Survival Analysis of Intrahepatic Cholangiocarcinoma with mRNA Sequencing and Machine Learning

Katy Scott | 15kls3 | 20009328

Supervisor: Dr. Amber Simpson and Dr. Randy Ellis

Queen’s University School of Computing

PATH828 – Fall 2020

Dr. Kathrin Tyryshkin

December 21st, 2020

Abstract

Intrahepatic cholangiocarcinoma is difficult to diagnose and has increasing levels of prevalence and mortality. Identifying genes with differential expression could be beneficial for prognosis prediction. mRNA Sequencing data from The Cancer Genome Atlas (TCGA) Program were run through feature selection, unsupervised and supervised learning methods, as well as statistical analysis, to investigate possible predictive features. A model built using five features chosen by sequential feature selection achieved a prediction accuracy of 76.9 percent. None of these features showed a significant difference in survival prediction based on expression levels. One feature was shown to be significantly differentially expressed in men.

1 Introduction

Cholangiocarcinoma is a cancer of the bile ducts that connect the liver, gallbladder, and small intestine. Subclassifications have been made based on the origin of the cancer; intrahepatic refers to those that start in the smaller branches within the liver, while perihilar and distal originate at the entrance to the liver and further down the duct closer to the small intestine, respectively. Although rare, there has been an increase in both the incidence and mortality of cholangiocarcinoma worldwide in the past few decades [2]. According to the National Cancer Institute, liver and intrahepatic bile duct cancer have a 5-year survival rate of 19.6 percent and are estimated to be responsible for 5 percent of all cancer related deaths in 2020 [7]. However, the survival rate drops to just 8 percent when only intrahepatic bile duct cancers are considered [15].

Intrahepatic cholangiocarcinoma (ICC) is known to be architecturally heterogenous in nature, likely due to the variation in cell function and morphology along the bile duct [14]. This leads to atypical clinical presentation and imaging, ultimately resulting in diagnosis coming at advanced stages of the disease [9]. Further, ICC is not curable, with viable treatment options, including resection, showing minimal or no improvement to survival chances [4]. Lastly, ICC has been shown to have greater incidence rates in men than in women in the United States [13].

The consistent rise in popularity and affordability of NextGen sequencing has led researchers to search for genetic markers to aid in earlier diagnosis of cholangiocarcinoma. In this project, a similar path has been followed to attempt to predict patient survival by analyzing genetic characteristics using unsupervised and supervised learning, as well as statistical analysis.

It was hypothesized that that mRNA expression levels of select genes will differ between ICC patients with a time to event prior to and later than two years past the diagnosis. In this study, the event in question is death.

2 Methods

The following section details each of the steps in the project pipeline. Data was retrospectively acquired from available online data sources. Since the goal was to predict survival, data censoring was completed for patients without a known time to event. The planned supervised learning method required a reduction in the number of genes to be analyzed in the dataset. To accomplish this, a combination of filtering and feature selection was used, including selecting out genes known to be related to ICC prognosis prediction. Once the data had been preprocessed, predictive models were built with the reduced feature sets using supervised learning to classify whether a patient survived or had a follow-up past two years from the initial diagnosis. The feature set with the best predictive results was further studied with statistical analysis, including Kaplan-Meier survival curves and the Mann-Whitney *U* test to explore if expression levels differed between men and women.

2.1 Data Collection

The original dataset intended for use was from a larger research project that would incorporate the analysis pipeline developed here. However, the data was not made available in time for this work. Instead, the process was designed using a similar dataset with the intention of it being easily applied to the other data once it was made available.

The cholangiocarcinoma (CHOL) cohort dataset in use in this project was retrospectively acquired from the Broad Institute TCGA Genome Data Analysis Center (GDAC) [ref Broad]. This cohort included data from 45 patients with intrahepatic bile duct carcinoma, malignant neoplasm of the extrahepatic bile duct, or liver cell carcinoma. This work is primarily interested in ICC, so only those patients were selected to be studied, leaving 39 patients for analysis.

The retrieved data contained 133 clinical data elements (CDEs) and mRNA sequencing (mRNA-Seq) data for 20,531 genes. Multiple versions of the mRNA-Seq data were available from TCGA, including raw gene count data, data that had been normalized, and data that had been normalized and log-transformed. The sequencing data selected had been aligned to the hg19 reference genome, was processed using the RNA-Seq by Expectation Maximization pipeline [12] and had been normalized. Both the clinical and mRNA-Seq datasets were imported into MATLAB (MathWorks, Natick, MA, USA) and processed using custom software.

Once loaded in, some initial observations about the data were recorded, including the number of patients who had a known time to event, the number of men and women, etc. A summary of these observations can be seen in Table 1.

2.2 Preprocessing

2.2.1 Normalization

As previously mentioned, mRNA sequencing data was preprocessed using the RNA-Seq by Expectation Maximation (RSEM) software package by the Broad Institute TCGA GDAC. According to the Broad Institute, this method is considered by many to be a better method than Reads Per Kilobase per Million (RPKM) for estimating expression levels from mRNA-Seq data. This preference is likely based in RSEM output being more comparable across samples and species as it is independent of the mean expressed transcript length [12]. One other notable difference in this technique is that it does not require a reference genome as many other existing tools do.

RSEM outputs gene abundance estimates in two measures: the estimated number of fragments that are derived from a given gene and the estimated fraction of transcripts made up by a given gene. The latter can be multiplied by a factor of 10^6 to be transformed to generate a value in terms of Transcripts Per Million (TPM). In the TCGA data, a normalized gene count was generated by taking the former estimate, referred to it as the “raw gene count”, and applying the following equation:

$$normalized\_count = \frac{raw\ gene\ count}{75\ percentile\ of\ sample\ estimate} * 1000$$

No explanation was found for the choice of 75th percentile and 1000 multiplication factor.

2.2.2 Visualization

This normalized count was visualized to ensure it was loaded into MATLAB properly and to check for any errors or problems with the data. Three plots were generated. The first was a scatter plot showing the total gene counts for each sample to ensure that the normalization had been successful. The next two were a box plot and a histogram of the expression level to check on the distribution of the data.

From the distribution plots it was determined that further transformation was required. The data was log-transformed and re-visualized.

2.2.3 Quality Control

Once the normal distribution of the data had been confirmed, additional visualizations were generated to check for outliers and batch effects. These included scatter plots to display and compare the interquartile range (IQR) for each sample and the Spearman correlation between the samples. The Spearman correlation was chosen as it is non-parametric and no assumption regarding the linear relationship between the features needed to be confirmed. These new plots, and those generated in section 2.2.2, were reviewed for batch effects and outliers.

Detected outliers were studied to determine if they were more likely technical errors or potentially biologically significant. If outliers were retained, they were marked for identification in any later analyses.

2.2.4 Filtering

The provided dataset included gene counts for 20,531 genes. Given that one of the intended uses of this data was supervised learning, this number had to be greatly reduced to prevent overfitting. Additionally, a large number of lowly expressed genes were noted in the initial visualization stage. These were diminished after the filtering process.

As a preliminary attempt at reducing the number of features, genes with low expression levels were removed. This was accomplished by computing a threshold based on a given quantile, Q, and removing any features that did not have a sample with an expression level above that threshold. Increasing levels of Q were used to achieve the greatest reduction in the feature count.

2.3 Unsupervised Learning

The goal of using unsupervised learning in this project was to see if any initial clusters could be observed. Ideally, these clusters would showcase a pattern distinguishing patients with a time to event before and after two years post diagnosis.

2.3.1 Hierarchical Clustering with Heatmap

The first method applied to the dataset was hierarchical clustering. This was performed using the clustergram function in MATLAB, which produced a combination heat map and dendrogram for both the samples and features.

This function requires that the data be median centered, so the following equation was applied:

$$median\_centered\_data = data - median(median(data))$$

In this formula, “data” is a n_features x n_samples matrix. The stacked calls of the median function find the median across the sample medians.

To generate labels for the columns (samples), the “days_to_death” and “days_to_last_followup” were extracted for the ICC patients from the clinical dataset. These values were converted to “years_to_death” and “years_to_followup” by dividing by 365 and flooring the result. These new variables were used to create a label vector containing the values described as follows:

- 0 = “years_to_death” less than or equal to 2
- 1 = “years_to_death” greater than 2
- 2 = “years_to_followup” less than or equal to 2
- 3 = “years_to_followup” greater than 2

The clustergram function provides many distance metrics to choose from for generating the dendrogram. Multiple options were explored for these metrics for comparison between the heatmaps and dendrograms. For further comparison, clustergrams were generated using the full feature set and the filtered feature set.

2.3.2 K-means Clustering

K-means clustering was the second method applied to explore the feature set. This was performed using the built-in k-means function in MATLAB. To investigate the optimal choice for k, the values ranging from 1 to 15 were tested. The median centered data was passed in such that the samples were clustered based on the expression levels of the genes. Since the dimensionality of the data was too high to properly be visualized using the typical two- or three-dimensional scatterplot, silhouette plots were generated for each k value, as well as an elbow plot for the average distance from each point to the centroid of its cluster. The elbow and silhouette plots could then be used to determine the optimal k-value for the data.

2.4 Supervised Learning

2.4.1 Feature Selection

Before any supervised learning could be applied to the data, further dimensionality reduction was required. This was done to reduce the possibility of overfitting in the models caused by including too many features and generating a complex high-dimensional function.

A variety of options were considered and attempted to reduce the number of features. Principal Component Analysis (PCA) and t-distributed stochastic neighbour embedding (t-SNE) were explored to transform the data into a lower dimension. However, both of these options resulted in a compression of the features, which hindered the identification of specific genes that could be used for assisting prognosis development.

This led to exploring different methods of feature selection. The most obvious avenue for selecting features was to utilize prior knowledge of genes shown to be associated with ICC prognosis. These were selected from various sources in literature [1][3][17]. Notably, when selecting these genes out from the filtered dataset, only a few had expression levels greater than the set threshold. To extract enough genes for model development, the genes were instead selected from the original unfiltered dataset.

Two additional methods were performed to reduce the dimension size: sequential feature selection using neighbourhood component analysis for classification (FSCNCA) and feature selection using the minimum redundancy maximum relevance algorithm (FSCMRMR). FSCNCA learns feature weights by using a diagonal adaptation of neighbourhood component analysis with regularization, based on work by Yang et al. [18]. This algorithm is functionally the same as k-nearest neighbours and was chosen based on recognition and its successful use in previous work with similar data. This function was used in conjunction with the built-in sequential feature selection function in MATLAB, sequentialfs, with 5-fold cross-validation. As there is an element of randomness to the function, both the cross-validation partitioning and the feature selection were run 25 times. The selected features from each run were stored and the number of times a feature was chosen was tallied. The initial threshold for the number of times a feature had to be selected was the midpoint between the minimum and maximum selection count. However, this ended up being too strict a threshold as it chose only one feature. Instead, the threshold was iteratively lowered until a reasonable number of features were left.

FSCMRMR was chosen as an alternative option when looking for another function for sequentialfs. Rather than returning a model for classification, the FSCMRMR function ranks features for classification [10]. This ranking is based in finding the optimal set of features that is minimally redundant and maximally relevant to the response variable. Both redundancy and relevance are quantified using mutual information between the features and between the features and the response variable, respectively. The algorithm chose the same feature set on each run, so it was only run once. Along with the ranking, predictor scores were also provided for each of the features and these were plotted to visually assess how many features could be selected.

For both FSCNCA and FSCMRMR, a response variable was required. In this work, the response variable in question was survival past two years. For exploratory purposes, this was constructed in two ways.

The first was as a binary vector marking samples with 1 if the “years_to_death” or “years_to_followup” value was greater than two and was otherwise set to 0. The second was the label vector with four classifications described in 2.3.1 that was used for hierarchical clustering.

This generated five feature sets to be used for predictive model construction.

2.4.2 Classification Learner App

To generate the predictive models, the Classification Leaner App (CLA) in MATLAB was used. Each of the feature sets were concatenated with their corresponding response variable, either the 2-class survival past two years or 4-class death or follow-up past two years. The feature set selected from literature was duplicated and concatenated with each of the response variables.

Each feature set was loaded into the CLA and set up with 5-fold cross-validation. To start with, all the available classifier types were trained to get an idea of what the best model would be to fine tune. Models with the top accuracy and area under the receiving operator characteristic curve (AUC) were noted for further investigation. The classifier types with the best scores were run again using the discriminant classifier that optimizes the hyperparameters for that type. If the optimizable run showed improvement, the number of iterations was increased to see if further improvement was possible.

This process was repeated with Principal Components Analysis enabled to transform the features and remove the redundant dimensions. The accuracy and AUC were recorded for comparison and a similar tuning process was followed as describe in the previous paragraph.

2.5 Statistical Analysis

The feature set selected by the FSCNCA with two classes was chosen to be further analyzed as it performed the best in the supervised learning section. The goal was to investigate if the selected genes held any statistical significance for survival using the Cox regression analysis. This was done using the SPSS Statistics software platform (IBM, Armonk, NY, USA).

To begin with, the normalized gene counts for the feature set was concatenated with a selected subset of CDEs in MATLAB, saved as an Excel spreadsheet, and loaded into SPSS. A censored variable for labelling patients without a known time to event was generated. Some initial descriptive statistics were generated to ensure the data was properly transferred. The gene counts showed a non-normal distribution and were log-transformed to address this. The descriptive statistics were generated again, confirming that this transformation had handled the issue.

2.5.1 Survival Analysis

Survival analysis was chosen to investigate if the expression levels of the selected genes held any significance in predicting 2-year survival.

The Cox regression analysis was the first choice as it is considered a multivariable survival analysis and could be used to analyze all of the features at once. The Cox regression has a number of assumptions that must be met before it can be applied. These include:

1. Censoring must occur randomly and independent of the outcome.
2. The relative hazard over time is constant (proportional).
3. The logarithm of the relative hazard changes linearly with the weighted sum of the independent variables.

To test for the censoring assumption, new variables were generated for each of the genes based on if the expression levels were above the median. The median threshold was chosen as it was suggested for use in class and in a previous assignment. This new binary variable was plotted against the days to follow up or death variable and visually evaluated.

To test for the proportionality assumption, a Kaplan Meier curve was generated for each gene. Kaplan Meier has the same censoring assumption as the Cox regression, so it was safe to use this test. It was at this point that the selected genes failed the assumption check.

Since the Cox regression was not possible for this gene set, and the Kaplan Meier survival curves had been generated already, the decision was made to evaluate these instead. To compare between the survival curves for the different genes, the log rank test was performed as the same censoring assumption was met.

2.5.2 Mean Comparison

The difference in incidence rate of ICC in men and women led to the analysis of gender’s effect on the FSCNCA selected features. To evaluate if there was a difference in expression levels between the groups, the T test for independent samples was chosen to be applied. This test has the following assumptions

1. The groups are independent of each other and a sample may only appear in one group.
2. The dependent variable is normally distributed within each population.
3. The two groups come from two populations whose variances are approximately the same.

The first assumption wass met since the two groups being compared are men and women, and each of the samples exists in only one of those groups. To check for the normal distribution, the normality tests provided in SPSS were run. One of the features was not normally distributed in both populations, so the T test could not be used.

To ensure the same analysis was applied across all the features, the Mann-Whitney U test was applied since it does not require normality and is also only slightly less powerful than the T test. The only additional steps that had to be taken was to generate a numeric representation for the gender variable.

3 Results

3.1 Data Collection

The following table presents some general information about the CHOL cohort from TCGA in use in this work.

Table 1. *Clinical characteristics of TCGA ICC patient set, where the time to event is death.*

Characteristic	**All (n=39)**	**Known time to event (n=18)**	**Unknown time to event (n=21)**
Age at diagnosis, years (median, range)	66 (29-82)	65 (31-81)	66 (29-82)
Gender			
Female	22	11	11
Male	17	7	10
Days to follow up or death (median, range)	741 (10-1976)	584 (28-1939)	1077 (10-1976)

3.2 Preprocessing

3.2.1 Visualization

Figure 1 and 2 were produced following the methodology described in section 2.2.2. The plots shown in Figure 1 were generated with the pre-normalized data before any further transformations had been applied. Some outliers are visible in 1A, while B and C show very little interpretable information.

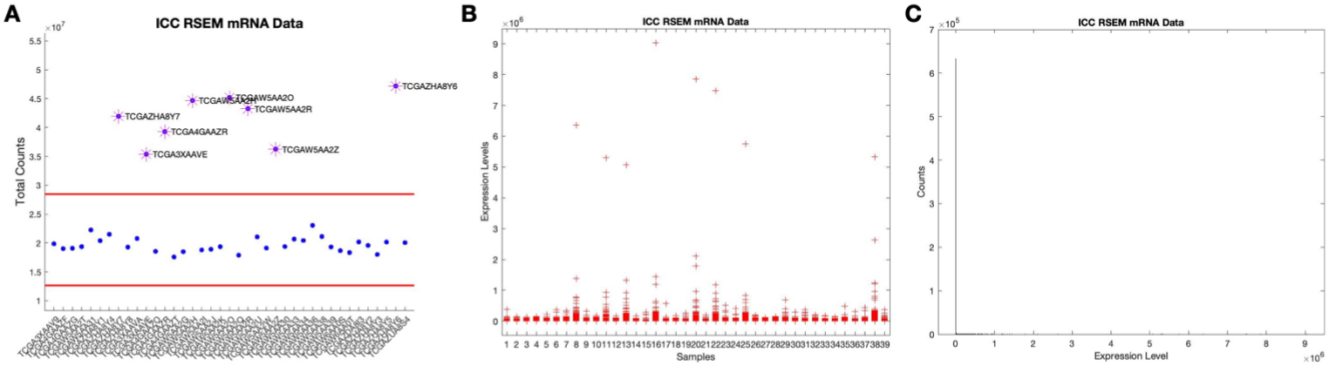

Figure 1. *TCGA ICC mRNA-Seq expression data that had been normalized as described in section 2.2.1. (A) shows a scatter plot of the total mRNA count for each sample, with the red lines as the outlier boundaries calculated based on the 25th and 75th quantile +/- α*IQR [16], and outliers marked in purple and labelled. (B) is a box plot of the expression data, and (C) is a histogram of the expression data.*

Figure 2 shows the same distribution plots as Figure 1B and C, but with the log-transformation applied to the data. The box plot in 2A shows little variance in the median and interquartile range for the data. It should be noted that the upper end of the boxes are aligned due to the normalization technique's use of the 75th percentile, shown in the equation in section 2.2.1. The histogram in 2B shows that the data are negatively skewed with an outlier for low expression levels. These low counts are addressed in the filtering stage describe in section 2.2.4.

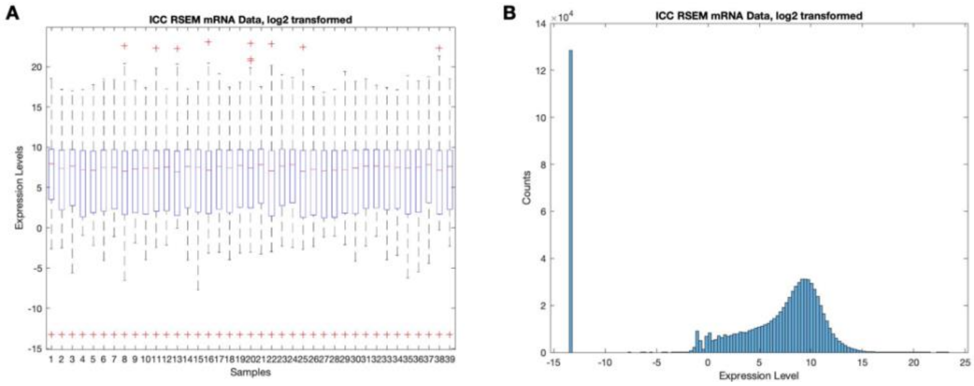

__Figure 2.__ TCGA ICC mRNA-Seq expression data after a log-transform had been applied. (A) is a box plot of the data and (B) is a histogram of the data.

3.2.2 Quality Control

To assess if any batch effects or outliers were present, the plots shown in Figure 3 were created supplementary to the boxplot in Figure 2A and the total counts scatter plot in Figure 1A. In reviewing the boxplot from 2A, no batch effects were visible, so no further processing was needed. Figure 3A shows no outliers for the interquartile range, and 3B shows a single outlier in the Spearman correlation. The correlation value is around 82%, so it was unlikely to be a technical variant and could hold biological significance and was not removed.

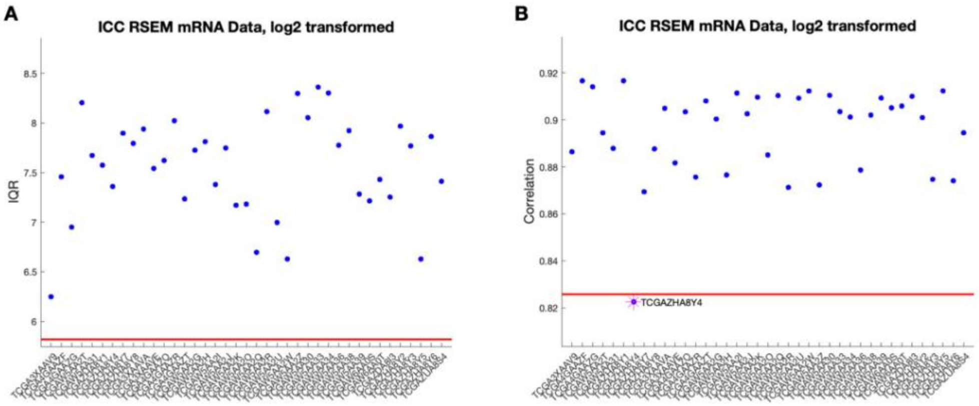

__Figure 3.__ Scatter plot of (A) the interquartile range and (B) Spearman correlation of the log-transformed expression data.

The outliers in the total counts plot were more thoroughly investigated, as the expression levels of these samples ranged from 1.75 to 2.5 times greater than the expression levels seen in the majority of the samples. Of the eight outliers present, two had a known time to event and neither event occurred later than two years after diagnosis. The outliers showed no clear trends upon examination but were marked for identification in any further analyses.

3.2.3 Filtering

The methods in section 2.2.4 produced the results shown in Table 2, which lists the quantile value that was used to set the threshold and the corresponding number of genes with expression levels that passed that threshold. A Q value of 0.99 was chosen for the final threshold and the filtered genes were used for the remaining methods unless otherwise specified.

Table 2. *Remaining feature count at different levels of filtering. Q refers to the quantile value used to generate the threshold.*

Filtering Threshold (Q)	Feature Count
0	20,531
0.99	6387
0.95	3929
0.98	1856
0.99	1041

As mentioned previously, the filtering also aimed to remove an outlier in the distribution histogram shown in Figure 2B. Figure 4 shows the distribution after filtering with a threshold value of 0.99, with the rightmost lowly expressed counts bar reduced greatly.

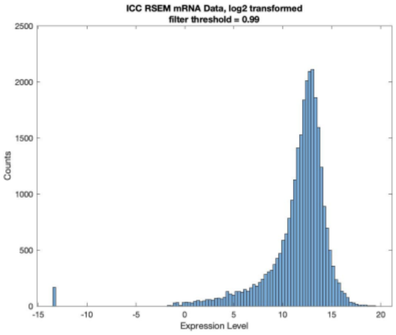

Figure 4. *Histogram showing the distribution of the data after filtering.*

3.3 Unsupervised Learning

3.3.1 Hierarchical Clustering with Heatmap

Figure 5 in this section was generated following the methods described in section 2.3.1. The clustergrams are two of several that were generated in the process. These were chosen as they best displayed the difference in clustering following the filtering process. In both figures, the red rectangle highlights the cluster of samples that are made up of the total counts outliers identified in section 2.2.3.

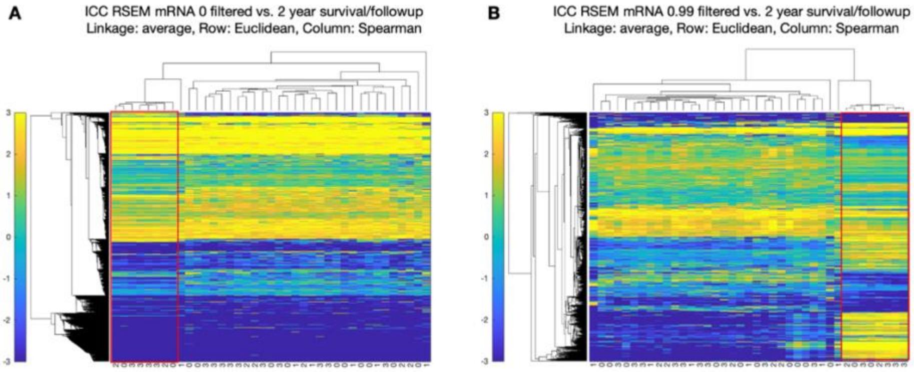

Figure 5. *Output from the MATLAB clustergram function using the average linkage, Euclidean distance metric for the feature clustering, and the Spearman distance metric for the sample clustering. Values along the x axis of the heatmap correspond to less than 2 years to death (0), greater than 2 years to death (1), less than 2 years to follow-up (2), and greater than 2 years to follow-up (3). (A) shows the clustergram generated when using all 20,531 genes. (B) shows the*

clustergram generated when using the remaining genes after filtering with a Q=0.99 (1041 genes). The red rectangle is drawn around the cluster of the total counts outlier samples.

3.3.2 K-means Clustering

As described in section 2.3.2, the usual scatter plot visualizations for k-means clustering were not producible as the dimensions of the data were too large (39 samples by 1041 features). Instead, the plots displayed in Figure 6 were created to investigate what an optimal k-value would be. To narrow the search range, the elbow plot in 6A was generated by plotting the average distance to the centroid for all the points at each k-value. Two possible inflection points at k = 2 and k = 3 were identified and the corresponding silhouette plots in 6B and C were visualized for examination. Since clustering results can vary between runs, the k-means function was run multiple times to subsequently produce multiple silhouette plots. A representative plot for each k value was selected to be shown here. From these, it became clear that k = 2 was the optimal choice.

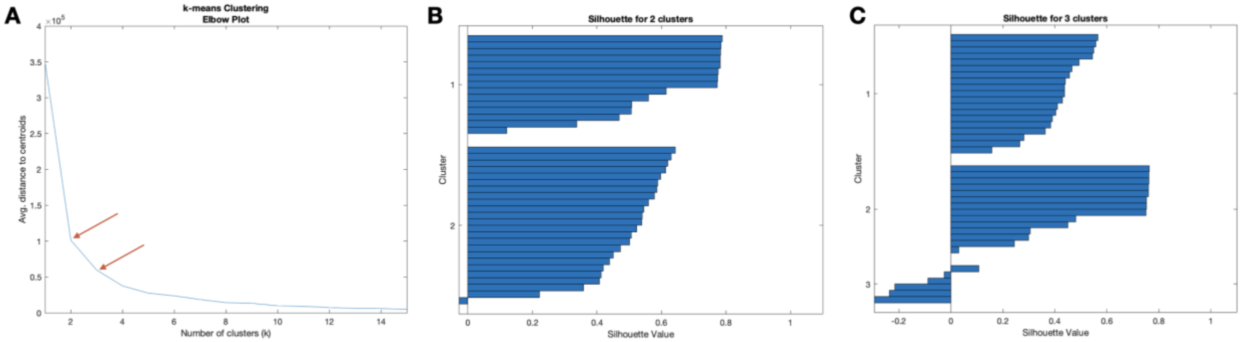

Figure 6. *Outputs from k-means clustering. (A) shows the elbow plot generated when running the algorithm with k-values ranging from 1 to 15. The red arrows denote the two potential inflection points, k = 2 and k = 3, that were investigated. (B) and (C) show one exemplary silhouette plot for k = 2 and k = 3, respectively.*

3.4 Supervised Learning

3.4.1 Feature Selection

This section summarizes the results of the process described in 2.4.1. Table 3 contains the final number of features selected using each method. For FSCNCA, the selected features had been chosen in at least three of the 25 iterations. This threshold was chosen as any higher selected only two features and lower included too many.

For FSCMRMR, the number of selected features was determined by the bar graphs shown in Figure 7, which displays the predictor scores for the top 25 ranked genes. For better comparison with FSCNCA, the number of genes selected was similar. For two classes, this meant using all the genes that had any predictive power, and for four classes the top ten were selected. The top ten also all had predictor scores above 0.1.

Table 3. *Number of features selected for each method and each number of classes considered.*

Method	# classes in response variable	# features selected
From literature	NA	25
FSCNCA	2	5
	4	10
FSCMRMR	2	6
	4	10

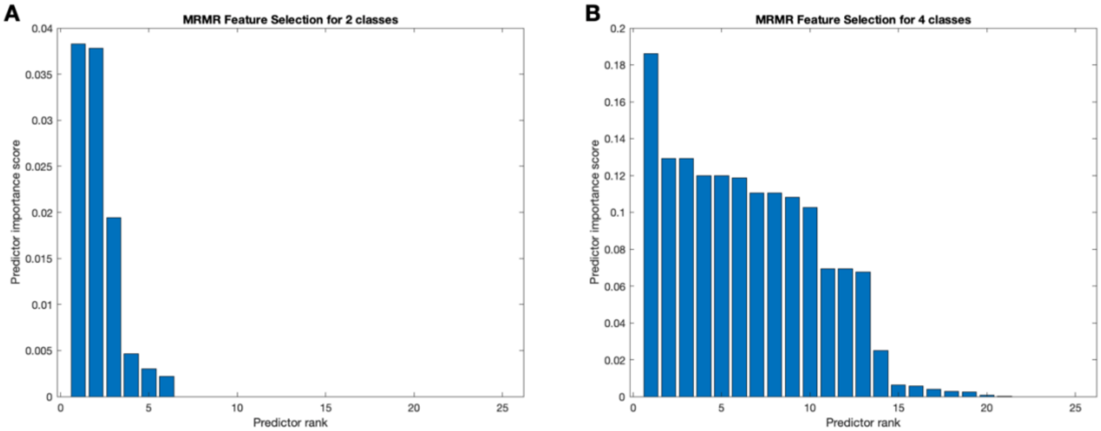

Figure 7. *Bar graphs showing the predictor score results for the top 25 ranked genes by FSCMRMR. (A) shows the genes selected when using the two classes. (B) shows the genes selected when using the four classes.*

3.4.2 Classification Learner App

Following the process described in section 2.4.2 generated the results shown in Table 4 and Figure 8.

Table 4. *Best model for each feature set and number of classes, with and without PCA applied. AUC is area under the receiving operator characteristic curve.*

Feature Set	Model Type	Accuracy	AUC
From literature			
2 classes	Ensemble RUSBoosted Tree	56.4	0.65
4 classes	Optimizable KNN	46.2	0.75
with PCA			
2 classes	Medium KNN	56.4	0.45
4 classes	Quadratic SVM	43.6	0.58
FSCNCA			
2 classes	Ensemble Bagged Trees	76.9	0.80
4 classes	Optimizable KNN	61.5	0.72
with PCA			
2 classes	Fine Tree	74.4	0.71
4 classes	Linear SVM	48.7	0.80
FSCMRMR			
2 classes	Optimizable SVM	66.7	0.72
4 classes	Optimizable Ensemble	61.5	0.86
with PCA			
2 classes	Optimizable Tree	76.9	0.73
4 classes	Optimizable SVM	38.5	0.40

Figure 8 shows the receiving operator characteristic (ROC) and confusion matrix produced by the best classification model: the Ensemble Bagged Trees built on the FSCNCA feature set with two classes.

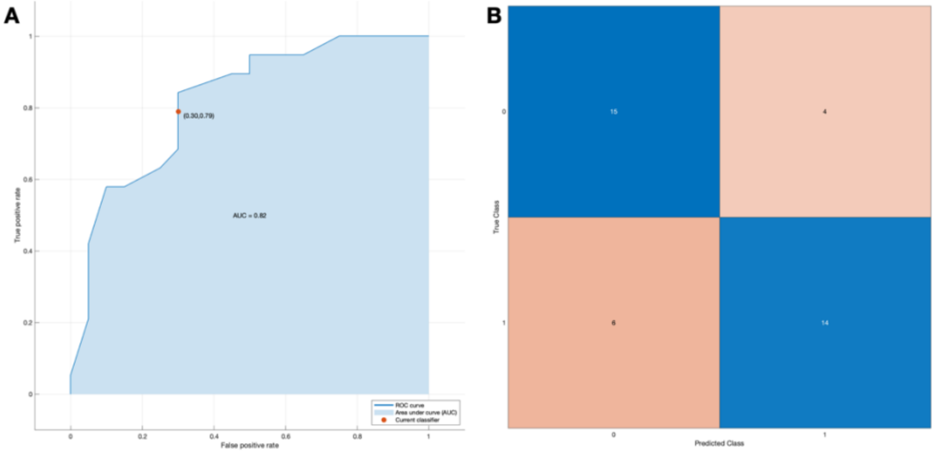

Figure 8. *Receiving operator characteristic curve and confusion matrix for the Ensemble Bagged Trees model built with the FSCNCA 2-class feature set.*

3.5 Statistical Analysis

The results in this section were produced using the log-transformed genes from the FSCNCA feature selection set.

3.5.1 Survival Analysis

The following results were obtained when the methods described in section 2.5.1 were applied. Figure 9 displays an example visualization of the censoring check performed for each of the selected features. The y-axis in this figure is the binary variable generated to represent high and low expression of the given gene based on the median threshold. From this graph, and those like it for each of the five genes, it was determined that the censoring assumption was met.

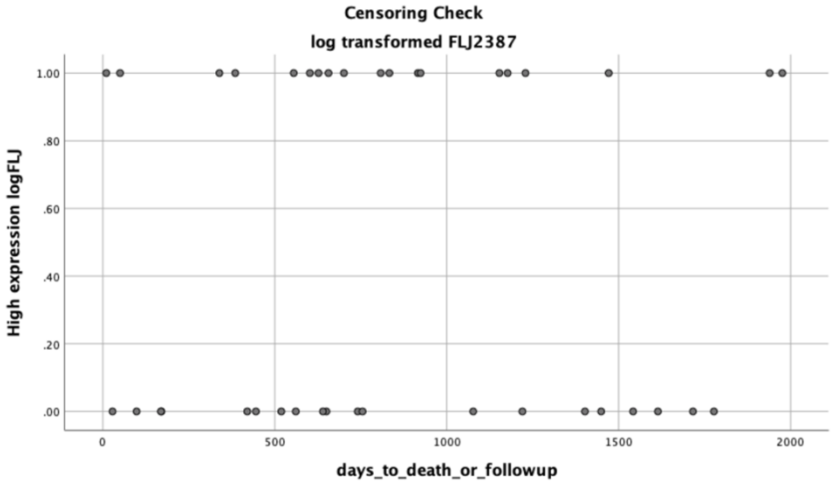

Figure 9. *Example of a censoring check plot that was created for each of the genes of interest.*

The Kaplan Meier curves generated for the proportionality assumption check are shown in Figure 10. Each of the graphs show the curves crossing, which determined that the Cox regression analysis could not be applied to this data. However, since it was going to be used as a multivariable survival analysis, the Kaplan Meier survival curves for the different expression levels were compared using the log rank test instead. The required censoring assumption had already been met, as shown in Figure 8, and the samples are independent since no expression level can be both low and high.

The results from the log rank test can be seen in Table 5. There was not enough evidence at the 5% level of significance to conclude that the expression level of any of the five genes results in differing survival distributions (p = 0.251, 0.546, 0.129, 0.710, 0.779)

Table 5. *Log Rank (Mantel-Cox) results for each of the genes of interest.*

Variable	Chi-Square	df	p
AQP3	1.317	1	.251
FLJ23867	.365	1	.546
MGLL	2.304	1	.129
PYGB	0.138	1	.710
RAP1GAP	0.079	1	.779

3.5.2 Mean comparison

MGLL expression was not normally distributed in the male population. This was validated using histograms, Q-Q plots, skewness, and the Shapiro-Wilk test of normality (p = .011). These results are summarized in Table 6.

Table 6. *Shapiro-Wilk test of normality results for each gene of interest in each population.*

Variable	Gender	Statistic	df	p
AQP3	female	.941	22	.205
	male	.974	17	.884
FLJ23867	female	.948	22	.293
	male	.924	17	.171
MGLL	female	.929	22	.118
	male	.850	17	.011
PYGB	female	.914	22	.057
	male	.928	17	.198
RAP1GAP	female	.938	22	.183
	male	.937	17	.284

Because of this, Mann-Whitney *U* tests were performed to compare the genders. The 22 female patients have higher mean ranks (23.36) than the 17 males (15.65) for RAP1GAP gene expression, *U* = 113, *p* = .036, *r* = -0.3, which was a statistically significant difference and, according to Cohen, is a medium effect size [8] None of the remaining genes showed a statistically significant difference in expression between male and female patients. The mean ranks, *U, p,* and *r* values for these are summarized in Table 7.

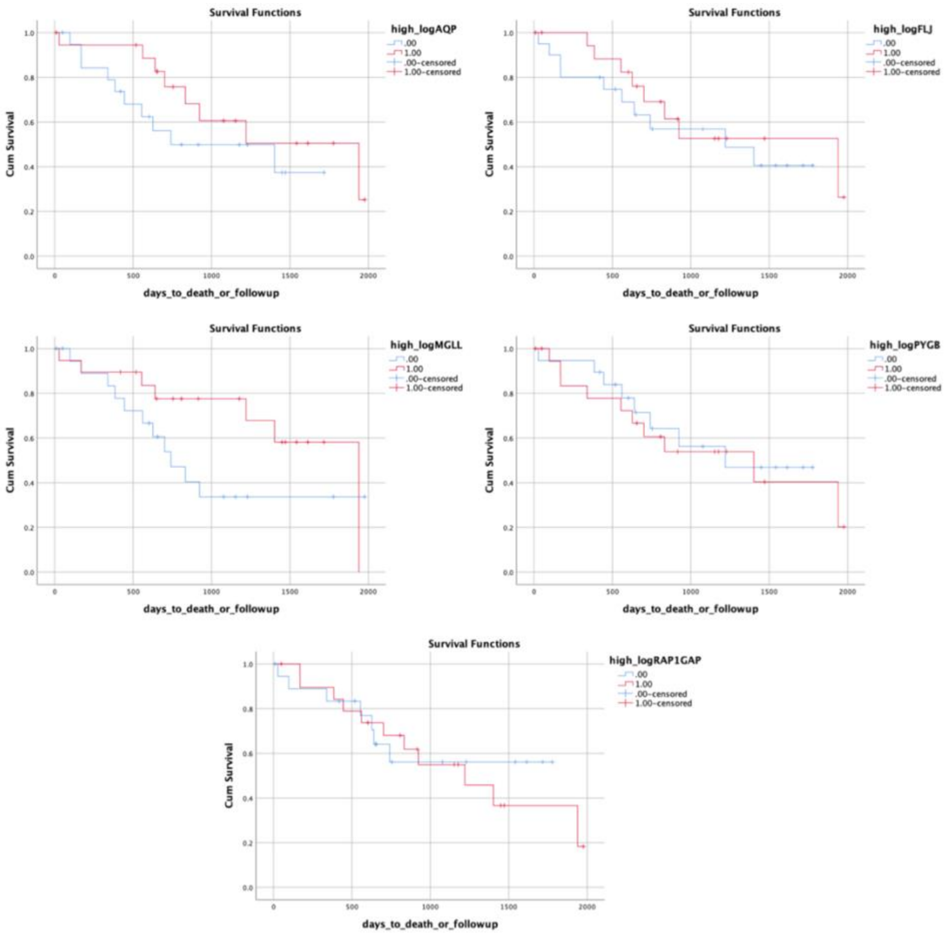

Figure 10. *Kaplan Meier curves for each of the genes of interest. The pink lines correspond to expression levels above the median and the blue lines correspond to expression levels below the median. Censoring occurred at the last follow-up for patients with no known time to event.*

Table 7. *Comparison of expression levels of selected genes in Male and Female ICC patients (n = 17 males and 22 females). r is effect size.*

Variable		Mean Rank	U	Z	p	r
AQP3			163.000	-0.680	0.497	-0.1
	Males	18.59				
	Females	21.09				
FLJ23867			151.000	-1.020	0.308	-0.2
	Males	17.88				
	Females	21.64				
MGLL			165.000	-0.623	0.533	-0.1
	Males	18.71				
	Females	21.00				
PYGB			184.000	-0.085	0.932	-0.01
	Males	20.18				
	Females	19.86				
RAP1GAP			113.000	-2.096	0.036	-0.3
	Males	15.65				
	Females	23.36				

4 Discussion

The rise in incidence and mortality of intrahepatic cholangiocarcinoma is a cause for global concern [2]. From late diagnosis due to heterogenous presentation [9] to poor results from all known treatment options [4], identifying prognostic markers could be crucial for reducing harm caused by ICC. This work aimed to assist in that search using methods growing in power and popularity based in artificial intelligence.

The motivation behind this work was to investigate if specific genes could predict individual survival past two years. To begin isolating these features, unsupervised learning was applied to reveal any relationships between them. Hierarchical clustering was performed without much success, as no discernible patterns were revealed in the clusters produced other than the outlier group. This could largely be due to the selection of distance metrics. The heat map did show that filtering was successful for removing features without variability between samples. K-means clustering provided clearer results, showing that two clusters could be identified. However, the interpretation of what those clusters represent was not easily understood given the inability to plot the data given the large dimensions.

The results from the supervised learning were promising. Although accuracy was not at an acceptable level for practical use, these results showed that genes can aid in prediction to some degree. This exploration was also rudimentary; only five sets of genes were explored, models were limited to those available in the CLA, and discretized response variables were used. The features chosen by the MATLAB functions performed decently, but the predictive models would require much higher accuracy values to be deemed useful in a practical setting. And given how poorly the features selected based on literature performed, future work investigating this kind of feature selection should involve deeper investigation of the genes of interest.

Using a high filtering threshold may have removed some of the better predictive features, so experimenting with a different combination of filtering and feature selection functions might improve the model accuracy. Furthermore, alternative feature selection functions could be explored. Many more options exist as built-in options in MATLAB, and more methods specific to mRNA sequencing data are being released, as recently as this year [11]. Currently, the implemented feature selection methods were only applied to the filtered gene set and the genes selected from literature are all used. Applying these functions to the selections from literature may lead to potential improvement.

Dimensionality reduction techniques could also be applied prior to model building. In this work, PCA was applied after the selected features had been loaded into the Classification Learner App. This, or other methods such non-negative matrix factorization (NNMF), could be applied sooner and explored as well.

For an introductory exploration of predictive models, the CLA was adequate. However, further tuning of the models was limited and could likely have been enhanced by exporting the high-performing models as functions.

In addition, the time to event variable was discretized based on if the event occurs prior to or after two years. This removed information from the variables that might have been useful for building a predictive model, which could be investigated.

With the number of combinations possible in fine tuning these elements, there is potential for a more successful prediction model to be built.

Similarly, statistical analysis was limited as only the FSCNCA feature set with two classes was investigated. These were selected as they had the best performance in the supervised learning section. The expression levels of this feature set did not hold any significance in relation to survival and only one (RAP1GAP) presented with a significant difference between male and female ICC patients (p=0.036). These results did not support the initial hypothesis, but further investigation of a broader set of genes may reveal otherwise. In work done by the Broad Institute with this dataset, 30 genes were shown to have significant associations with time to event by Cox regression test [5].

Moreover, it could be of interest to perform statistical analysis with features shown to have importance in literature, such as IDH1 [17] and select those shown to be significant to build the predictive model with.

Working with publicly available data also introduced some minor issues. Mainly, not knowing or understanding entirely how the data had been processed before its use in this work. The normalization technique introduced values with unexplained relevance to the problem. For future work, working with raw gene counts and applying a different normalization technique may generate more successful results than those shown here. For instance, the authors of the RSEM method state that the raw data is easily translatable to Transcripts Per Million (TPM) with a simple multiplication factor [12].

While reducing large dimensions was an important issue to address, it should be noted that a larger number of samples is also something to take into consideration for future work. More than 39 samples will be necessary, especially if methods based in deep learning are included as potential avenues for exploration. These methods depend on large datasets, so introducing more samples, for example from additional public sources, is vital.

This project did not produce any conclusive results in regard to differential mRNA expression in ICC patients. However, the pipeline that was built in the process would allow for easier investigation of new feature sets or additional data.

References

1. Andersen JB, Thorgeirsson SS. Genetic profiling of intrahepatic cholangiocarcinoma. Current opinion in gastroenterology. 2012 May;28(3):266.
2. Banales JM, Marin JJ, Lamarca A, Rodrigues PM, Khan SA, Roberts LR, Cardinale V, Carpino G, Andersen JB, Braconi C, Calvisi DF. Cholangiocarcinoma 2020: the next horizon in mechanisms and management. Nature Reviews Gastroenterology & Hepatology. 2020 Sep;17(9):557-88.
3. Boerner T, Drill E, Pak LM, Nguyen B, Sigel CS, Doussot A, Shin P, Goldman DA, Gonen M, Allen PJ, Balachandran VP, Cercek A, Harding J, Solit DB, Schultz N, Kundra R, Walch H, D'Angelica MI, DeMatteo RP, Hechtman JF, Vakiani E, Lowery MA, Ijzermans JNM, Buettner S, Chandwani R, Koerkamp BG, Doukas M, Jarnagin WR. Alterations in TP53, KRAS and CDKN2A are independent prognostic factors for survival in intrahepatic Cholangiocarcinoma. In submission.
4. Brandi G, Farioli A, Astolfi A, Biasco G, Tavolari S. Genetic heterogeneity in cholangiocarcinoma: a major challenge for targeted therapies. Oncotarget. 2015 Jun 20;6(17):14744.
5. Broad Institute TCGA Genome Data Analysis Center (2016): Correlation between mRNAseq expression and clinical features. Broad Institute of MIT and Harvard. doi:10.7908/C1R210RT
6. Broad Institute TCGA Genome Data Analysis Center (2016): Firehose stddata__2016_01_28 run. Broad Institute of MIT and Harvard. doi:10.7908/C11G0KM9
7. Cancer of the Liver and Intrahepatic Bile Duct - Cancer Stat Facts [Internet]. National Cancer Institute. National Institutes of Health; 2020 [cited 2020 Oct 6]. Available from: https://seer.cancer.gov/statfacts/html/livibd.html.
8. Cohen J. Statistical power analysis for the behavioral sciences. Academic press; 2013 Sep 3.
9. Chong YS, Kim YK, Lee MW, Kim SH, Lee WJ, Rhim HC, Lee SJ. Differentiating mass-forming intrahepatic cholangiocarcinoma from atypical hepatocellular carcinoma using gadoxetic acid-enhanced MRI. Clinical radiology. 2012 Aug 1;67(8):766-73
10. Ding C, Peng H. Minimum redundancy feature selection from microarray gene expression data. Journal of bioinformatics and computational biology. 2005 Apr;3(02):185-205.
11. Kim S. A miRNA-and mRNA-seq-Based Feature Selection Approach for Kidney Cancer Biomakers. Cancer Informatics. 2020 Feb;19:1176935120908301.
12. Li B, Dewey CN. RSEM: accurate transcript quantification from RNA-Seq data with or without a reference genome. BMC bioinformatics. 2011 Dec 1;12(1):323.
13. Saha SK, Zhu AX, Fuchs CS, Brooks GA. Forty-year trends in cholangiocarcinoma incidence in the US: intrahepatic disease on the rise. The oncologist. 2016 May;21(5):594.

14. Sigel CS, Drill E, Zhou Y, Basturk O, Askan G, Pak LM, Vakiani E, Wang T, Boerner T, Do RK, Simpson AL. Intrahepatic cholangiocarcinomas have histologically and Immunophenotypically distinct small and large duct patterns. The American journal of surgical pathology. 2018 Oct;42(10):1334.

15. Survival Rates for Bile Duct Cancer [Internet]. American Cancer Society, Inc.; 2020 [cited 2020 Dec 19]. Available from: https://www.cancer.org/cancer/bile-duct-cancer/detection-diagnosis-staging/survival-by-stage.html

16. Tukey JW. Exploratory data analysis. 1977.

17. Wang T, Drill E, Vakiani E, Pak LM, Boerner T, Askan G, Schvartzman JM, Simpson AL, Jarnagin WR, Sigel CS. Distinct histomorphological features are associated with IDH1 mutation in intrahepatic cholangiocarcinoma. Human pathology. 2019 Sep 1;91:19-25.

18. Yang W, Wang K, Zuo W. Neighborhood Component Feature Selection for High-Dimensional Data. JCP. 2012 Jan;7(1):161-8.