Contents

1 Introduction

Cancer, a disease marked by its complexity encompasses both invasive and noninvasive types, each presenting unique diagnostic and therapeutic challenges. Invasive cancer refers to cancer that has spread from the original tissue where it developed into surrounding tissues, on the other hand noninvasive cancers is one which do not spread. This project leverages gene expression data to explore the differences between invasive and noninvasive cancer, employing advanced statistical techniques to analyze patterns and identify biomarkers. Through a methodical approach, including variance-covariance analysis, dimensioanlity reduction, and unsupervised and supervised machine learning, we aim to uncover the genetic signatures that distinguish invasive from noninvasive cancer, thereby contributing to the broader field of cancer research and its application.

2 Data Exploration and Preliminary Analysis

Preliminary analysis of gene expression data is essential for understanding the dataset's structure, identifying potential issues and making informed decisions about further statistical analysis. Our approach began with an initial phase of data cleaning and preprocessing. We initiated the process by identifying missing and infinite values within the dataset, uncovering 177 missing values and no infinite values. The missing values were substituted with the mean of the dataset. A boxplot was drawn to understand the outliers present.

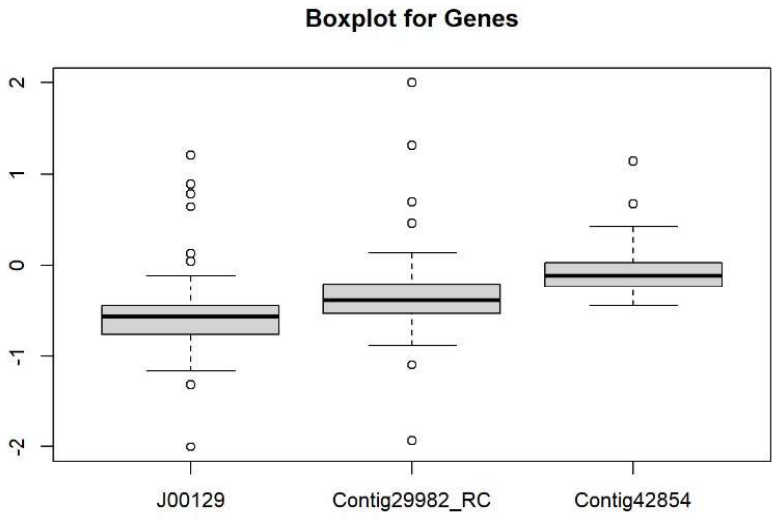

Figure 1: Boxplot showing outliers for 3 gene variables

We then calculated the statistical measures (mean, median, standard deviation, variance, and range) for each gene expression variable and tabled measures of the first gene expression variable. A histogram and Q-Q plot were drawn to assess the distribution of gene variable.

	Mean	Median	StandardDeviation	Variance	Range
J00129	-0.55766667	-0.574	0.4499578	0.20246199	3.206
Contig29982_RC	-0.33938462	-0.396	0.4673664	0.21843138	3.927
Contig42854	-0.07625641	-0.110	0.2508342	0.06293222	1.593
Contig42014_RC	-0.03465385	-0.058	0.2094581	0.04387270	1.292
Contig27915_RC	-0.04126923	-0.090	0.2436339	0.05937550	1.127

Table 1: Statistical Measures for the first 5 Genes

Variance Analysis has been performed to identify genes with high variance, as these might contribute most to distinguishing between invasive vs noninvasive cancer.

To minimize multicollinearity within the dataset, pairs of variables with high correlation were identified, and one variable from each pair was eliminated. The criterion for elimination was set at a correlation greater than 0.7.

By performing these steps our dataset reduced from (78 x 2000) to (78 x 1429). We established the basis for our analysis by setting the seed value to the highest registration number (2315740) amongst our team members.

3 Main Analysis: Models, Results and Evaluation

3.1 Task 1: Dimensionality Reduction

Consider unsupervised and supervised dimension reduction of the 2000 observed gene expression values in your data set

3.1.1 Comparing Unsupervised and Supervised Dimensionality Reduction

Dimensionality reduction is aimed to reduce the number of features under consideration. Dimensioanlity reduction can be performed through unsupervised and supervised methods. Although given the nature of our dataset and the tasks we were required to perform, after thoughtful consideration, Principal Component Analysis (PCA) as means for dimensionality reduction fit our criteria. Following are the reason why Principal Component Analysis (PCA) is better suited for our dataset and analysis.

Compatibility with Clustering: PCA is an unsupervised dimensionality reduction technique that focuses on capturing the maximum variance in the data. Clustering algorithms such as K-means and

hierarchical clustering can be applied directly to the principal components obtained from PCA.

Compatibility with Classification: PCA can also be used as a pre-processing step for classification tasks. By reducing the dimensionality of the feature space, PCA can help improve the performance of classification algorithms by reducing the risk of over-fitting and multi-collinearity.

Interpretability: PCA provides interpretable components that represent combinations of original features. This can aid in understanding the underlying structure of the data and identifying important features for clustering and classification.

Efficiency: PCA is computationally efficient and can handle large datasets with many features. This makes it suitable for both clustering and classification tasks, especially when dealing with high-dimensional data

Overall, PCA offers a versatile and effective approach to dimensionality reduction that is well-suited for both clustering and classification tasks. It provides a balance between preserving important information in the data and reducing its dimensionality, making it a valuable technique in various data analysis scenarios such as ours.

Based on the given factors, we reduced the dimensions of our dataset based on principal component analysis in Task 2.

3.2 Task 2: Unsupervised Learning

Unsupervised learning models. Principal Component Analysis, k-means clustering and hierarchical clustering.

3.2.1 Principal Component Analysis

On our pre-processed dataset (78 x 1429), we applied principal component analysis. The summary of the results can be observed in the table below.
It can be seen from above table that PC1 is covering about 11.87 percent of total variance similarly for PC2 the percentage coverage of variance is 6.7 percent up to PC78 where variance coverage is 0 percentage. The cumulative proportion shows that first three components are providing about 28.49 percentage of coverage of variance. Using a benchmark of 80% explained variance, we created reduced the feature space from 1429 variables to 36 variables. Creating a dimensionally reduced dataset (78 x 26). This dataset, becomes the base of our analysis in unsupervised and supervised learning techniques.

Table 2: Summery Table of PCA

	Standard deviation	Proportion of variance	Cumulative variance
PC1	13.0223	0.1187	0.1187
PC2	9.8284	0.0676	0.1863
PC3	9.25056	0.05988	0.24615
.	.	.	.
.	.	.	.
.	.	.	.
PC77	1.74541	0.00213	1.00000
PC78	5.783e-15	0.000e+00	1.000e+00

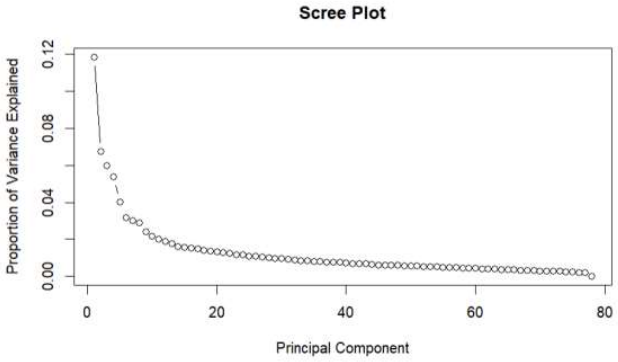

Figure 2: Individual Explained Variance

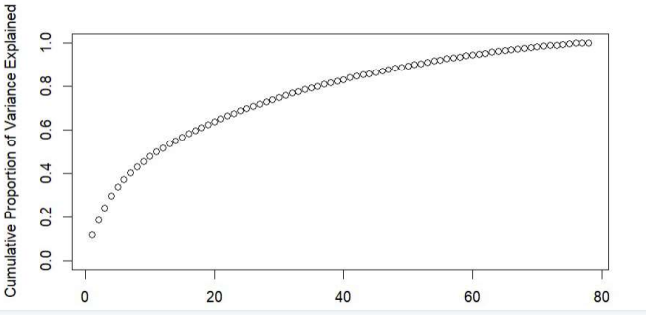

Figure 3: Cumulative Explained Variance graph

3.2.2 K-means Clustering

In this clustering algorithm, the dataset is partitioned into k distinct, non overlapping clusters. After performing the k-means, we get the output as a set of k=3 clusters with minimized within-cluster variances and maximized between-cluster distances. (Refer Figure 7 - Appendix) Cluster 1 (red) seems to have a relatively smaller number of points compared to the blue cluster and is spread over a range of values on Principal Component 2. The points in cluster 2 (blue) are closely packed, indicating high similarity within this group according to the PCA features. Cluster 3 (green) points are less dispersed than Cluster 1 and are mostly found in the positive regions of both principal components.

The tight grouping of the blue cluster indicate a strong underlying pattern, while the more dispersed red and green clusters could be areas of interest for further analysis to understand the broader variability within those groups.

3.2.3 Hierarchical Clustering

This clustering approach begins by considering each data point as an individual cluster, leading to a total of X clusters for X data points. Euclidean distance is used to calculate similarity between all cluster

pairs, the method merges the two closest clusters using Ward's minimum variance method. (Refer Figure 8 - Appendix)

The x-axis represents the individual data points (or initial clusters), and the y-axis represents the distance or dissimilarity between clusters. The height of the horizontal line indicates the distance at which the two clusters were merged.Red rectangles represents number of clusters that we want for our analysis and we can adjust it according to our requirement.

3.2.4 Agglomerative Clustering

Agglomerative Clustering is a hierarchical method for grouping objects according to similarities in unsupervised learning. Every data point is initially regarded as a single cluster. A selected metric, like the Euclidean distance, is used to compute the distances between clusters. Then, using Ward's method to minimise within-cluster variance, clusters are merged iteratively. The procedure keeps going until the required number of clusters is reached or all data points are combined into a single group. The merging process is represented visually by a dendrogram, where the y-axis denotes dissimilarity and the x-axis data points or initial clusters. The number of clusters selected for analysis is indicated by red rectangles; this number can be changed in accordance with requirements.

3.3 Task 3: Supervised Learning

Supervised learning models. Apply Logistic Regression, LDA, QDA, k-NN, Random Forest and SVM. Discuss why you choose specific hyper parameters of a supervised.

3.3.1 Splitting the Dataset

Splitting Dimensionally Reduced Dataset into Train and Test Dataset: Considering our dimensionally reduced dataset (78 x 36) obtained through Principal Component Analysis as base, the dataset is then split into training and testing sets with and 80-20 ratio. The reason to select 80-20 ratio is due to the limited set of observations. Doing so, will help in training the models, and limiting class imbalance in our training dataset. Below is the split of each dataset.

	Training Dataset		Test Dataset	
Class Label	1	2	1	2
Frequency	31	31	3	13

Table 3: Class distribution in Training and Test Datasets

3.3.2 Logistic Regression

Logistic Regression uses sigmoid function, providing a probability-based approach to perform classification. After defining control parameters, specifically a 3-fold cross validation approach, the logistic regression model is trained. The best performing hyper parametric values in "glmnet" package for logistic regression used for regularization namely "alpha" and "lambda" are plugged in to train and develop the best performing logistic regression model. The "alpha" parameter controls the type of regularization (0 for ridge regression, 1 for lasso regression), while "lambda" regulates the strength of regularization. The model exhibits mean accuracy of roughly 58% across different validation sets. The accuracy of the model on the test dataset is 62.5%. This could largely be due to the effectiveness of model performance on unseen datasets.preparation to model evaluation in the analytical process.

3.3.3 Linear Discriminant Analysis

LDA seeks to find a linear combination of features that best separates multiple classes in the data by maximizing the between-class variance while minimizing the within-class variance, resulting in a low-dimensional representation that preserves the discriminatory information between classes (Fisher, 1936)[1]. Using 3-fold cross validation, LDA model was trained. The accuracy of the model of test dataset is 68.75% indicating decent level of performance. Overall, these results indicate that the LDA model, trained without parameter tuning, achieves moderate accuracy and a fair level of agreement beyond chance in its predictions.

3.3.4 Naïve Bayes

Naive Bayes predicts the class label of a given instance by calculating the conditional probability of each class given the features and selecting the class with the highest probability (Rish, 2001).[2] The results indicate the effectiveness of incorporating kernel density estimation in Naïve Bayes model. The model with kernel density estimation outperforms across different validation sets. Leading to more accurate and predictable results. The overall accuracy of the best Naïve Bayes model trained in consideration to above results, against the test dataset is 62.5% suggesting decent performance.

3.3.5 K-Nearest Neighbours

In classification, k-NN predicts the class of a query point by identifying the majority class among its k nearest neighbors in the feature space (Hastie, 2009)[3]. For example, if k=5 and three out of the five nearest neighbors belong to class A while the other two belong to class B, the 5NN algorithm assigns class A to the new data point.

Due to the nature of our dataset, the model was trained using 3-fold cross validation as it offers balance with sample size and the number of times the validation is performed. Hyper-parametric tuning is performed using "grid search" method where "k (no of nearest neighbours)" is the control parameter. From the validation results, "k=7" resulted in best model performance with the accuracy of 87.5% against the test dataset. The mean accuracy across validation set was roughly 60% suggesting that the knn-model either handles unseen data well or classifies either class label more accurately than other.

3.3.6 Support Vector Machines

SVM works by finding the optimal hyperplane that best separates the different classes in the feature space, maximizing the margin between the closest data points of different classes (Cortes & Vapnik, 1995)[4]. SVM model is prepared using "radial basis function" in R that allows hyper-parametric tuning of "sigma (indicating control of width)" and "c (penalty/cost for error)". After cross validation, best parameters are selected to tune the model. The accuracy of the resultant model is 56.25% indicating improvement over random chance – 50% in binary classification. This could be due to limited dataset, as SVM requires large number of support vectors to generate hyperplanes.

3.3.7 Random Forest

Random Forest is an ensemble learning method that builds multiple decision trees during training and outputs the mode of the classes (classification) of the individual trees. Given our dataset, certain aspects of Random Forest model are important to consider. Number of trees in the model, maximum depth of the trees, and feature selection through bagging or bootstrap sampling are amongst the most important aspects of the model. We have tuned these parameters after cross validation to create the best model that avoids underfitting and overfitting. The average accuracy across different validation sets was roughly 55% whereas the accuracy of the model against test dataset is 62.5%.

3.3.8 Model Results and Evaluation

Using resampling, the best models for each algorithm were tested against accuracy metric. The number of resamples was set to three. Below is the performance of each model on resampled dataset vs the test dataset.

	Accuracy on Resampled Dataset			Accuracy on Test Dataset
Model	Min.	Mean	Max.	Through Best Model
Logistic Regression	57%	64%	72%	63%
Linear Discriminants	52%	60%	69%	69%
Naïve Bayes	52%	62%	67%	63%
k-Nearest Neighbours	50%	66%	70%	88%
Support Vector Machines	52%	62%	69%	56%
Random Forest	59%	63%	69%	63%
Extreme Gradient Boost	57%	62%	67%	69%

Figure 4: Consolidated Model Results and Evaluation

On the resampled datasets, the mean accuracy ranges around 62%, with k-Nearest Neighbors exhibiting the highest mean accuracy of 66%, and Linear Discriminant Analysis the lowest with 60%. On the test dataset, the accuracy through the best model varies, with k-Nearest Neighbors achieving the highest accuracy of 88%, and Support Vector Machines the lowest with 69%. These results provide insights into the relative performance of different models and their generalization capabilities. The discrepancies in the model accuracies could be due to sampling variability and data distribution leading to differences in characteristics of the data seen by the model during training, tuning and evaluation.

3.4 Task 4: Comparing and Evaluating Best Machine Learning Model

Investigate if clusters established under 2 improve your 'best' machine learning model.

Based on the accuracy against test dataset and resampled dataset, the k-NN model generally performed better amongst others with an accuracy of 87.50%. Since our supervised models were trained and tested against data obtained through PCA, the semblance of clustering and feature engineering was already built in. To understand the impact of clustering, we tested the k-NN model against our unfiltered dataset containing over 1000 variables (78 x 1430). This approach allows us to gauge the impact and performance of the model across different types of datasets. The 80:20 ratio of training and testing split was kept reducing variability. The same approach incorporating k-fold cross validation and hyper parametric tuning of "k" was used to train the model against unfiltered dataset.

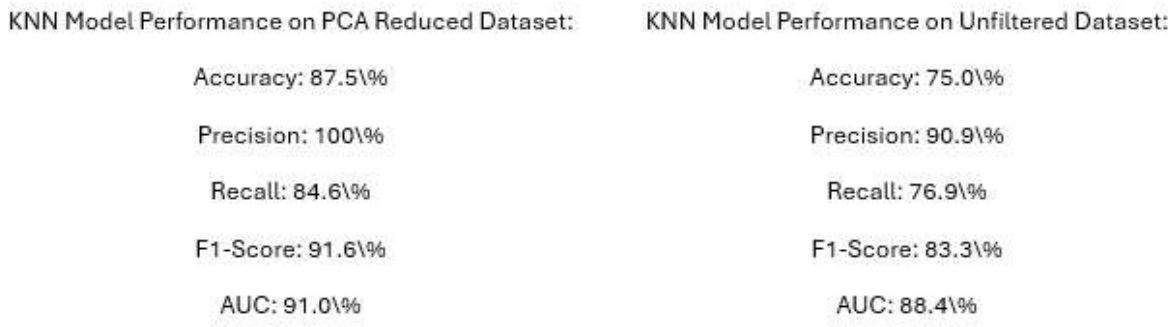

Figure 5: KNN Model performance

Overall, these findings show that although the k-NN model performs well on both datasets, the dimensionality and complexity of the input features have an influence on the model's performance. Higher accuracy, precision, recall, F1-score, and AUC values when compared to the unfiltered dataset suggest that the smaller feature space obtained through PCA contributes to improved performance as observed in ROC Curve aswell(Refer to Figure 9 - Appendix). This shows that by removing noise from the dataset and extracting pertinent information, feature engineering methods like PCA can significantly improve the performance of machine learning models.

4 Conclusion

All things considered, this work provides a thorough analysis of gene expression data-based cancer classification approaches that make use of a variety of statistical methods and machine learning algorithms. Principal Component Analysis (PCA) was utilized to reduce the dimensionality of the high-dimensional dataset and extract important insights that allowed for a more in-depth comprehension of the underlying patterns and structures.

Distinct gene clusters and subgroups were revealed by unsupervised learning tasks, such as k-means clustering and hierarchical clustering, offering important new insights into potential biomarkers and disease stratification. This advances our understanding of the heterogeneity of cancer and sets the stage for individualized treatment plans catered to individual patient circumstances.

After extensive training and evaluation, K-Nearest Neighbours (KNN) emerged as the best classifier among supervised learning models, which ranged from Logistic Regression to Extreme Gradient Boosting. The KNN model's high accuracy in both the test and resampled datasets, along with its robust performance, highlights this model's potential as a dependable tool for cancer classification tasks.

Intriguing findings about the relationship between data dimensionality and model performance were also made during investigation into the effects of feature engineering methods like PCA. The effectiveness of dimensionality reduction in improving predictive accuracy and generalization abilities is demonstrated by the KNN model's superior performance on the PCA-reduced dataset as compared to unfiltered dataset.

This study highlights the transformative potential of using machine learning algorithms and advanced statistical methods in cancer research, even beyond technical nuances. By deciphering complex patterns in gene expression data, we improve our knowledge of the biology of cancer and open the door to novel therapeutic and diagnostic approaches.

All things considered, this work is a comprehensive attempt to leverage data-driven strategies in the fight against cancer, with a focus on methodological rigor and interdisciplinary cooperation as key drivers of scientific discovery and clinical translation. The insights gained from this study have great potential in establishing the use of such algorithms to transform cancer diagnosis, prognosis, and treatment paradigms in the era of precision oncology with further validation and refinement.

References

[1] R. A. FISHER. The use of multiple measurements in taxonomic problems. *Annals of Eugenics*, 7(2):179–188, 1936.

[2] Irina Rish. An empirical study of the naïve bayes classifier. *IJCAI 2001 Work Empir Methods Artif Intell*, 3, 01 2001.

[3] T. Hastie, R. Tibshirani, and J.H. Friedman. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer series in statistics. Springer, 2009.

[4] C Cortes and Vladimir N Vapnik. A training algorithm for optimal margin classifiers. In *Support-vector networks. Machine Learning, 20(3)*, pages 144–152, 1995.