**Molecular Characterization of Pancreatic Cancer Survival using Machine Learning**

**Abstract ID: XXX**

**Abstract**

In 2021, pancreatic cancer was the fourth cause of cancer-related deaths as stated by the American Cancer Society (ACS) [1]. Pancreatic Ductal Adenocarcinoma (PDAC), which incidence rate of 3%, has been proven difficult to detect due to lack of symptoms during early stages. As a result, it inhibits quicker diagnosis and lowers the possibility of removing large scale tumors during surgery. Developing effective diagnostics would increase the survival of patients with pancreatic cancer, where gene expression can be used to profile genes as possible biomarkers. Pancreatic cancer research communities such as TCGA and ICGC have begun performing large cohort studies with mRNA expression and survival status to aid understanding of the biological mechanisms of this disease. Therefore, this study intends to extract a gene expression from both databases that can assess survival risk of PDAC patients using machine learning approaches. To implement predictive classification models to distinguish between living and deceased status, feature selection was required and executed using Python and R programming to only include the most important predictors. Recursive Feature Selection (RFE) was used to reduce the dimensionality of the dataset since as a wrapper method it has the advantage to measure the interdependencies between variables. Python acknowledged 100 predictors for TCGA using SupportVectorClassifier (Accuracy~74.6%) and 10 predictors for ICGC using RandomForestClassifier (Accuracy~82.2%). R programming was not able to increase the accuracy, however, its subsets were randomized while Python held bias towards the features at the end of the dataframe.

**Keywords**

Pancreatic Cancer, Survival Rate, mRNA Gene Expression, Feature Selection, Biomarkers

1. **Introduction**

The human pancreas is in charge of regulating blood sugar; it uses exocrine cells to produce digestive juices and endocrine cells to produce hormones. These cells can sometimes grow out of control, which forms malignant tumors which develop into cancer. Pancreatic cancer is a deadly disease in which these cancerous cells form in the tissues of the pancreas. According to the ACS, PDAC is the most common subtype of cancer, associated with exocrine cells which make up most of the pancreas. What troubles pancreatic cancer patients’ survival is its lack of diagnostics in early stages (given three percent of incidence) and limited treatments, only ten percent are expected to live after 5 years. This body tissue is located in an uncomfortable area to identify tumors, and most patients do not show symptoms until after the third stage. During these advanced stages, the cancer is often spread throughout the rest of the body in problematic areas near blood vessels [2] which is why surgery is only recommended to less than twenty percent of patients. Even so, surgeries to remove lethal large-scale tumors do not necessarily help with survival (ninety percent of cases are fatal). The statistics for pancreatic cancer position it as the fourth leading cause of cancer-related deaths, accounting for seven percent of all cancer-related deaths [1].

Research to improve cancer diagnostics involve studies on gene expression and their advantages to identify biomarkers of pancreatic cancer. Gene expression is the process by which cells build proteins based on DNA instructions carried through RNA, it controls how and when proteins are made [3]. It helps analyze genetic mutations and study cellular abnormality associated with tumor cells, characterization of gene expression has been used in other types of cancer research. As for pancreatic cancer, where the only approved biomarker is CA 19-9 [4], the search for other biomarkers is important to assess patients’ survival rates. To find possible biomarkers across large cohorts of genes, statistical data analytics can build classification models with the optimal genes associated with pancreatic cancer survival rate. Hence, the objective of this work is to implement machine learning approaches to identify possible biomarkers which assess survival risk of pancreatic cancer patients. This will be achieved by selecting important features using machine learning methods and conducting a comparison across TCGA and ICGC, to evaluate if similar findings provide information about putative validated biomarkers.

1. **Methodology**

To categorize a gene as a possible expression-based biomarker, information from two large cancer patient cohorts were incorporated. TCGA is a historical portal whose research has helped characterize around 33 types of cancer including PDAC. From this repository, survival status from 184 patient with their respective mRNA expression values for 20,531 gene probes were extracted and preprocessed. After pre-processing (e.g.,eliminate null values)a resulting dataset with 177 patient samples characterized across 20,025 gene probes were obtained. The ICGC is a voluntary scientific organization among the world's leading cancer and genomic researchers. ICGC provided samples of pancreatic cancer survival status and mRNA expression for 92 patients, and after preprocessing, resulted in a 90 patients by 35,601 gene probes dataset. A large number of predictors often lead to complex models with unreliable predictions, which is why TCGA and ICGC datasets needed to be reduced using computational methods. Their high dimensionality required feature selection to obtain the most relevant predictors to distinguish between living and deceased status from gene expression. Comparison between the features obtained for the TCGA and ICGC datasets would also be executed in both Python and R programming languages to evaluate the fit with classification models.

Feature Selection is a method in computer programming that takes a set of data and reduces the number of variables. Inside feature selection, the genes can be reduced based on a scoring method (also known as filters), training a subset of genes to a machine learning algorithm for evaluation (otherwise known as wrappers), and even a combination of both (called embedded methods). Each of these contains shared and unique qualities as they search the optimal features, where the wrapper methods proved to incorporate dependency between variables while adding genes to an initial subset and evaluating the fit. The wrapper technique used for this study was the RFE, which starts with all features in the training dataset and removes features by their rank of importance depending on the algorithms or statistical tests [5]. Although using this method can be advantageous to understand the biological interdependencies between genes, wrappers are often characterized by extensive computational times. In the case of RFE feature selection in Python language with ‘sklearn’ library, its effects on number of predictors for analysis limited its selection parameter for 3, 10 and 100 predictors for TCGA and ICGC datasets. On the other hand, R programming’s version of RFE evaluated up to 200 and 400 predictors with step 5 and chose the best subset for evaluation through ‘caret’ and ‘randomForest’ libraries.

An initial learner evaluation using cross-validation was applied in Python and R using 3 repeats and 5 folds to reduce overfitting, a recommended step when working with machine learning methods as well as the classifiers within the wrapper methods. Python computational run times were collected using a Mac 16GB RAM device with a 2.3 GHz processor while R codification was run in a Mac 8GB RAM device with a 1.1 GHz processor. This included the addition of data storage necessary to moderate prolonged run time for RFE and validate features between datasets while building classification models. To store the results from feature selection, the built-in functions would save the reduced matrix in a CSV file and information regarding run time and cross-validation scores for every computational run for further evaluation.

Once the feature selection step was completed, the predictive performance of the reduced set of important genes obtained from TCGA and ICGC were evaluated using cross validation of 10 folds in three classification models: Logistic Regression (LR), Support Vector Classifier (SVC), and Random Forest Classifier (RFC). LR predicts the probability of a target variable where the dependent variable is a binary response [6]. SVC adapts to the data in a hyperplane that ideally categorizes it. Lastly, RFC is a model consisting of a large number of decision trees to obtain a predictive performance [9]. Classification performance involved four metrics: accuracy represents the ratio of correct predictions for the test data, precision measures the number of predictors (true positives) the model correctly predicted divided by the total number of predictors, recall analyzes how many positive predictions were selected (true positive rate) and ROC AUC measures the probability of a model selecting a positive example over a negative [7]. Likewise, ROC AUC curves graphed the true positive rate versus the false positive rate, which assists in visualizing how much the models distinguish between a living and deceased patient [10]. Additional to the respective performance for each subset, predictors extracted from R were validated by evaluating TCGA’s optimal subset in ICGC’s complete dataset and vice versa to analyze the selected features’ relevance in another set of data using the same performance metrics.

[Enrichment analysis methodology]

1. **Results**

Tables 1 and 2 shows a summary of the predictive performance metrics from Python’s RFE function across different sets of important genes and classifiers for TCGA, and ICGC, respectively. The optimal prediction results are highlighted in the tables to display which number of features best predict survival status along with its computational time. Accordingly, Figure 1 plots the ROC AUC curve metric for each dataset.

Table 1. TCGA Classification Model Metrics with Python RFE using Random Forest Classifier

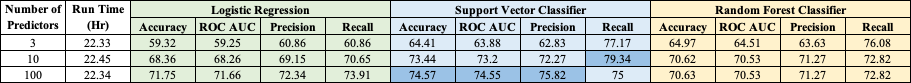
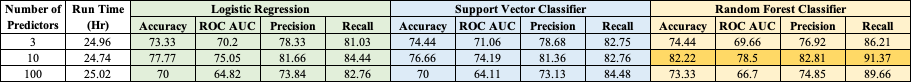


Table 2. ICGC Classification Model Metrics with Python’s RFE using Random Forest Classifier



A picture containing text, sky, line

Description automatically generatedA picture containing shape

Description automatically generated

Figure 1. ROC AUC Curves for 100 TCGA predictors modeled by SVC (left) and 10 ICGC predictors modeled by RFC (right)

Even though the ICGC was a smaller dataset in terms of the number of samples, its computational times were slightly higher than those from TCGA, TCGA took an average of 22.4 hours while ICGC 24.9 hours. Also, increasing the number of features had little to no effect on the run time, as shown in Table 1, the SVC for TCGA offered higher accuracy, precision, and ROC AUC score when using 100 genes than 10 genes except for recall which was higher using 10 features. Similarly, RFC presented optimal metrics when using 10 features, adding more features did not increase results (see Table 2). Fitting the features to a Logistic Regression model did not present optimal metrics for neither of the datasets

Python’s optimal features were compared in terms of their intersection, separating 10 and 100 optimal features from ICGC and TCGA, respectively. In terms of exact gene names, there was no overlap. However, when studying the families of genes, UCK, UNC, and WNT appeared on both datasets as relevant variables to predict survival status. WNT proteins’ role is to send signals from outer cells to inner cells; variations of UCK have been found in cancerous cells and considered a target in anti-cancer therapies, UNC families are used in experiments with worms to identify genetic defects. Furthermore, predictions achieved with Python’s RFE function held an apparent bias towards the features at the end of the list for both datasets, which were initially ordered alphabetically. R programming did not share this behaviour, it displayed random features selected by their importance in the model, hence its application of RFE methodology.

For functions executed in R’s RFE package, Table 3 presents predictive performance metrics in each classifier after finding the optimal features in TCGA datasets and their common features when evaluated with ICGC samples. On the other hand, Table 4 gathers performance metrics from ICGC along with its corresponding evaluation on TCGA samples. Highlighted sections in both tables indicate which classifier model offers the best metrics in terms of the subset selected by RFE.

Table 3. Classification Model Metrics in TCGA and Validation in R’s RFE using Random Forest Classifier

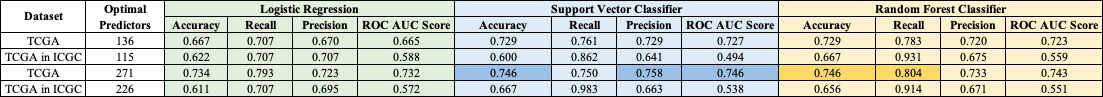


Table 4. Classification Model Metrics in ICGC and Validation in R’s RFE using Random Forest Classifier

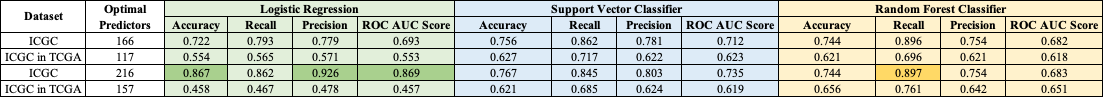


Table 3 proves 136 features extracted from 200 predictor size and 271 from 400 limit size for TCGA dataset. SVC model obtained the highest precision and ROC AUC score when 271 features were selected, yet recall was higher in RFC. Top accuracy was achieved both in SVC and RFC with 271 features also, more predictors increased classification metrics. Their respective validations with ICGC dataset are displayed below their separate optimal predictors, where 115 out of 136 predictors were modeled in ICGC patient samples and 226 out of 271 predictors, respectively. [Comment about validation]

ICGC on RFE considered 166 features when RFE function size went up to 200 predictors and later 216 when limited to 400 predictors, as described on Table 4. Out of the optimal 166 predictors, 117 were found in TCGA patient samples while 157 were discovered when 216 predictors were chosen. Similar to TCGA metrics, adding predictors presented higher overall metrics. LG displayed highest accuracy, precision and ROC AUC score with 216 features, RFC model provided higher recall. [Comment about validation metrics]

[Enrichment Results]

1. **Conclusions**

The ability of wrapper methods to incorporate dependency between variables can be relevant in studying potential biomarkers given computational time limitations since some genetic expressions require the presence of others to generate certain proteins [8]. The computer device used to generate the selected features took around a day (22.4 and 24.9 hours for TCGA and ICGC, respectively); however, the number of predictors did not affected the run time. Adding more features to the model does not necessarily increase the performance, proven when ICGC reached higher metrics with 10 variables rather than 100. In terms of accuracy, ICGC dataset scored 82.22% while TCGA scored 74.57%. Although the comparison did not find identical genes in common, UNC, UNC, and WNT families were present in both models and are known to be relevant in cancer-related studies such as therapies, genetic defects, and protein signals.

[R conclusions]

This opens the door to generate codes that calculate the optimal number of features by taking into account estimated running times. Applying other variations of the RFE method such as different learners aside from Random Forest Classifier is needed to possible improve the selection of markers. Though the results were highly predictive and encouraging, it also showed some possible bias on the selection of important features since most of them seem to be at the end of the gene list if order alphabetically which could be a problem in terms of biological interpretation of these putative biomarkers. Before conducting correlation analysis on the mRNA expressions, investigations on how Python libraries generate their results are needed, as well as, exploring other implementations where the R programming is a possibility.

**Acknowledgements**

I would like to thank the Puerto Rico Louis Stokes Alliance for Minority Participation (PRLSAMP) organization for their funding and support in this undergraduate research opportunity. I also wish to acknowledge the professor and students who previously worked in pancreatic cancer research for their mentorship in these upcoming findings.

**References**

[1] American Cancer Society. Key Statistics for Pancreatic Cancer. (2021, January 21)., from <https://www.cancer.org/cancer/pancreatic-cancer/about.html>

[2] Marks, Julie. Inoperable Pancreatic Cancer. (2020, October 18)., from <https://www.healthline.com/health/inoperable-pancreatic-cancer>

[3] [J Cancer.](https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6036716/) 2018; 9(13): 2249–2265. Published online 2018 Jun 5. doi: [10.7150/jca.24744](https://dx.doi.org/10.7150%2Fjca.24744)

[4] Hasan, Syed, et al. "Advances in pancreatic cancer biomarkers." Oncology reviews 13.1 (2019).

[5] Brownlee, Jason. Recursive Feature Elimination (RFE) for Feature Selection in Python. (2020, May 25). from <https://machinelearningmastery.com/rfe-feature-selection-in-python/>

[6] Machine Learning – Logistic Regression. Tutorials Point., from https://www.tutorialspoint.com/machine\_learning\_with\_python/machine\_learning\_with\_python\_classification\_algorithms\_logistic\_regression.htm

[7] Classification: ROC Curve and AUC., from https://developers.google.com/machine-learning/crash-course/classification/roc-and-auc

[8] Nature Education. Gene Expression. (2014)., from <https://www.nature.com/scitable/topicpage/gene-expression-14121669/>

[9] Yiu, Tony. Understanding Random Forest. (2019, June 12)., from <https://towardsdatascience.com/understanding-random-forest-58381e0602d2>

[10] Narkhede, Sarang. Understand AUC – ROC Curve. (2018, June 26).; from https://towardsdatascience.com/understanding-auc-roc-curve-68b2303cc9c5