Lung Cancer Survival Prediction with Synthetic Data

Henry Marie MONT, Jean-Côme CARISSAN

Supervisor: Markus Haug, Institute of Computer Science, University of Tartu

UNIVERSITY OF TARTU
1632

Introduction

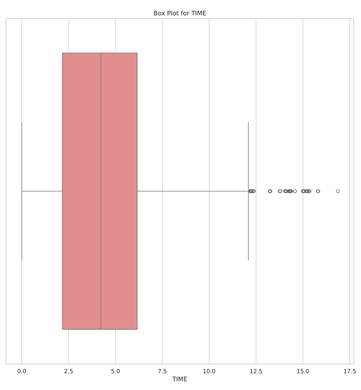
Cancer treatment is a complex and multifaceted endeavor involving an array of procedures, drugs, conditions, and examinations. The intricacy of the treatment process, combined with the diversity of medical interventions, poses a significant challenge in discerning the true efficacy of each element on a patient's survival time. As medical professionals navigate through a myriad of data points, from drug administrations to various diagnostic procedures, determining the pivotal factors influencing a patient's long-term survival becomes a daunting task.

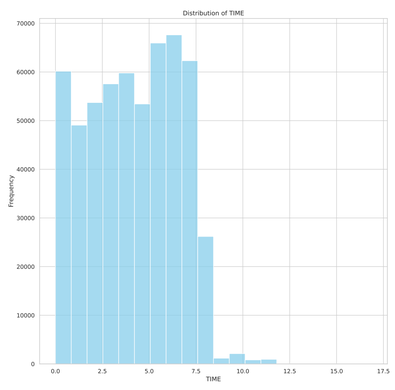
This project delves into the realm of lung cancer survival prediction, utilizing synthetic medical data to unravel the intricate web of treatments and their implications on a patient's journey.

The project is aimed at leveraging synthetic medical data to predict 5-year survival rates post-lung cancer diagnosis. The measure of success for our project is marked by achieving a significantly higher accuracy in predicting the 5-year survival rate. Currently, we are targeting an AUC-ROC value of 0.75 after feature selection as a benchmark.

Data exploration

The synthetic data-set at the heart of this project was provided by Marcus HAUG, it encapsulates the complexity of cancer treatment trajectories. Each entry represents a distinct medical event, from drug administrations to specific conditions and measurements, meticulously recorded over time. The "SUBJECT_ID" serves as a unique patient identifier, while "DEFINITION_ID" encapsulates a diverse array of medical interventions. The "TIME" column gives information about when the recorded event took place. Upon exploring data, we noticed that the data-set exhibited no missing values or duplicate entries, ensuring the integrity of our foundational data. A noteworthy observation emerged from the TIME distribution, revealing outliers, particularly those extending beyond 12.5 years. Furthermore, a significant challenge in the data-set lies in the presence of noise. Amidst the data, there are crucial definitions, such as "chemotherapy," coexisting with less relevant ones, like "eyesight check."





Data pre-processing

For data pre-processing, we prepared our data through the following steps. We addressed outliers using statistical methods to maintain data quality. We created a target feature, indicating whether a patient survived beyond 5 years post-lung cancer diagnosis. To clean the data and not create any bias for our model, we removed the rows corresponding to patients dying ('death' status).

To capture the temporal aspect, we binned the TIME feature to extract the year of each medical event, aligning with our 5-year survival prediction goal. Merging "DEFINITION_ID" and TIME, we formed a new feature to uniquely identify each event over time. After one-hot encoding "DEFINITION_ID" to represent medical procedures categorically, we grouped the data by "SUBJECT_ID", aggregating features with a logical OR operation.

This process resulted in a simplified and consolidated data-frame, with one row for each patient and features representing medical events for each year.

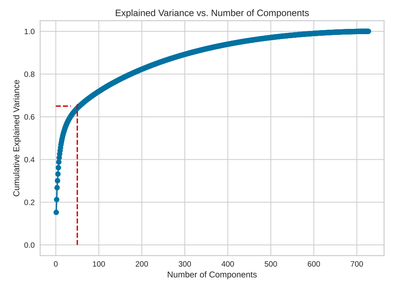
	SUBJECT_ID	DEFINITION_ID	TIME
0	1	drug_217	0.004807
1	1	condition_1922	0.008643
2	1	measurement_132	0.056765

SUBJECT_ID	DIED	DEF_condition_1000_0	DEF_condition_1000_1	...
1	0	False	False	...
2	0	False	False	...
3	0	False	False	...
4	0	False	False	...
6	0	False	False	...

Feature engineering

We explored various techniques to identify crucial features and ultimately settled on Principal Component Analysis (PCA) due to its efficiency and ability to capture the variance in our data-set. The PCA implementation involved standardizing the features, applying PCA, and then evaluating the explained variance ratio for each component. The cumulative explained variance was visualized, revealing an elbow in the curve, indicating that around 50 components could capture a significant portion of the variation in the data.

While we initially considered Stepwise Selection, a computationally intensive approach, we encountered challenges due to the large number of features, leading to performance issues on our laptop. The graph-based approach, which involved creating a graph with a node for each feature, also proved computationally intensive and was eventually discarded.

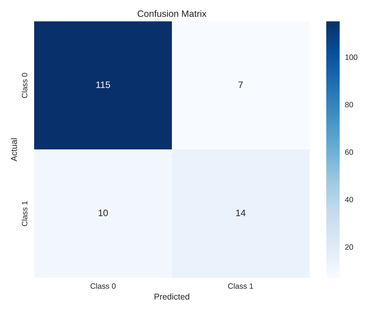


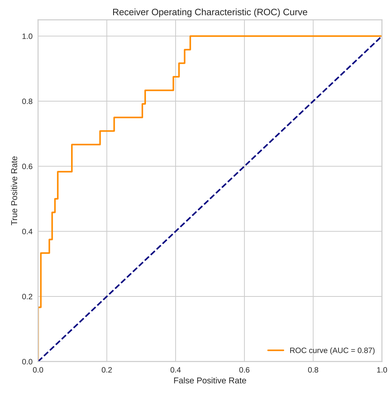
Model tuning

Considering we were dealing with a binary classification task, we evaluated the performance of several suited machine learning models, including Logistic Regression, RandomForestClassifier, XGBClassifier, GradientBoostingClassifier, SVC, BernoulliNB, and MLPClassifier. Our assessment, based on AUC-ROC scores, revealed Gradient Boosting as the standout performer with the highest score of 0.8682, showcasing its robust discriminatory power. XGBoost, Random Forest, SVM also demonstrated notable effectiveness with AUC-ROC scores around 0.85. Then, we refined our approach by tuning hyperparameter using grid search on the top four models: Gradient Boosting, XGBoost, Random Forest, and SVM. The optimized hyperparameters enhanced the performance of these models, as reflected in their AUC-ROC scores on the test set. To leverage the complementary strengths of these models, we crafted an ensemble using the VotingClassifier. This collaborative strategy resulted in a notably elevated AUC-ROC score of 0.8706 on the test set, surpassing the individual model performances.

Results

The confusion matrix and classification report depict a comprehensive overview of our model's performance. The model achieved an overall accuracy of 88%, effectively classifying instances into true positives (115) and true negatives (14). The confusion matrix reveals a notable precision of 92% for class 0, indicating a high proportion of correctly identified negative cases. However, precision for class 1 is slightly lower at 67%, implying a moderate level of false positives. The recall for class 1 is 58%, indicating that the model captures a substantial portion of actual positive cases. The weighted average f1-score stands at 0.88, signifying a well-balanced performance across precision and recall. These results collectively underscore the model's effectiveness in achieving a high overall accuracy while maintaining a reasonable balance between precision and recall for both classes.





References

- *scikit-learn Documentation*. scikit-learn. URL: https://scikit-learn.org/stable/documentation.html
- *seaborn Documentation*. seaborn. URL: https://seaborn.pydata.org/documentation.html