Outcomes after Thoracic Surgery for Patients with Lung Cancer

Capstone 2
V. Moore

Introduction

Lung cancer is a disease with a high rate of mortality, though the factors associated with success after surgery for tumor removal are unclear, particularly as a patient's health history can influence outcomes in ways that may not be readily apparent. This can be an important problem for patients and caregivers as they consider the risks of surgery.

Since thoracic surgery is not without its own inherent risks, it could be beneficial to patients and care providers to have some insight into expected risk/benefit scenarios as they may relate to a patient's own health history. Health policy organizations and payers could also benefit from such insight for planning for optimal health outcomes.

Data acquisition and preprocessing

Data used for this machine learning project will come from the University of California, Irvine, Machine Learning Repository, accessible here. This dataset includes one-year mortality outcomes for 470 patients, along with data for 16 features in each patient's health history. Patient data were collected from the Wroclaw Thoracic Surgery Centre in Poland from patients who experienced tumor resections for primary lung cancer at the Department of Thoracic Surgery of the Medical University of Wroclaw and Lower-Silesian Centre for Pulmonary Diseases, Poland, from 2007-2011. Data are part of Poland's National Lung Cancer Registry.

Table 1 shows the structure of the dataset used for this analysis. The features are found in all columns except for the final column, "Risk1Y", which represents the target variable. The first feature, "DGN", represents diagnosis codes regarding the primary tumor or tumors for each patient, as combinations of ICD-10 codes, and they are here identified as DGN1, DGN2, DGN3, DGN4, DGN5, DGN6, or DGN8. Forced vital capacity is shown in the "FVC" column, and this is a continuous non-integer numeric variable. Forced expiratory volume is shown in the "FEV1" column, and this, too, is a continuous non-integer numeric variable. Values on the Zubrod scale are shown in the next column, and this scale represents a patient's performance status, with 0 being asymptomatic and worsening symptomatic states reflected in higher numbers, with 4 representing a bedbound patient and 5 representing death. This score is also known as the Eastern Cooperative Oncology Group (ECOG) score. This study includes only those patients with Zubrod scores of 0, 1, or 2 (perhaps relating to fitness for major surgery).

Pain represents whether the patient experienced pain prior to surgery, and the dyspnea, cough, and weakness also refer to Boolean values for presence in a patient prior to surgery. Tumor size refers to size of the original tumor, with OC11 being the smallest, ascending to OC14 for the largest, with OC12 and OC13 being intermediate.

Type II diabetes mellitus ("T2DM"), myocardial infarction ("MI"), peripheral arterial disease ("PAD"), and asthma are all shown as Boolean variables for presence or absence of these medical conditions prior to surgery. Smoking is present as a Boolean as well, and age refers to age at time of surgery.

The target variable, “Risk1Y”, is present as a Boolean variable with “T” occurring if the patient died during the one-year period following surgery.

Table 1. Dataset structure for features tested in this analysis

	DGN	FVC	FEV1	Zubrod	Pain	Haemoptysis	Dyspnea	Cough	Weakness	Tumor_size	T2DM	MI	PAD	Smoking	Asthma	Age	Risk1Y
0	DGN2	2.88	2.16	PRZ1	F	F	F	T	T	OC14	F	F	F	T	F	60	F
1	DGN3	3.40	1.88	PRZ0	F	F	F	F	F	OC12	F	F	F	T	F	51	F
2	DGN3	2.76	2.08	PRZ1	F	F	F	T	F	OC11	F	F	F	T	F	59	F
3	DGN3	3.68	3.04	PRZ0	F	F	F	F	F	OC11	F	F	F	F	F	54	F
4	DGN3	2.44	0.96	PRZ2	F	T	F	T	T	OC11	F	F	F	T	F	73	T

The dataset originally contained no column names, so those were added. There were no missing values for these variables in this population of n=470. The Boolean variables appearing as “T” or “F” were replaced with a 0 for each “F” entry and a 1 for each “T” entry. The values that contained a combination of letters and numbers had the letters stripped from them so that all remained in those columns were the numerical portions of the values. All features became set as integers apart from FVC, FEV1, and age, which were all cast as floats.

Exploratory data analysis

The first step in analysis of this dataset involving thoracic surgery outcomes for patients with lung cancer was to examine histograms of all feature and target data in order to assess distributions and be able to understand the overall nature of the data. Many distributions were somewhat imbalanced, and, importantly, the target data about mortality were imbalanced. Figure 1 shows a thumbnail image of the histograms from this dataset.

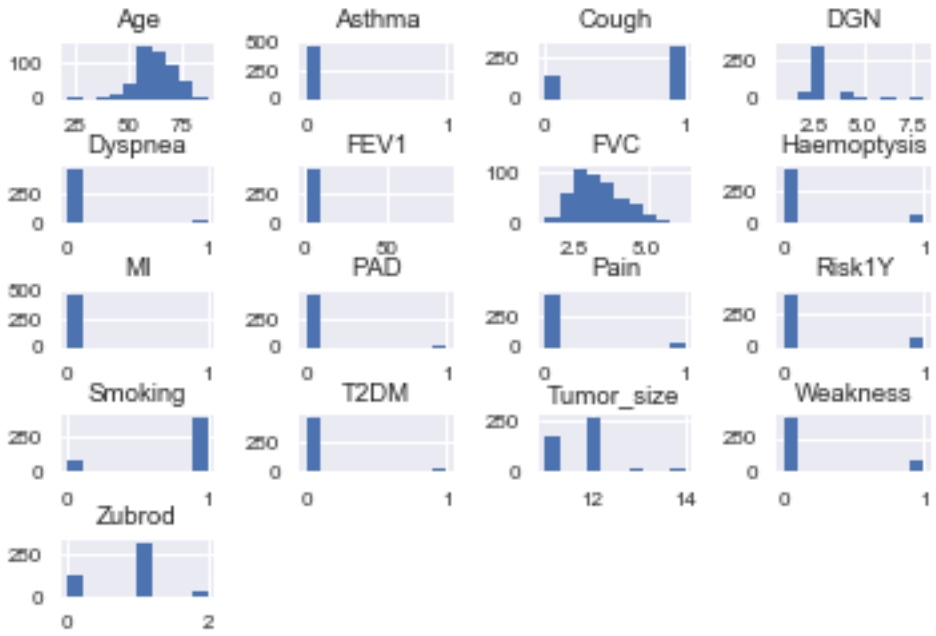


Figure 1. Histograms of feature and target data from the thoracic surgery dataset.

Three columns of data contain continuous variables, being Age, FVC, and FEV1, though FEV1 does not appear as such in Figure 1, in which it appears to have an extraordinarily long x-axis that may be indicative of outliers.

Figures 2 and 3 show simple regression plots of age and FVC or age and FEV1, respectively, with discrimination between survival (0, blue) and mortality at one year (1, green) shown for each.

From Figure 2 we can see that there is a tremendous amount of scatter within the data between age and FVC, and the trendline through these data for patients who passed away during the study shows a very large confidence interval. For both categories of survival data, with advancing age the FVC values trend slightly downward, though they perhaps started at a lower level for those who were younger who passed away during the one-year period following surgery.

From Figure 3, which plots age versus FEV1 values, it is apparent that there are approximately 14 obvious outliers in the FEV1 data. It is highly unlikely that for a feature for which most patients vary from each other by a few integers and with most data points below 10, that the data points that appear above, for instance, 40 along the y-axis are accurate if scaled with the same units as the rest of the data are.

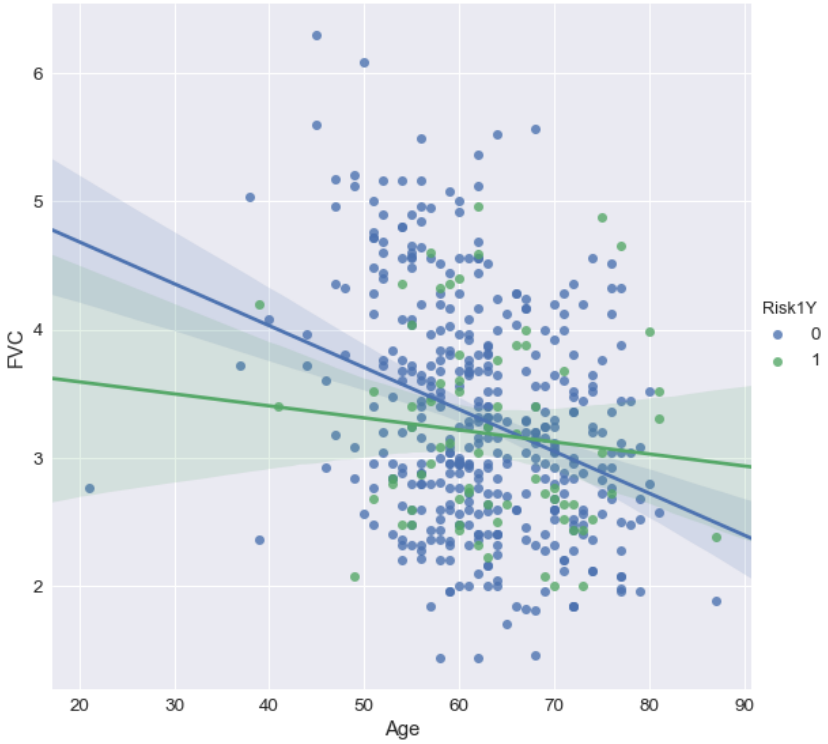


Figure 2. Age versus FVC, by one-year mortality.

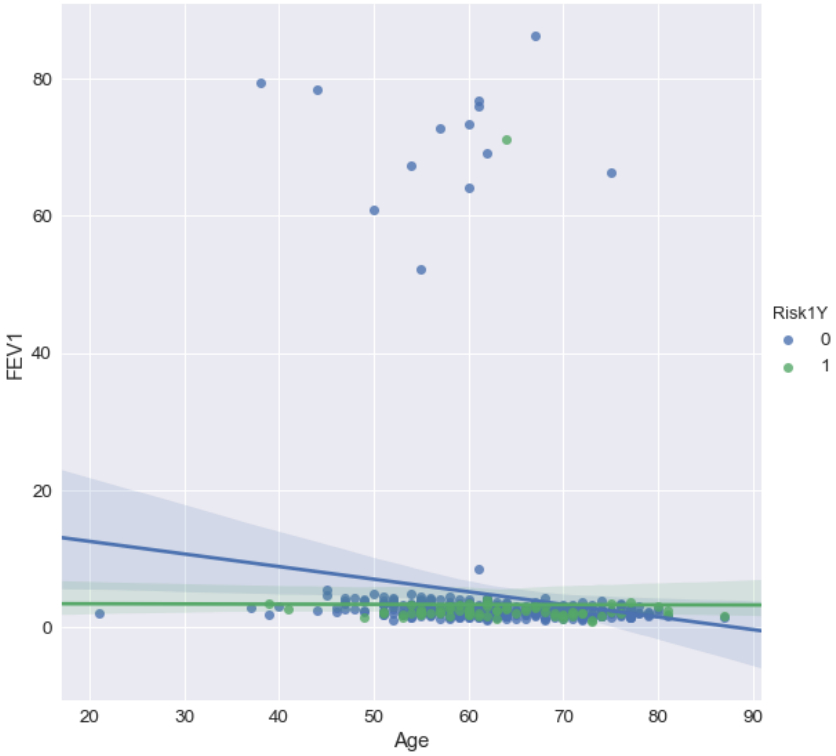


Figure 3. Age versus FEV1, by one-year mortality.

Analyzing the relationship between age and FEV1 further through ordinary least-squares regression using Python's Statsmodels package reveals that these 14 high-FEV1 data points fall quite far from their predicted range (Figure 4) and that their residual values depart greatly from the line of best fit (Figure 5).

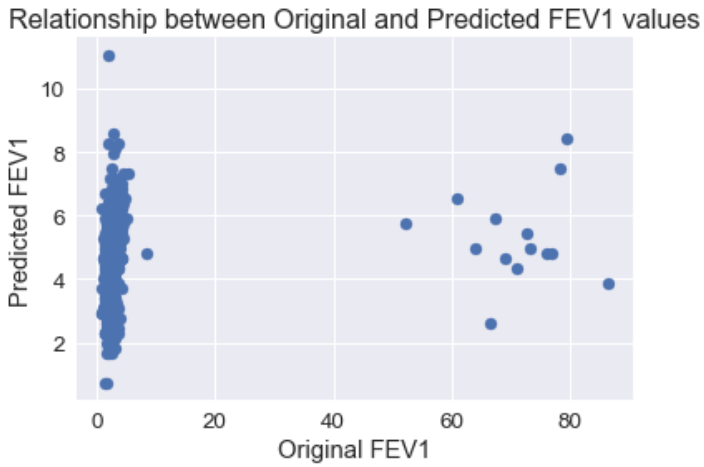


Figure 4. Actual (original) FEV1 values versus predicted FEV1 values.

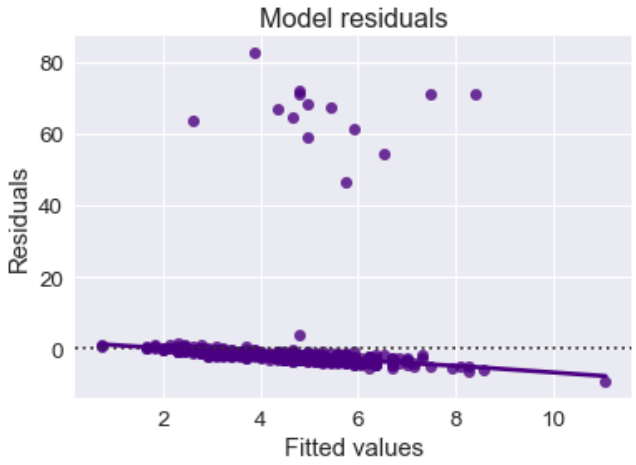


Figure 5. FEV1 residuals versus the model line of best fit.

In a separate ordinary least-squares regression analysis of mortality data fitted against FEV1 data, it appears at least one of the data points (which is for an entry marked with an FEV1 of 71.1 in a patient who passed away within a year of surgery) imparts an outsized influence on this regression (Figure 6).

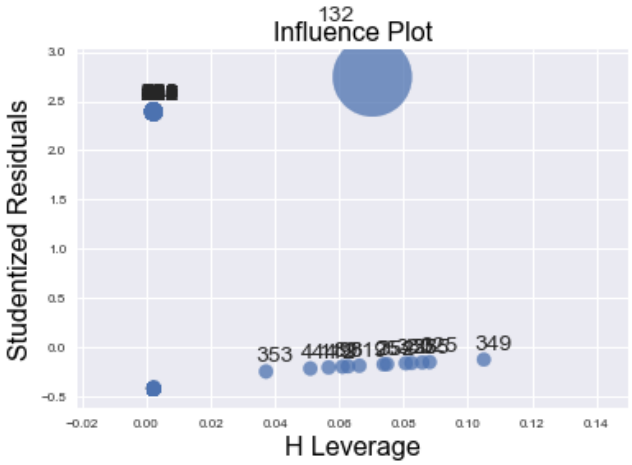


Figure 6. Influence plot (based on Cook's distances for each entry) of residuals for mortality data regressed with FEV1 data.

Assuming that the 14 FEV1 likely outliers exist in the dataset in error, it may be prudent to remove them from the dataset prior to predictive modeling. Incorrect information is not helpful in designing a predictive model, and the dataset of n=470 is large enough that removal of 14 entries should not cripple model development. The next step in treatment of this dataset was to remove these entries with FEV1 values above 40 units, resulting in n=456 for further analysis. All interpretations presented from this point forward do not contain data using the 14 identified outliers.

Figure 7 shows a new residual plot for the FEV1 data regressed against age.

Figure 7. FEV1 residuals versus the model line of best fit after removal of outliers.

Figure 8 shows another presentation of age versus FEV1 data, but this time with the described outliers removed. The pattern for these data is nearly identical to what we saw in Figure 4 for age versus FVC.

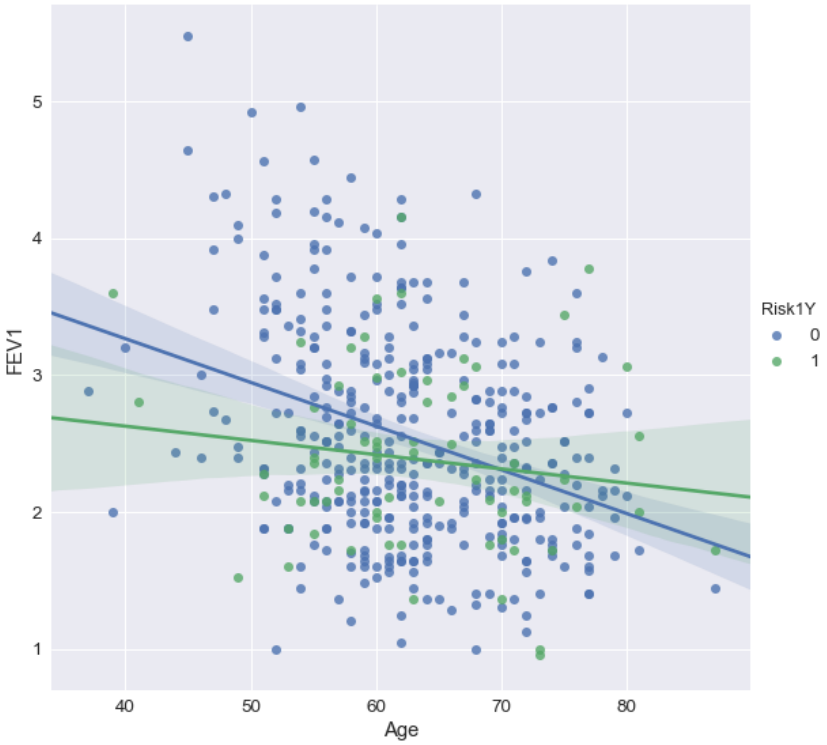


Figure 8. Age versus FEV1 with outliers removed.

While tumor size appears in the dataset as a discrete variable, we can examine it versus age, as in Figure 9 to get a rough idea of any patterns.

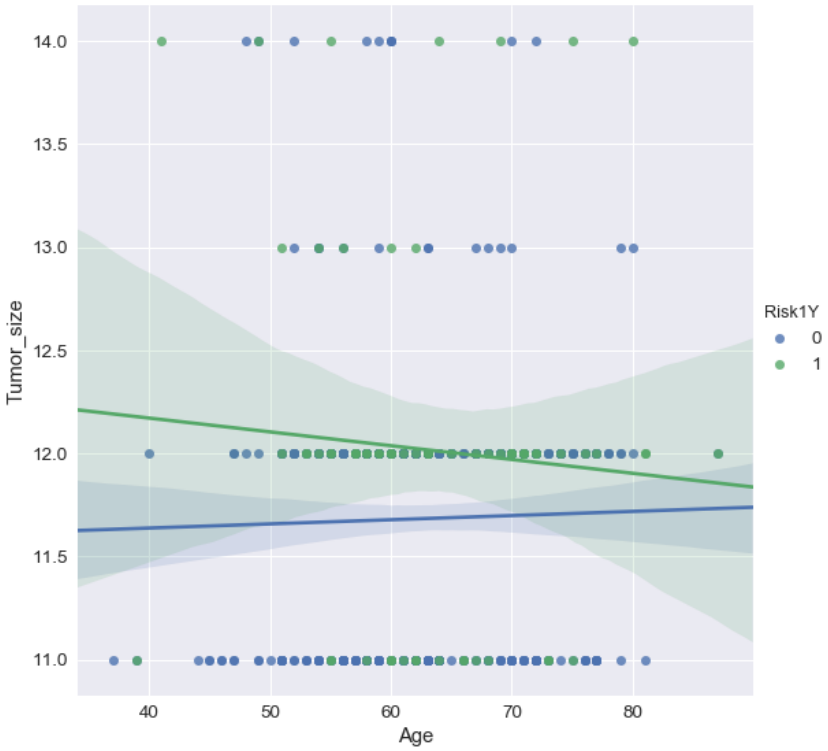
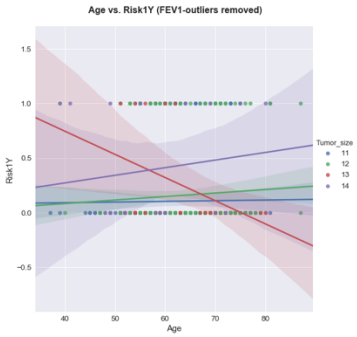
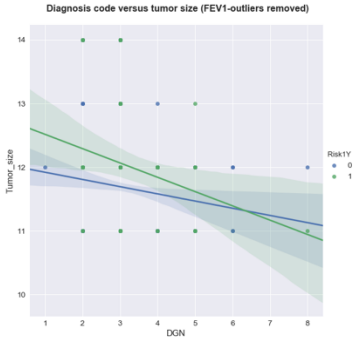


Figure 9. Age versus tumor size, by one-year mortality.
Most tumors in this dataset appeared to be in the smaller size groups (categories 11 and 12), and survivors overall showed smaller tumors, though there is considerable overlap in confidence intervals. Across ages, tumor size did not appear to vary much for survivors. For those patients who passed away, younger patients seem to have had larger tumors, and mortality with smaller tumors increased as patients were older. ***(Some of the following figures have interesting content, but I don't know if I will keep these messy figures in or not.)***





Overall the dataset seems to be low in correlations among variables with each other, as shown in the heatmap of Pearson's correlation coefficients in Figure 10. Most variables are categorical, so Pearson's correlation coefficient is not the best statistics for measuring their relationships, but it can give a quick view of relationships.

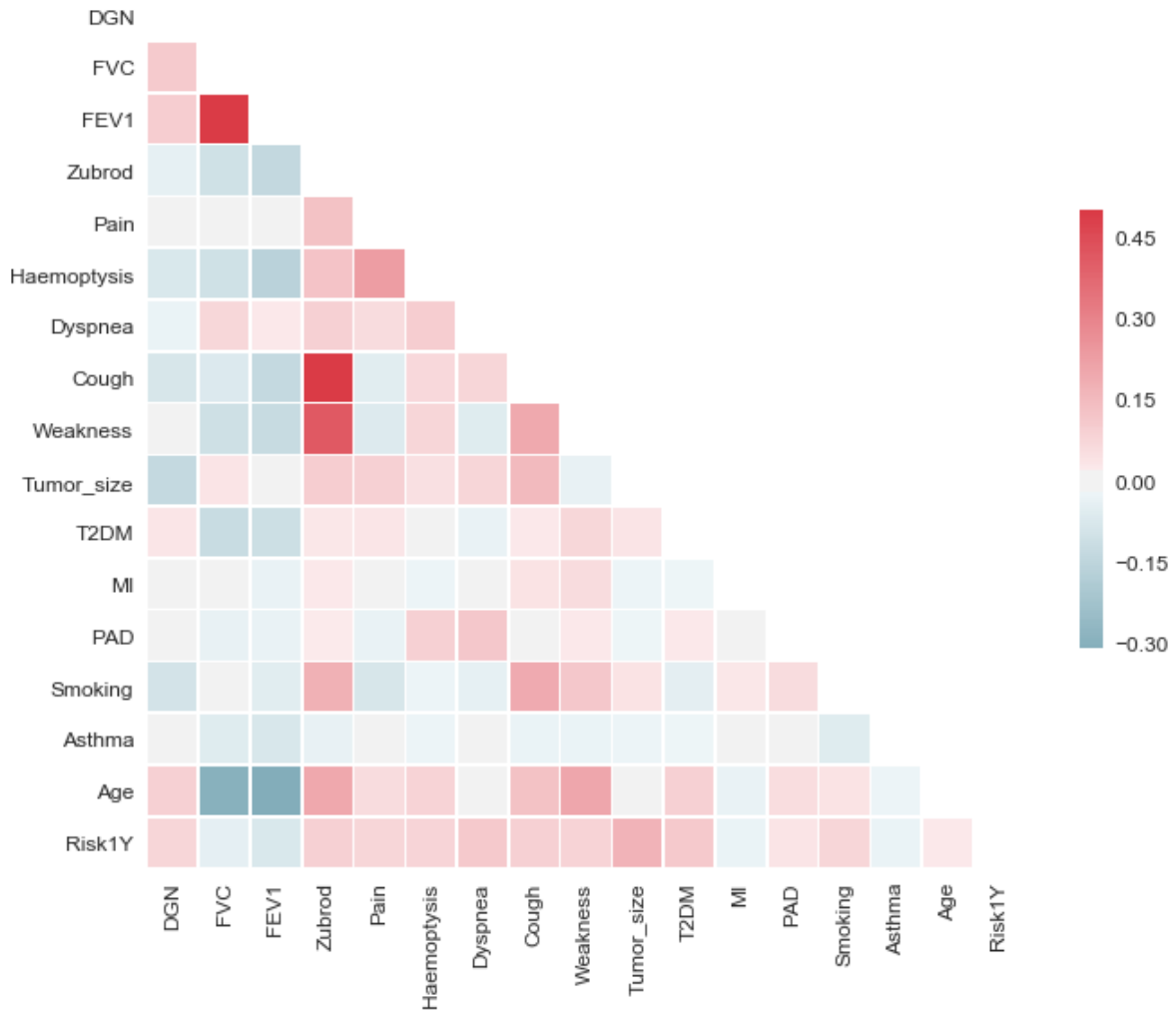


Figure 10. Heatmap of correlations among variables of this dataset.

Most relationships in Figure 10 show correlation statistics near 0, with the exceptions of positive relationships to be spotted for FVC plotted against FEV1 and cough and weakness each plotted against Zubrod score. None of these are surprising. FEV1 and FVC are both very similar metrics that measure breathing capacity, so they would be expected to covary. As Zubrod score is a metric of a person's level of function, weakness would be expected to appear with a worse Zubrod score. Severity of cough seems to relate to Zubrod score more strongly, but this, too, might be expected for patients who are suffereing with a more severe lung condition.

Negative relationships appear for both age with FVC and age with FEV1. It is again unsurprising that FVC and FEV1 would trend the same way in this case, but it may not necessarily be obvious that age would, on its own, have a negative effect on a person's respiratory capacity.

More quantitative analyses of relationships among variables show that the Pearson correlation coefficients are all highly statistically significant among continuous variables in this study, with FVC and FEV1 showing a Pearson coefficient of 0.888 (p=0.0). Age shows Pearson coefficients of -0.299 (p=0.0) and -0.031 (p=0.0) for FVC and FEV1, respectively.

Chi-squared statistics of relationships between categorical variables in this study show most variables to not covary, with a few exceptions. Relationships that are present a level of $p<0.001$ exist for Zubrod score with cough ($\chi^2 = 259.27$), Zubrod with weakness ($\chi^2 =103.59$), Zubrod with smoking ($\chi^2 =16.53$), pain with haemoptysis ($\chi^2 =20.38$), pain with tumor size ($\chi^2 =16.48$), cough with weakness ($\chi^2 =16.93$), cough with smoking ($\chi^2 =16.01$), and myocardial infarction with asthma ($\chi^2 =27.63$).

Chi-squared statistics that are statistically significant at the level of $0.001 \leq p < 0.05$ include diagnosis code with one-year mortality ($\chi^2 =21.56$), Zubrod score with pain ($\chi^2 =8.21$), Zubrod score with haemoptysis ($\chi^2 =7.76$), Zubrod with tumor size ($\chi^2 =14.86$), dyspnea with tumor size ($\chi^2 =10.24$), dyspnea with one-year mortality ($\chi^2 =4.5$), cough with tumor size ($\chi^2 =10.8$), tumor size with one-year mortality ($\chi^2 =14.28$), and peripheral arterial disease with asthma ($\chi^2 =6.27$). The remaining 61 relationships tested with chi-squared analysis revealed no close associations.

Overall, this appears to be a dataset for which predictive modeling should be able to proceed without serious concerns of covariance among variables. It is perhaps unnecessary for FVC and FEV1 to both be included, but this can be considered if there are difficulties in generating a model with good accuracy metrics.

Analytical approach

The target variable to be predicted in our machine learning model is mortality assessed at one year post-surgery, which is organized within the data as a binary variable of survival versus mortality. We can choose a binary classification predictive approach based on labeled data, and a supervised classification model (such as random forest or XGBoost) would be a prudent approach to predict probability of categorization into either class. The dataset shows imbalance in mortality data and some other features, so we can apply SMOTE or another resampling technique to the dataset to generate synthetic samples through interpolation prior to modeling. We can use SHAP values to determine feature importance collectively or individually. As appropriate, based on patterns that emerge from these values, we can also re-examine the model to focus on complex features individually against the rest of the features. Most features are present as Boolean values or categorical variables representing diagnosis codes or classes of tumor size. However, some features, such as age, forced vital capacity, and forced expiration volume, are continuous. All potentially can contribute to results in a complex fashion, however.

Even among supervised classification model types there are several options, including classifier and hyperparameter options. The dataset does not included missing data, so there is no need to consider imputing data or restricting analysis to a model that can handling missing data. Options for classifiers and potential hyperparameters can be evaluated on the training data using an algorithm first described here here , updated for Python 3 here, and with further modifications added in order to add more classifiers and hyperparameters for this study. Table 2 shows the results of this analysis based on training data that have had SMOTE treatment applied to them. Prior to SMOTE, data were split into training and testing subsets (75% of data for training, 25% for testing).

Table 2. Ranking of tuned models on SMOTE-treated training data

	estimator	min_score	mean_score	max_score	std_score	C	gamma	kernel	learning_rate	max_depth	min_child_weight	n_estimators
2	ExtraTreesClassifier	0.790419	0.884606	0.938776	0.0668502	NaN	NaN	NaN	NaN	NaN	NaN	100
0	ExtraTreesClassifier	0.807229	0.880992	0.94359	0.0562261	NaN	NaN	NaN	NaN	NaN	NaN	16
1	ExtraTreesClassifier	0.780488	0.878229	0.938776	0.0697677	NaN	NaN	NaN	NaN	NaN	NaN	32
5	RandomForestClassifier	0.666667	0.840078	0.938144	0.122971	NaN	NaN	NaN	NaN	NaN	NaN	100
4	RandomForestClassifier	0.644295	0.838666	0.942408	0.137545	NaN	NaN	NaN	NaN	NaN	NaN	32
3	RandomForestClassifier	0.649007	0.825538	0.927083	0.125296	NaN	NaN	NaN	NaN	NaN	NaN	16
19	XGBClassifier	0.625	0.811982	0.915423	0.132465	NaN	NaN	NaN	NaN	20	NaN	NaN
14	GradientBoostingClassifier	0.630137	0.805018	0.902564	0.123935	NaN	NaN	NaN	1	NaN	NaN	100
13	GradientBoostingClassifier	0.62585	0.797504	0.90099	0.122231	NaN	NaN	NaN	1	NaN	NaN	32
16	XGBClassifier	0.6	0.796051	0.901554	0.138763	NaN	NaN	NaN	NaN	NaN	NaN	32
11	GradientBoostingClassifier	0.591549	0.790465	0.905473	0.141226	NaN	NaN	NaN	0.8	NaN	NaN	100
18	XGBClassifier	0.565217	0.788103	0.909091	0.157796	NaN	NaN	NaN	NaN	4	NaN	NaN
12	GradientBoostingClassifier	0.60274	0.778511	0.88	0.124784	NaN	NaN	NaN	1	NaN	NaN	16
15	XGBClassifier	0.621951	0.770931	0.862559	0.10627	NaN	NaN	NaN	NaN	NaN	NaN	16
23	XGBClassifier	0.544118	0.770461	0.917526	0.162424	NaN	0.1	NaN	NaN	NaN	NaN	NaN
22	XGBClassifier	0.540146	0.766109	0.912821	0.162136	NaN	0.01	NaN	NaN	NaN	NaN	NaN
20	XGBClassifier	0.540146	0.766109	0.912821	0.162136	NaN	NaN	NaN	NaN	NaN	1	NaN
17	XGBClassifier	0.540146	0.766109	0.912821	0.162136	NaN	NaN	NaN	NaN	NaN	NaN	100
10	GradientBoostingClassifier	0.529412	0.762074	0.881188	0.164533	NaN	NaN	NaN	0.8	NaN	NaN	32
9	GradientBoostingClassifier	0.554745	0.761682	0.865979	0.146328	NaN	NaN	NaN	0.8	NaN	NaN	16
6	AdaBoostClassifier	0.598726	0.760191	0.843137	0.114187	NaN	NaN	NaN	NaN	NaN	NaN	16
21	XGBClassifier	0.602564	0.756589	0.854369	0.110224	NaN	NaN	NaN	NaN	NaN	10	NaN
8	AdaBoostClassifier	0.492537	0.742172	0.881188	0.176899	NaN	NaN	NaN	NaN	NaN	NaN	100
7	AdaBoostClassifier	0.540146	0.724799	0.868293	0.13709	NaN	NaN	NaN	NaN	NaN	NaN	32
28	SVC	0.694444	0.723312	0.745763	0.0214364	10	0.001	rbf	NaN	NaN	NaN	NaN
24	SVC	0.682692	0.697809	0.720379	0.0162633	1	NaN	linear	NaN	NaN	NaN	NaN
25	SVC	0.673575	0.69684	0.724638	0.0210911	10	NaN	linear	NaN	NaN	NaN	NaN
31	LogisticRegression	0.643564	0.671679	0.701422	0.0236482	0.1	NaN	NaN	NaN	NaN	NaN	NaN
30	LogisticRegression	0.639175	0.665085	0.715686	0.0357838	0.01	NaN	NaN	NaN	NaN	NaN	NaN
34	LogisticRegression	0.632124	0.65759	0.680851	0.0199536	100	NaN	NaN	NaN	NaN	NaN	NaN
33	LogisticRegression	0.616162	0.652171	0.673684	0.025623	10	NaN	NaN	NaN	NaN	NaN	NaN
32	LogisticRegression	0.622449	0.646094	0.67	0.0194135	1	NaN	NaN	NaN	NaN	NaN	NaN
29	SVC	0.542714	0.624169	0.687259	0.0604223	10	0.0001	rbf	NaN	NaN	NaN	NaN
26	SVC	0.552764	0.61075	0.646154	0.0413354	1	0.001	rbf	NaN	NaN	NaN	NaN
27	SVC	0.443182	0.547079	0.679537	0.0985822	1	0.0001	rbf	NaN	NaN	NaN	NaN

From Table 2 it is evident that the extra trees and random forest classifiers, in that order, perform the best on these SMOTE-adjusted training data for any number of estimators. The highest mean accuracy score is for the extra trees classifier with a score of 0.884. The boosting classifiers follow these tree-based ensemble methods, with SVC and logistic regression methods appearing last in this approach.

These results look pretty promising for model accuracy, and they can give a good indication of model choice to apply to test data, but accuracy with training data does not necessarily translate to accuracy with test data.

Figure 11 shows a series of receiver-operating curves for extra trees, random forest, and XGBoost models applied to this dataset's test data, based on training with SMOTE-treated data.

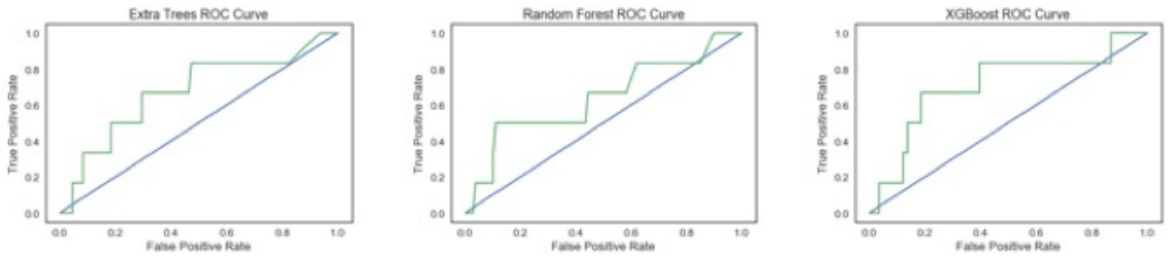


Figure 11. ROCs for the top three model classifiers for this study: extra trees, random forest, and XGBoost.

For each of the models examined in Figure 11, there appears to be greater accuracy than what is expected by randomness, and in fact it could be very difficult to predict mortality from a condition within a year, but Table 3 reveals a possible issue with model development so far.

Table 3. Classifiers and accuracy, precision, and recall scores from SMOTE-treated training data.

	Accuracy	Class	Precision	Recall	F1-score	Support
Extra trees:						
Training	1.0					
Testing	0.877					
AU-ROC	0.6736					
		0	0.95	0.92	0.93	108
		1	0.10	0.17	0.12	6
		Avg/total	0.91	0.88	0.89	114
Random forest:						
Training	1.0					
Testing	0.904					
AU-ROC	0.6404					
		0	0.95	0.94	0.95	108
		1	0.14	0.15	0.15	6
		Avg/total	0.91	0.91	0.91	114
XGBoost:						
Training	1.0					
Testing	0.868					
AU-ROC	0.7083					
		0	0.95	0.91	0.93	108
		1	0.09	0.17	0.12	6
		Avg/total	0.91	0.87	0.89	114

While accuracy (sum of true positives and true negatives over the total) and precision (sum of true positives over predicted positives) show each of these tests to perform well for identifying those patients who survive over the one-year period, each analysis shows low scores for precision and recall for those who do not survive one year. This likely relates to the imbalance in data, which SMOTE treatment was expected to assist with, but it does not seem that this approach worked well here.

SMOTE is not the only method by which sampling of data can balance classes within a dataset for predictive analysis. While SMOTE is based on sampling based on nearest-neighbor distances between points in order to bolster the number of data points, another approach that can be taken is to resample data

points from the minority class. In the case of the training partition of this dataset, there are 277 members in the survivor class, compared with 63 members in the non-surviving class. A resampling approach to balance these class is to "upsample" the minority class (non-survivors in this case) so that each class includes 277 samples. The next analyses show modeling results for data that have been upsampled in such a manner after data were split into training and test sets (75% of data for training, 25% for testing). Table 4 shows the classifier and hyperparameter optimization test on upsampled training data (just as performed to generate Table 3).

Table 4. Ranking of tuned models on minority-upsampled training data

	estimator	min_score	mean_score	max_score	std_score	C	gamma	kernel	learning_rate	max_depth	min_child_weight	n_estimators
5	RandomForestClassifier	0.918367	0.93587	0.946237	0.0124461	NaN	NaN	NaN	NaN	NaN	NaN	100
2	ExtraTreesClassifier	0.932642	0.935717	0.941176	0.00387041	NaN	NaN	NaN	NaN	NaN	NaN	100
3	RandomForestClassifier	0.916667	0.933683	0.946237	0.0124773	NaN	NaN	NaN	NaN	NaN	NaN	16
1	ExtraTreesClassifier	0.907216	0.932357	0.956522	0.0201406	NaN	NaN	NaN	NaN	NaN	NaN	32
4	RandomForestClassifier	0.927835	0.93232	0.939394	0.00506178	NaN	NaN	NaN	NaN	NaN	NaN	32
0	ExtraTreesClassifier	0.9	0.931384	0.947917	0.0222027	NaN	NaN	NaN	NaN	NaN	NaN	16
19	XGBClassifier	0.895522	0.905759	0.919192	0.00992369	NaN	NaN	NaN	NaN	20	NaN	NaN
11	GradientBoostingClassifier	0.895522	0.904224	0.919192	0.0106302	NaN	NaN	NaN	0.8	NaN	NaN	100
14	GradientBoostingClassifier	0.882353	0.903454	0.920792	0.0159167	NaN	NaN	NaN	1	NaN	NaN	100
10	GradientBoostingClassifier	0.895522	0.896678	0.897959	0.000998802	NaN	NaN	NaN	0.8	NaN	NaN	32
13	GradientBoostingClassifier	0.878049	0.895902	0.916256	0.015698	NaN	NaN	NaN	1	NaN	NaN	32
9	GradientBoostingClassifier	0.864322	0.875916	0.887805	0.00958926	NaN	NaN	NaN	0.8	NaN	NaN	16
18	XGBClassifier	0.848485	0.873168	0.886598	0.0174764	NaN	NaN	NaN	NaN	4	NaN	NaN
12	GradientBoostingClassifier	0.835821	0.867713	0.907317	0.0296933	NaN	NaN	NaN	1	NaN	NaN	16
23	XGBClassifier	0.836735	0.846701	0.865672	0.0134201	NaN	0.1	NaN	NaN	NaN	NaN	NaN
22	XGBClassifier	0.829787	0.844946	0.865672	0.0151695	NaN	0.01	NaN	NaN	NaN	NaN	NaN
20	XGBClassifier	0.829787	0.844946	0.865672	0.0151695	NaN	NaN	NaN	NaN	NaN	1	NaN
17	XGBClassifier	0.829787	0.844946	0.865672	0.0151695	NaN	NaN	NaN	NaN	NaN	NaN	100
8	AdaBoostClassifier	0.791667	0.801017	0.817734	0.0118483	NaN	NaN	NaN	NaN	NaN	NaN	100
16	XGBClassifier	0.737968	0.755537	0.775956	0.015639	NaN	NaN	NaN	NaN	NaN	NaN	32
7	AdaBoostClassifier	0.687831	0.739915	0.776596	0.0378398	NaN	NaN	NaN	NaN	NaN	NaN	32
21	XGBClassifier	0.705263	0.731954	0.748663	0.019072	NaN	NaN	NaN	NaN	NaN	10	NaN
15	XGBClassifier	0.663043	0.717782	0.764045	0.0416669	NaN	NaN	NaN	NaN	NaN	NaN	16
6	AdaBoostClassifier	0.700508	0.715896	0.733668	0.0136423	NaN	NaN	NaN	NaN	NaN	NaN	16
34	LogisticRegression	0.592179	0.6323	0.659218	0.0289171	100	NaN	NaN	NaN	NaN	NaN	NaN
25	SVC	0.587571	0.631284	0.659574	0.0313532	10	NaN	linear	NaN	NaN	NaN	NaN
33	LogisticRegression	0.596685	0.627877	0.655367	0.0240995	10	NaN	NaN	NaN	NaN	NaN	NaN
24	SVC	0.571429	0.610722	0.630435	0.0277848	1	NaN	linear	NaN	NaN	NaN	NaN
31	LogisticRegression	0.557377	0.608379	0.666667	0.0449138	0.1	NaN	NaN	NaN	NaN	NaN	NaN
28	SVC	0.549451	0.603264	0.640777	0.0390249	10	0.001	rbf	NaN	NaN	NaN	NaN
30	LogisticRegression	0.568306	0.602031	0.657143	0.039294	0.01	NaN	NaN	NaN	NaN	NaN	NaN
32	LogisticRegression	0.569832	0.601551	0.633663	0.0260604	1	NaN	NaN	NaN	NaN	NaN	NaN
27	SVC	0.494253	0.509745	0.538012	0.0200185	1	0.0001	rbf	NaN	NaN	NaN	NaN
29	SVC	0.375	0.416579	0.48	0.0455636	10	0.0001	rbf	NaN	NaN	NaN	NaN
26	SVC	0.366197	0.401969	0.447552	0.0339301	1	0.001	rbf	NaN	NaN	NaN	NaN

With these training data, Table 4 shows that upsampling of the minority class gave an improved level of accuracy for the top tests, and assortment of each classifier was largely similar as with SMOTE-treated data. The random forest classifier performed best, with a top average score of 0.936 here.

The ROCs in Figure 12, however, show that success with training data does not necessarily equate to success with test data.

Figure 12. ROCs with classifiers developed through minority-upsampling.

ROCs give the impression that, on the test data, XGBoost may perform the best of these three models chosen here. However, precision-recall numbers reveal that the original problem of difficulty with correctly classifying the minority class (non-survivors) persists with this small testing dataset. With 6 samples here, there is little room for error, but in all cases the models

Table 5. Classifiers and accuracy, precision, and recall scores from upsampled training data.

	Accuracy	Class	Precision	Recall	F1-score	Support
Extra trees:						
Training	1.0					
Testing	0.904					
AU-ROC	0.6127					
		0	0.94	0.95	0.95	108
		1	0.00	0.00	0.00	6
		Avg/total	0.90	0.90	0.90	114
Random forest:						
Training	1.0					
Testing	0.921					
AU-ROC	0.7014					
		0	0.95	0.96	0.98	108
		1	0.20	0.17	0.18	6
		Avg/total	0.91	0.92	0.92	114
XGBoost:						
Training	1.0					
Testing	0.86					
AU-ROC	0.733					
		0	0.95	0.90	0.92	108
		1	0.08	0.17	0.11	6
		Avg/total	0.91	0.86	0.88	114

Low precision and recall with mortality

For this dataset it is not difficult to generate a model that can report high accuracy and even a high AUC for ROC. It is even possible with few optimizations to generate a model with a level of accuracy that is higher than the prevalence of survival is in the population; in this dataset about 85% of the patients survived the first year after surgery, so any model that reports greater accuracy than 85% should in theory perform better than random guessing would. However, it is clear from precision and recall values associated with the target class (1) that refers to patients who did not survive the first year post-surgery

that most of these models fail to address the goal of providing actionable information relating to a real risk of mortality.

A variety of types of classifiers have been analyzed as well as parameter and feature selection techniques, and at this time logistic regression applied to the minority-upsampled target training data and set for balancing of class weight with C=100, additionally, shows more balanced precision-recall statistics between the target classes (Table 6). For the data tested in the analysis presented in Table 6, 94/141 surviving patients were correctly assigned as such, while 47 were predicted to not survive. Among patients who did not survive, 10/18 were correctly predicted as such, while 8 were predicted to have survived. This is still not high resolution, but perhaps shows finer assignment than previously presented models in this report do.

The model of choice is still in development as of this writing for the milestone report. Various techniques of classifier choice and manual feature selection reveal trade-offs. Some models are biased toward classifying all or nearly all patients as survivors, while others will identify all or nearly all as not predicted to survive. This suggests there is a complex relationship between some features and patient outcomes, and these continue to be examined.

Table 6. Accuracy, precision, and recall for logistic regression applied to upsampled training data.

	Accuracy	Class	Precision	Recall	F1-score	Support
Logistic regression:						
Training	0.682					
Testing	0.654					
AU-ROC	0.6379					
		0	0.92	0.67	0.77	114
		1	0.18	0.56	0.27	18
		Avg/total	0.84	0.65	0.72	159

References

Batista, DS. “Hyperparameter optimization across multiple models in scikit-learn.” Web. Accessed: 6/11/18. http://www.davidsbatista.net/blog/2018/02/23/model_optimization/ .

Katsaroumpas, P. “Hyperparameter Grid Search across multiple models in scikit-learn.” Web. Accessed: 6/11/18. http://www.codiply.com/blog/hyperparameter-grid-search-across-multiple-models-in-scikit-learn/ .

University of California, Irvine, Machine Learning Repository. Heart Disease Data Set. Web. Accessed: 6/20/18. http://archive.ics.uci.edu/ml/datasets/Thoracic+Surgery+Data . Zieba, M, Tomczak, JM, Lubicz, M, & Swiatek J. 2014. Boosted SVM for extracting rules from imbalanced data in application to prediction of the post-operative life expectancy in the lung cancer patients. *Applied Soft Computing*. 14A:99-108. Principal investigator information for component datasets used in this study: Marek Lubicz (1), Konrad Pawelczyk (2), Adam Rzechonek (2), Jerzy Kolodziej (2)

-- (1) Wroclaw University of Technology, wybrzeze Wyspianskiego 27, 50-370, Wroclaw, Poland
-- (2) Wroclaw Medical University, wybrzeze L. Pasteura 1, 50-367 Wroclaw, Poland