely blood transfusion occurs.

# References

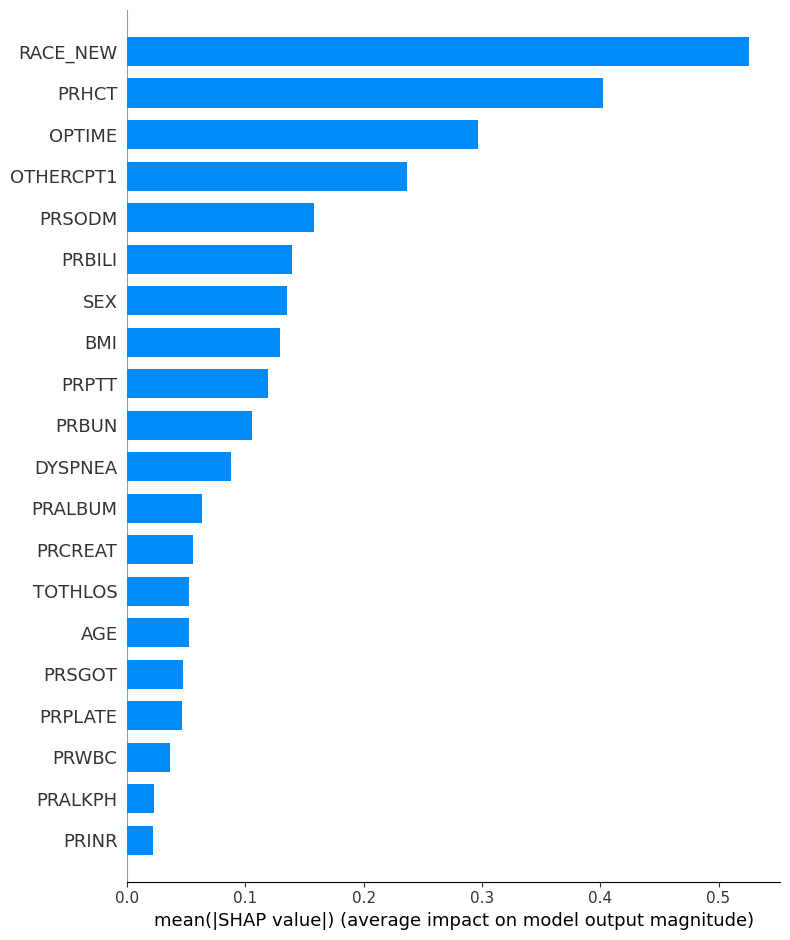
Gao, Y., Liu, X., Wang, L., Wang, S., Yu, Y., Ding, Y., . . . Ao, H. (2022, July 28). Machine learning algorithms to predict major bleeding after isolated coronary artery bypass grafting. *Front Cardiovasc Med.*

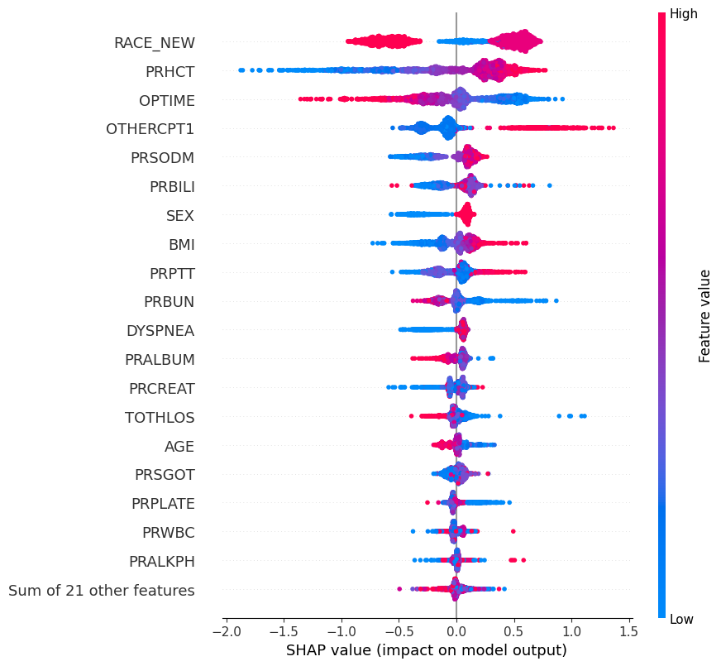
Li, Q., Lv, H., Chen, Y., Shen, J., Shi, J., Zhou, C., & Yan, F. (2024, April). Development and validation of a machine learning prediction model for perioperative red blood cell transfusions in a cardiac surgery. *International Journal of Medical Informatics, 184*.

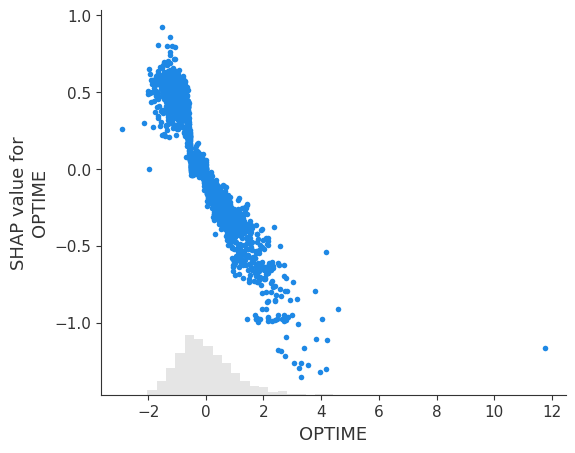
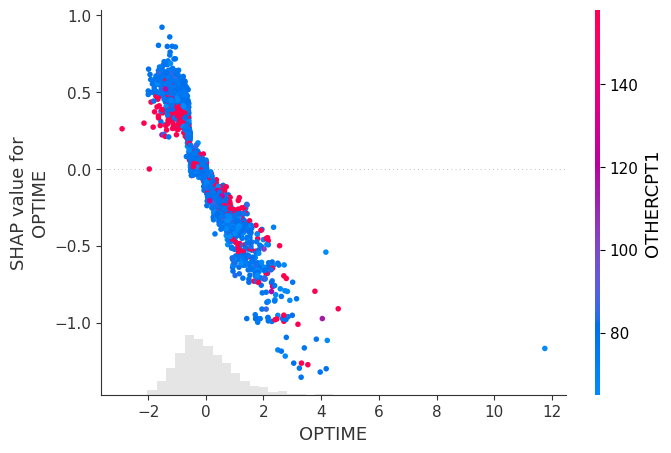
Tschoellitsch, T., Bock, C., Mahecic, T., Hofmann, A., & Meier, J. (2022, September). Machine learning-based prediction of massive perioperative allogeneic blood transfusion in cardiac surgery. *European Society of Anaesthesioloty and Intensive Care, 39*(9), 766-773.

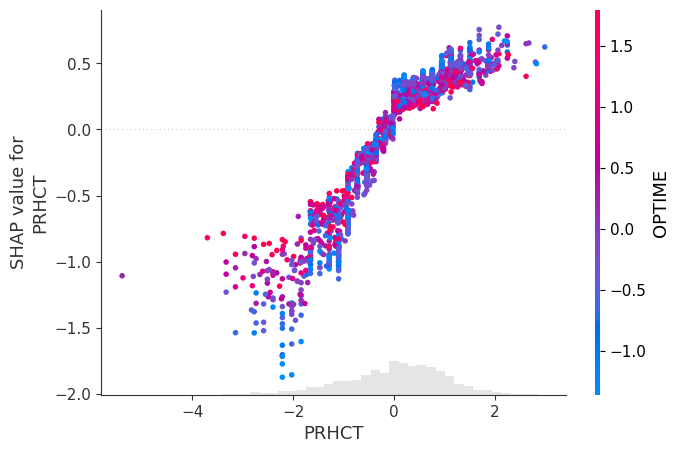
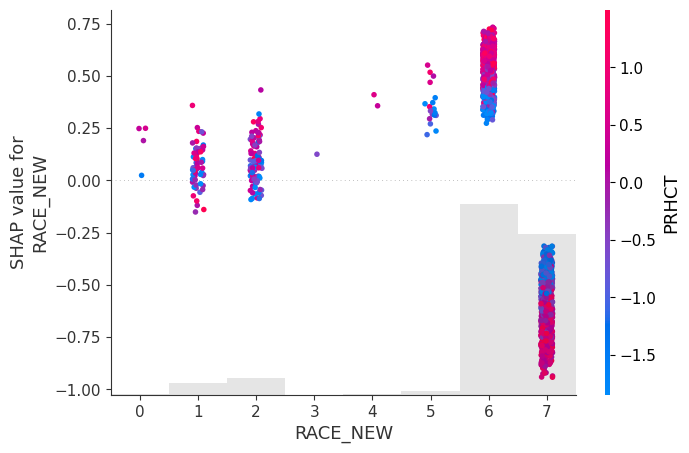
# Appendix

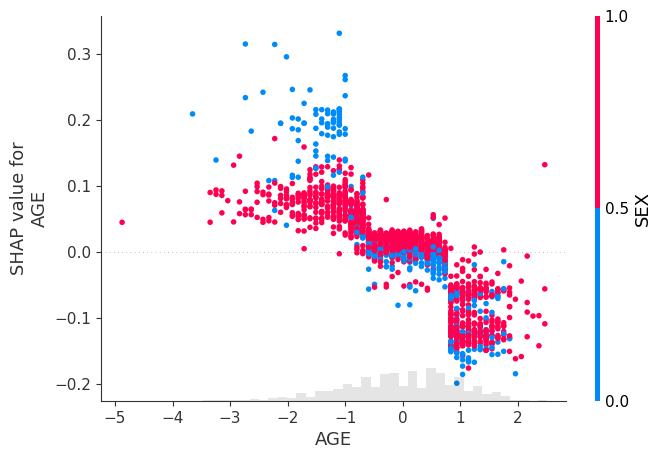
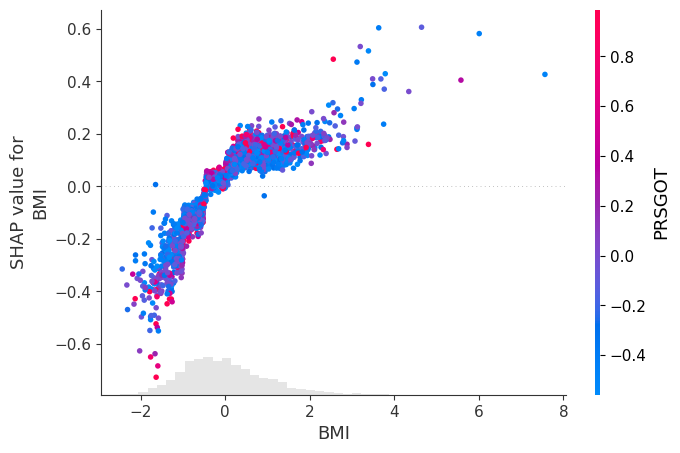
*Feature Impacts analysis from Gradient Boosting Model in Iteration #3*







1. Disseminated cancer [↑](#footnote-ref-1)
2. Immunosuppressive Therapy [↑](#footnote-ref-2)
3. Diabetes mellitus with oral agents or insulin [↑](#footnote-ref-3)
4. Days from Albumin Preoperative Labs to Operation [↑](#footnote-ref-4)