Machine Learning-Based Cancer Stage Prediction Using Clinical, Genomic, and Demographic Data

Joshua Efraim Rawatan
Computer Science Department
School of Computing and Creative Arts
Bina Nusantara University
Jakarta, Indonesia 11480
joshua.efraim001@binus.ac.id

Registan
Computer Science Department
School of Computing and Creative Arts
Bina Nusantara University
Jakarta, Indonesia 11480
registan@binus.ac.id

Zhandos Yessenbayev
Computer Science Department
School of Computing and Creative Arts
Bina Nusantara University
Jakarta, Indonesia 11480
zhandos.yessenbayev@binus.edu

***Abstract*—The accurate prediction of metastatic cancers with clinical data still poses a significant challenge. Machine learning techniques have demonstrated potential for assisting in the prediction of metastatic cancer. This study evaluates five different machine learning models for cancer metastasis prediction using clinical, demographic, and genomic indicator data. Model performance was measured and compared using macro-averaged F1-score and class-specific sensitivity to account for class imbalances. Results from model assessment indicate that tree-based models including XGBoost and Random Forest achieved the highest macro-averaged F1-scores of 0.73. Between the two models, XGBoost demonstrated higher sensitivity for the metastatic class, achieving a sensitivity score of 0.75. These findings highlight the importance of model robustness and strength of tree-based models for clinical metastasis prediction.**

I. Introduction

Cancer is not a single disease but a group of more than 100 distinct subtypes. All cancers share something in common: a breakdown in the process of cell division and programmed cell death. As a result, abnormal or damaged cells meant to die continue to grow and multiply, forming tumors [1]. When tracking the progression of cancer in the body, staging from 0-4 is used. Stage 0 often refers to abnormal cells that have not formed a tumor. Stages 1-3 indicates that there is a tumor which is growing in size but only in a local area. In stage 4 the cancer has become metastatic, meaning that the cancer cells have spread to further regions of the body from the initial site [2]. Determining cancer stage quickly is valuable, it directly correlates with the progression of the disease. Cancer stage can also be used to measure the effectiveness of a treatment on a patient. This allows for the finding and testing of new treatments and judging their effectiveness [2]. Recent advances in machine learning technology makes it possible to train a model to predict cancer stage. With this model, it is possible to determine whether a cancer is metastatic quickly with only clinical and demographic data without the need of genomic expressions. A major challenge of training a model to accurately predict cancer metastasis is the data. The data obtained from the dataset contains significant noise (Missing Values) that could cause worse model performance. Picking suitable models fit for this study and proper preprocessing is required to have a good final result.

The use of machine learning techniques in cancer prognosis and prediction has already been widely explored. Multiple studies have proven the effectiveness of machine learning in the analysis of complex medical data [3]-[5]. Earlier studies have demonstrated the use of SVM (Support Vector Machine), Decision Trees, and Neural Networks to generalize and classify cancer patients into low- and high-risk categories [3]. These findings highlight the ability of machine learning to recognize patterns from large datasets. However, predictions from these trained models have been reported to require external validation [3]. Further studies have investigated the use of gene expression data in cancer stage prediction. In one study, an extreme gradient boosting (XGBoost) model was trained on gene expression data from cancer patients and achieved an accuracy of 82% based on the Wilcoxon rank test [4]. These results demonstrate the effectiveness of advanced tree-based models in predicting cancer stage using genomic expression data. Advanced neural network architectures have also been proposed to improve precision in cancer stage prediction. More recently, a study evaluated the use of hybrid deep learning that includes Extreme Learning Machine (ELM) and Deep Belief Network (DBN) trained on data from The Cancer Genomic Atlas [5]. This hybrid model was reported to achieve classification accuracies ranging from 89.35% to 98.75% for early-

979-8-3315-2768-6/24/$31.00 © 2024 IEEE

and late-stage cancers. This result demonstrates the advantages of hybrid neural networks over conventional models in large multi-modal cancer datasets.

II. Contributions

This study is less focused on a singular type of cancer and uses 5 different types of cancer such as non-small cell lung cancer, colorectal cancer, breast cancer, prostate cancer, and pancreatic cancer. The multi-modal integration in this study also helps in a more holistic approach using 3 different independent variables which are demographic, clinical, and genomic.

III. Methodology

A. Dataset

The dataset was obtained from cBioPortal where it contained 25 thousand samples of patients that were diagnosed with cancer. It contains the background details of the patient and the characteristics of the cancer such as the detailed cancer type, the stage, and tumor sites. Demographic, clinical, and genomic features are going to be used from this dataset as the dependent variable, and the independent variable would be the cancer stage.

B. Libraries

The libraries that are used throughout the code are Pandas, NumPy, Scikit-learn, XGBoost, Matplotlib, and Seaborn. Pandas and NumPy are essential when preprocessing the data, to read, format, remove unnecessary features, and transform the values. Scikit-learn is the main framework that is used for processing the dataset and training the models. Finally, Matplotlib and Seaborn are used for visualisations to help with interpreting the results and data.

C. Preprocessing

The preprocessing pipeline involves many steps to ensure that the training is optimized. The first step is to read the MSK_CHORD tsv dataset. Then the dependent variable (cancer stage) is filtered to only keep values of "Stage 1-3" and "Stage 4". The demographic, clinical, and genomic features are then selected as independent variables. These demographic attributes include age, sex, ethnicity, and other background details. Clinical descriptors are cancer type, the detailed type, and the tumor site. The genomic variables are mutation count, msi score, msci type, etc.

The missing values are then filled using median for numerical features which also helps smooth the dataset from distant outliers, while missing categorical features are filled with "Unknown".

Case normalization is then done on the features because some values may be the same however because there is an uppercase will result in being categorized differently. Features that contain rare values with frequencies less than 10 will be set as "other", this helps to minimise noise that these very few categories create. Features such as "TMB (nonsynonymous)" and "Mutation Count" contain numerical values that are mostly small and fewer large ones, so log1p(x) = log(x+1) is applied so that the difference in values are less significant. The tumor site features which contain "Yes/No/Unknown" are encoded into "1/0/0 " so that it can be processed. Binary feature engineering is also done on the dataset for features such as "TMB (nonsynonymous)", "Fraction Genome Altered", "MSI Score", and "Tumor Purity". Where if the tumor mutational burden value is more than 10, it will be set as 1 (high), this rule applies to fraction genome altered if it is higher than 20%, MSI score if more than 10, and tumor purity if higher than 50% of the tumor cells. Extremely sparse one-hot encoded features are then removed where the threshold is 1%, this helps to lower the noise even more and optimise the model training process.

After preprocessing the data, the features are set as the independent variable (x) and the "Stage (Highest Recorded)" as the dependent variable (y). Then split the dataset for training and testing, where 80% of the dataset is used for training and the other 20% for testing. After that, the imbalance of the dependent variable is handled to ensure that the model will not be biassed, and finally feature scaling is done for logistic regression and SVM to also minimise any biases involving the larger scale features.

D. Model Selection and Training

The models selected for this study are Logistic Regression, Random Forest, XGBoost, Artificial Neural Network, and Support Vector Machines.

1) Logistic Regression: Logistic Regression is a linear model that is often used for binary classifications. Class weighting is applied to ensure that the balance between non-metastatic (Stage 1-3) and metastatic (Stage 4) cancer samples are maintained. The model is then trained on the features that were already previously scaled.

2) Random Forest: Random Forest is a model that builds multiple decision trees, where each tree is made to look at another perspective of the data. The results are then combined and averaged. The hyperparameters are tuned to ensure that the model does not overfit, the model is limited to only build 400 trees for efficiency, the max depth is set to 20 so that the model does not memorize very specific patterns, the minimum samples split is set to 10 so that there is a minimum of 10 splits, and lastly the minimum samples leaf is set to 5 to make each leaf have a minimum of 5 samples.

3) XGBoost: The third model is XGBoost, where it is an advanced machine learning algorithm that uses the Gradient Boosting framework. XGBoost creates decision trees and each decision tree is made to learn from the previous errors. The model is tuned with 600 trees, a low learning rate of 0.2 to maintain precision, a max depth of 8 to minimise the risk of overfitting, and subsample of 0.85 to use 85% of data per tree to avoid overfitting and help with generalisation.

4) Artificial Neural Network: Artificial Neural Network is a model that imitates how the human brain processes data, where it uses artificial neurons to learn. The model has two layers where one is 100 neurons and the other is 50 neurons. The first layer learns the high level patterns in the data, while

the second layer helps to combine the correlations. ReLU activation function is used for learning non-linear correlations and more complex relationships. Adam optimizer is also used for updating the weights of the features during training. L2 regularization parameter of 0.01 and early stopping is used to prevent the model from overfitting.

5) Support Vector Machine: The last model is Support Vector Machines, where it is a supervised machine learning algorithm that is often used for classifications. It finds the hyperplane or the decision boundary to separate the classes. The RBF kernel helps the model to find complex non-linear relationships and uses the regularization parameter C of 1.0 for balancing flexibility and generalization.

E. Evaluation Metrics

The evaluation metrics used in this project are accuracy, precision, recall, and F1-score. Each metric is used to measure the usability and help compare every model. The metrics that are used:

1) Accuracy: Accuracy measures the proportion of correct predictions among all predictions made by the model. It reflects the overall correctness of the classifier.

$$\text{Accuracy} = \frac{Correct\ Predictions}{Total\ Predictions}$$

2) Precision: Precision: measures how many predicted positives are actually positives.

$$\text{Precision} = \frac{True\ Positive}{True\ Positive + False\ Positive}$$

3) Recall: Recall: measures how many positive samples are predicted correctly.

$$\text{Recall} = \frac{True\ Positive}{True\ Positive + False\ Negative}$$

4) F1-score: F1-score: measures the mean of precision and recall.

$$\text{F1-score} = 2 \text{ x } \frac{Precision\ X\ Recall}{Precision + Recall}$$

IV. Results and Discussion

TABLE I
MACRO-AVERAGED MODEL PERFORMANCE COMPARISON

Model	Precision	Recall	F1-Score	Accuracy
Logistic Regression	0.70	0.71	0.70	71.09%
Random Forest	0.73	0.73	0.73	73.46%
XGBoost	0.73	0.74	0.73	73.92%
ANN	0.72	0.72	0.72	72.54%
SVM	0.72	0.73	0.72	73.04%

The most important metric in this paper will be F1-score and this is because F1-score is generally more relevant than using accuracy. The portion of stage 1-3 (non metastatic) and stage 4 (metastatic) is 60:40, meaning that a model constantly predicting stage 1-3 would immediately get a 60% accuracy. By using F1-score, the model's performance is filtered through precision and recall. If either precision or recall is low, then the F1-score metric will be penalized, thus having a high F1-score means that the model is performing well and the metric is more reliable than using accuracy. Macro-average is used in this study so that both classes are treated equally as there is an imbalance of testing data.

Overall, all 5 models are able to achieve an F1-score of over 0.7, XGBoost and Random Forest performed the best but are tied with a score of 0.73. Logistic Regression is the baseline as it has the lowest performance macro averaged F1 score of 0.70 which is followed by ANN and SVM with a score of 0.72.

TABLE II
RECALL FOR STAGE 4 METASTATIC

Model	Recall (Stage 4)
XGBoost	0.75
Random Forest	0.73

When sensitivity (Recall) is considered for Class 1 (Metastatic), XGBoost is superior to Random Forest with a score of 0.75 over 0.73. In this case, sensitivity is prioritized due to its relevance in classifying the proportion of metastatic cancer cases correctly. This is critical as falsely classifying a metastatic cancer stage as non metastatic can result in delayed/improper treatment and a wrong prognosis. It is observable that XGBoost is superior and this is because it is well suited to tabular datasets and can deal with noise in datasets by creating a new tree to fix errors from the previous tree. In contrast, Random Forest averages predictions from multiple trees, improving its robustness but not to the extent of XGBoost's error correction. Logistic Regression did not perform well as it assumes a linear relationship between features and the outcome.

The figure above shows that when doing stage predictions, clinical features contribute the most with 78.6%. This is a large margin compared to demographic and genomic that is only able to contribute with 10.9% and 10.6% respectively. A reason for this is because clinical characteristics of the cancer are much more relevant than the demographic details and genomic features of the patient. Another reason is that the dataset is much more heavy on clinical features and the categorical ones expand after being one-hot encoded.

The figure above shows the top 5 contributing features for prediction. It shows that all 5 comes from the clinical data, meaning that not only do clinical features have the largest share in the feature importance, but the top 5 contributing features are also dominated as well. The result of this study is important, other studies go in depth in risk grouping or stage predictions. The difference for stage prediction is that

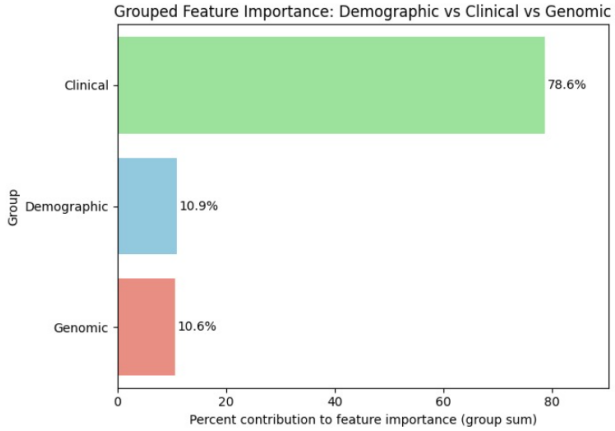

Fig. 1. Feature importance

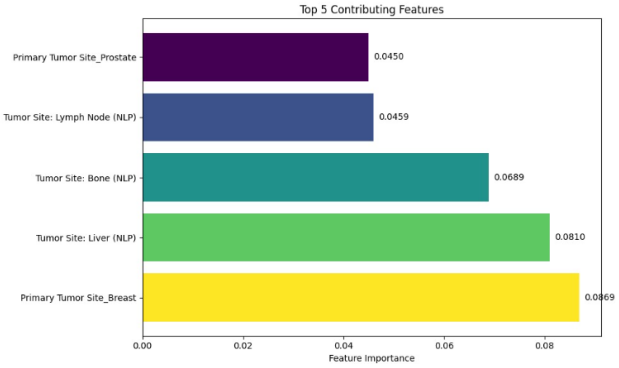

Fig. 2. Top 5 contributing features

this study does not use the gene expression or genomic data but more widely available clinical data. This is important as clinical data can be obtained earlier in the treatment of a patient, our model could quickly assist in determining whether a patient's cancer is metastatic. This will help in the choice of treatment of a patient. Although other studies have higher accuracies of 80-99%, the accuracy of this study is only 74% for the best model (XGBoost). This is due to the dataset only containing clinical data points and not the genomic data of the other studies, thus it is not directly comparable to the accuracies of the other studies.

V. Limitations

1) The model only predicts between stage 1-3 and 4, so it cannot be used to predict the specific stage of cancer. The model can only be used to help determine whether the cancer is metastatic or not.
2) The model f1-score is at 0.73 with accuracy of 70%, this shows that the model still has almost a 30% chance of being incorrect. Thus it can only be used as a supporting decision rather than the main conviction when predicting the cancer stage.
3) There might be an imbalance in values of the features, some features may contain more of one value thus the model may be more biased towards the certain cancer stage.

VI. Conclusion

In this study, multiple machine learning techniques were evaluated for cancer metastasis prediction using clinical, demographic, and genomic indicator data. The results indicate that tree-based models, specifically XGBoost and Random Forest, achieved superior macro averaged F1-scores compared to other evaluated models. While XGBoost and Random Forest had the same macro averaged F1-scores of 0.73, but XGBoost particularly had a superior sensitivity for the metastatic class with a score of 0.75. The analysis also revealed a slight variability in model performance across multiple runs highlighting the importance of model robustness when applied in medical datasets. In future work, the integration of a larger dataset could be explored to improve predictive performance and accuracy.

References

[1] "What is cancer?," *Cancer.gov*, Oct. 11, 2021. https://www.cancer.gov/about-cancer/understanding/what-is-cancer

[2] C. C. M. Professional, "Cancer staging," *Cleveland Clinic*, May 06, 2025. https://my.clevelandclinic.org/health/diagnostics/22607-cancer-stages-grades-system

[3] K. Kourou, T. P. Exarchos, K. P. Exarchos, M. V. Karamouzis, and D. I. Fotiadis, "Machine learning applications in cancer prognosis and prediction," *Computational and Structural Biotechnology Journal*, vol. 13, pp. 8–17, Nov. 2014, doi: 10.1016/j.csbj.2014.11.005.

[4] S. Chen, "Predicting lung cancer stage by expressions of Protein-Encoding genes," *Advances in Bioscience and Biotechnology*, vol. 14, no. 08, pp. 368–377, Jan. 2023, doi: 10.4236/abb.2023.148024.

[5] A. Amanzholova and A. Coşkun, "Enhancing cancer stage prediction through hybrid deep neural networks: a comparative study," *Frontiers in Big Data*, vol. 7, p. 1359703, Mar. 2024, doi: 10.3389/fdata.2024.1359703.

[6] GeeksforGeeks, "Logistic regression in machine learning," *GeeksforGeeks*, Nov. 18, 2025. https://www.geeksforgeeks.org/machine-learning/understanding-logistic-regression/

[7] GeeksforGeeks, "Random Forest algorithm in machine learning," *GeeksforGeeks*, Oct. 31, 2025. https://www.geeksforgeeks.org/machine-learning/random-forest-algorithm-in-machine-learning/

[8] GeeksforGeeks, "XGBoost," *GeeksforGeeks*, Oct. 24, 2025. https://www.geeksforgeeks.org/machine-learning/xgboost/

[9] GeeksforGeeks, "Artificial Neural Networks and its Applications," *GeeksforGeeks*, Jul. 12, 2025. https://www.geeksforgeeks.org/artificial-intelligence/artificial-neural-networks-and-its-applications/

[10] GeeksforGeeks, “Support Vector Machine (SVM) algorithm,” *GeeksforGeeks*, Nov. 13, 2025. https://www.geeksforgeeks.org/machine-learning/support-vector-machine-algorithm/