Comparing the Performance of Traditional Machine Learning and Deep Learning Algorithms for Breast Cancer Survival Prediction

Navaneeth Ranjit
navirn2009@gmail.com
Private International English School
Mussafah, Abu Dhabi, UAE

Abstract

Breast cancer remains the most commonly diagnosed cancer among women worldwide, with disparities in outcomes influenced by geographic and socioeconomic factors. In this study, we address the challenge of predicting breast cancer survivability, specifically in scenarios with limited data.Using the METABRIC (Molecular Taxonomy of Breast Cancer International Consortium) dataset, which comprises clinical and genetic attributes from 1,980 primary breast cancer samples, we performed binary classification to predict patient survival outcomes. Clinical and genetic data were retrieved from the cBioPortal platform. We explored a range of machine learning (ML) and deep learning (DL) models, including 13 traditional ML algorithms, viz., k-Nearest Neighbors (KNN), logistic regression, decision trees, as well as 6 deep learning architectures including multilayer perceptrons (MLPs), TabNet, and DeepFM. In addition, we implemented a time-to-event analysis using a Cox proportional hazards pipeline with Elastic Net regularization and deep survival models such as DeepSurv and DeepHit to explore survival prediction beyond binary classification. These models were trained and tested on individual clinical and genetic attributes as well as their combination. We also evaluated the effectiveness of model stacking and voting classifiers to combine the strengths of multiple ML models in terms of accuracy, precision, recall, and F1 score. A soft voting classifier combining LightGBM, CatBoost, and AdaBoost, trained and tested on clinical attributes, provided the best performance, while deep learning models underperformed likely due to the limited dataset size. Overall, we observed that the models performed better when trained on clinical features rather than using genetic attributes, thereby suggesting that the clinical indicators remain more predictive of breast cancer survivability.

Keywords

Breast Cancer Survival, Machine Learning, Deep Learning, Ensemble Models, Tabular Data, METABRIC, Genetic Attributes, Clinical Attributes

ACM Reference Format:
Navaneeth Ranjit. 2026. Comparing the Performance of Traditional Machine Learning and Deep Learning Algorithms for Breast Cancer Survival Prediction. In *Proceedings of International Journal of Secondary Computing and Applications Research (IJSCAR VOL. 3, ISSUE 1).* ACM, New York, NY, USA, 11 pages. https://doi.org/10.5281/zenodo.18404485

This paper is published under the Creative Commons Attribution-NonCommercial-NoDerivs 4.0 International (CC-BY-NC-ND 4.0) license. Authors reserve their rights to disseminate the work on their personal and corporate Web sites with the appropriate attribution.
IJSCAR VOL. 3, ISSUE 1
© 2026 The Author(s). Published by IJSCAR.

1 Introduction

Breast cancer is an abnormal growth within the body and is an invasive tumor that occurs in the breast tissue and is characterised by the uncontrolled growth of abnormal cells, which can invade surrounding tissues and distant organs.

Breast cancer is the most diagnosed cancer affecting women globally, contributing to approximately 25% of all cancer cases and 15% of cancer-related mortality in women. In 2022, there were an estimated 2.3 million new cases, and by 2040, the incidence is projected to be 38% higher [6]. Additionally, breast cancer also imposes a substantial socioeconomic burden, as the estimated annual treatment cost exceeds $20,000 per patient in developed countries and also causes productivity loss due to the illness and mortality [3]. As a result, many research groups have looked to develop newer approaches to predicting patient survival outcomes as early and accurately as possible. Many studies have provided examples of applying machine learning (ML) models for predicting survival for breast cancer, with structured clinical tabular data [4].

Traditional models (Random Forests, Support Vector Machines, Decision Trees, Logistic Regression, and others) are typically known to be strong and interpretable methods to model survival. The models used standard clinical features such as tumor size, lymph node status, age, and hormone receptor status and for most; importance analyses and tumor variables would typically be top variables to aid in survival prediction.

The performance metrics of chosen models are evaluated using accuracy, sensitivity, specificity, area under the curve (AUC), and concordance index. The presence of missing data is typically addressed with different imputation methods. Despite these advancements, many widely-used ML models are not inherently designed to accommodate censored survival data, which may introduce bias and undermine the validity of clinical inferences. Deep learning (DL) techniques, although extensively used in imaging and genomic domains, remain underexplored in the context of tabular clinical data.

In this study, we address these gaps by applying advanced approaches that have shown strong potential in other domains but remain underutilized in survival prediction using tabular clinical data.

Advanced architectures such as TabNet and Deeptable, which have demonstrated strong potential for handling tabular data effectively, have yet to be fully leveraged for survival analysis in

clinical settings. Specifically, we incorporate gradient boosting algorithms such as LightGBM and CatBoost due to their ability to handle complex feature interactions and missing values efficiently.

Furthermore, we employ ensemble methods, including voting and stacking classifiers, to leverage the strengths of multiple models and improve overall predictive performance and generalizability.

2 Materials and Methods

2.1 Dataset

The dataset we used in this study is provided by the Molecular Taxonomy of Breast Cancer International Consortium (METABRIC), which contains clinical and genomic data from 1,980 primary breast cancer samples. The data was collected by Professor Carlos Caldas of the Cambridge Research Institute and Professor Sam Aparicio of the British Columbia Cancer Centre, and was published in Nature Communications [5].The dataset has been used to train and evaluate models under three experimental conditions: using only clinical attributes, only genetic attributes, and a combination of both .Clinical and genetic data were accessed via cBioPortal[2]. The outcome variable of interest is overall survival, represented as a binary label (survived or not survived). The grouped attributes were used as input for a binary classification task to predict overall survival. The dataset was split to train and test the models with a 67%-33% training-test ratio. K-Fold Cross-Validation was used during training to ensure generalizability.The data was split into 5 folds for cross-validation. Model evaluation was based on standard classification metrics, including the F1-score, recall, precision, and overall accuracy.For interpretability, we additionally consider a fixed 5-year (60-month) survival horizon. Patients who died within 60 months were labeled as non-survivors, while patients alive beyond 60 months were labeled as survivors.

2.1.1 Clinical Attributes

Below given are the clinical attributes used with their respective datasets. They comprise of object, float, and integer data types. These clinical attributes are used to train our models under study. Clinical Attributes with their Type and description are explained below. In our first case we trained our models with the below clinical attributes.

2.1.2 Genetic Attributes

The genetic attributes in the dataset include mRNA expression levels and mutation data. It provides mRNA z-scores for 331 genes and mutation information for 175 genes. In our second case we have trained our models with genetic attributes.

2.1.3 Combination

In our third case, we trained our models with both Genetic Attributes and Clinical Attributes.

2.1.4 Exploratory Data Analysis

Relationship between Clinical Attributes and Outcomes The distributional analysis of clinical variables shows us patterns and relationships relevant to the patient outcomes. Age at diagnosis demonstrates an approximately normal distribution, with a central tendency between 50 and 70 years. Patients who died tend to be older at the time of diagnosis, suggesting age may be associated with poorer prognosis. In contrast, variables such as `lymph_nodes_examined_positive`, `mutation_count`, and `tumor_size` exhibit strong right-skewness, characterized by a high concentration of lower values and the presence of significant outliers. For example, most patients had few or no positive lymph nodes, but those with elevated counts were more likely to be in the non-survivor group, underscoring the prognostic value of lymph node involvement. Similarly, although mutation_count is heavily skewed with a long tail, the distributions between survivors and non-survivors are largely overlapping, indicating that mutation burden alone may not be a strong discriminator of survival in this cohort. The Nottingham Prognostic Index (NPI) displays a roughly symmetrical, possibly multimodal distribution, with survivors showing greater density in lower index ranges and non-survivors skewed toward higher values. This observation aligns with its established clinical utility as a composite prognostic tool. Overall survival time (in months) also follows a right-skewed distribution, with survivors displaying a broader and more extended range of survival durations compared to those who died, who are concentrated in shorter survival intervals. Lastly, tumor size is markedly right-skewed, and larger tumors are more frequently observed among non-survivors, suggesting its role as a critical prognostic indicator. In this study, survival was primarily framed as a binary classification problem (alive vs. deceased at last follow-up) to enable a clear comparison between traditional machine learning and deep learning models on tabular data. While the METABRIC dataset supports time-to-event analysis with right-censoring, this simplified formulation was chosen to keep the modeling approach accessible and reproducible at the secondary-school level. As a result, all classification-based results should be interpreted as relative risk signals rather than clinical survival estimates.

Relationship Between Genetic Attributes and Outcomes The histogram presents the distribution of correlation coefficients between individual gene expression levels and overall survival. Here, survival labels were defined for each sample based on patient status (alive = 1, dead = 0), using the METABRIC clinical data to link genetic attributes with outcomes. The majority of correlations cluster around zero, indicating that most genes show weak linear association with survival outcomes. The distribution appears slightly skewed to the left, with a large amount of genes exhibiting negative correlations. This suggests a subset of genes whose increased expression may be associated with reduced survival, potentially pointing to oncogenic roles. Conversely, fewer genes display stronger positive correlations with survival, indicating possible tumor suppressive or protective effects. Survival was framed as a binary classification problem, with labels based solely on patient status (alive = 1, dead = 0), without considering overall survival time. Thus, no explicit prediction horizon (e.g., 5-year survival) was defined, and right-censoring information in the METABRIC dataset was not utilized.

Attribute	Type	Description
patient_id	object	This is used to uniquely identify a sample or patient
age_at_diagnosis	float	Age of the patient at diagnosis time
type_of_breast_surgery	object	Breast cancer surgery type: MASTECTOMY (removal of all breast tissue) or BREAST CONSERVING (removal of only cancerous part)
cancer_type	object	Breast cancer type: Breast Cancer or Breast Sarcoma
cancer_type_detailed	object	Detailed breast cancer type: Invasive Ductal Carcinoma, Mixed Ductal and Lobular Carcinoma, Invasive Lobular Carcinoma, Invasive Mixed Mucinous Carcinoma, Metaplastic Breast Cancer
cellularity	object	Cancer cellularity post-chemotherapy (amount and arrangement of tumor cells)
chemotherapy	int	Whether the patient had chemotherapy (yes/no)
pam50_+_claudin-low_subtype	object	Tumor profiling subtype based on gene expression (e.g., claudin-low, EMT characteristics)
cohort	float	Cohort group (values from 1 to 5)
her2_status_measured_by_snp6	object	HER2 status by advanced molecular techniques (e.g., SNP6)
her2_status	object	HER2 status (positive/negative)
tumor_other_histologic_subtype	object	Cancer subtype based on microscopic examination (e.g., Ductal/NST, Lobular, Metaplastic)
hormone_therapy	int	Whether the patient had hormone therapy (yes/no)
inferred_menopausal_state	object	Menopausal state (post/pre)
integrative_cluster	object	Molecular subtype based on gene expression (e.g., 4ER+, 7, 10)
primary_tumor_laterality	object	Tumor location (right breast or left breast)
lymph_nodes_examined_positive	float	Number of lymph nodes positive for cancer
mutation_count	float	Number of relevant gene mutations
nottingham_prognostic_index	float	Prognostic score based on tumor size, lymph nodes involved, and tumor grade
oncotree_code	object	Standardized cancer type code from OncoTree
overall_survival_months	float	Months from intervention to death
overall_survival	object	Target variable: whether patient is alive or dead
pr_status	object	Progesterone receptor status (positive/negative)
radio_therapy	int	Whether the patient had radiotherapy (yes/no)
3gene_classifier_subtype	object	Three Gene Classifier subtype (e.g., ER-/HER2-, ER+/HER2- High Prolif, HER2+)
tumor_size	float	Tumor size measured via imaging
tumor_stage	float	Cancer stage based on spread and lymph node involvement

Figure 1: Clinical Attributes

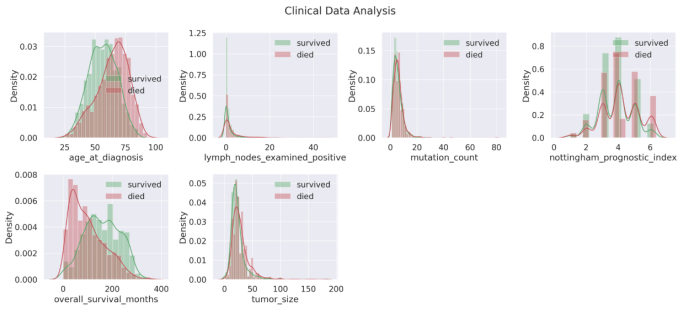


Figure 2: Distribution of clinical features showing key differences between survivors and non-survivors

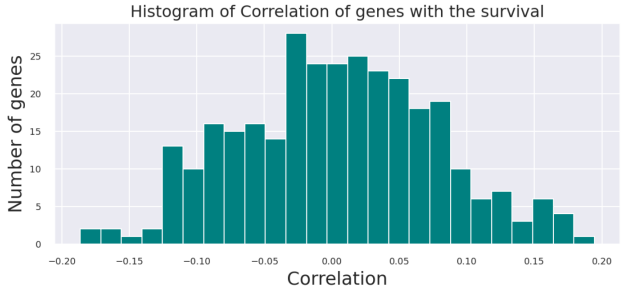


Figure 3: Distribution of correlation coefficients between gene expression levels and overall survival, showing that most genes have weak correlations, with a slight skew toward negative associations.

2.2 Machine Learning Models Used

We have used the below 10 machine learning models to train and evaluate with the datasets provided by METABRIC in 3 case studies with clinical, genomic and a combination of both.

K-Nearest Neighbors (KNN): This is a non-parametric approach that classifies data points based on the majority label of their k nearest neighbors in the feature space.

Decision Tree: The Decision tree model splits data based on feature values to make decisions. Features are indicated by internal nodes, and class labels or values are indicated by leaf nodes.

Logistic Regression: It is a linear classification model predicting the probability of a binary outcome through the logistic (sigmoid) function. This model performs well with linearly separable classes.

Random Forest: Random Forest is an ensemble model that builds multiple decision trees on different data and feature

subsets. It averages their outputs to improve accuracy and avoid overfitting.

Extra Trees (Extremely Randomized Trees): Similar to Random Forest with yet more randomness while choosing feature splits. It builds uncorrelated trees faster and sometimes results in better generalization.

AdaBoost (Adaptive Boosting): It is a boosting ensemble method that trains a series of weak learners, with each subsequent model placing greater emphasis on the errors of the previous models.

XGBoost: An extremely effective gradient boosting library that uses regularization and aggressive tree building algorithms. efficiency and speed on structured data.

CatBoost: It is a gradient boosting framework that is designed to work with categorical features natively with minimal preprocessing. It uses ordered boosting and has overfitting resistance.

Light Gradient Boosting Machine (LightGBM): This model is a fast, distributed gradient boosting method that uses histogram-based algorithms and grows tree's leaf-wise which makes it ideal for large data and high performance.

Support Vector Classifier (SVC): A model that attempts to find the optimal hyperplane to separate classes. It can deal with both linear and non-linear classification using kernel functions.

2.3 Deep Learning Models Used

We have used the below 6 Deep learning models to train and evaluate with the datasets provided by METABRIC in 3 case studies with clinical, genomic and a combination of both.

AutoInt: It uses self-attention mechanisms to automatically learn feature interactions, suitable for modeling complex relationships in tabular data.

DeepFM: It combines Factorization Machines for low-order feature interactions with deep neural networks for high-order interactions, convenient in recommendation and structured data tasks.

xDeepFM: This model is an enhanced version of DeepFM that uses a Compressed Interaction Network (CIN) for explicitly modeling high-order feature interactions in an efficient way.

Wide & Deep: A hybrid model that includes a linear (wide) component for memorization and a deep neural network for generalization, providing a balance of both types of learning.

MLP (Multilayer Perceptron): A simple feedforward neural network composed of dense layers. Learns complex non-linear relationships from input data through backpropagation.

TabNet: This is a tabular data deep learning model. It uses attention to select relevant features at each decision step and provides interpretability along with good performance.

2.4 Ensemble Method

We chose to study ensemble learning methods to improve the accuracy since ensemble methods have recently been praised with strong and universal findings supporting their utility and promise. Ensemble methods work by combining varying base learners to promote generalization, reduce overfitting, and use all data, including unbalanced or noisy data. Ensemble methods can also aid in balancing the bias-variance trade-off, which is an important consideration to make as we leave the safe confines of data, and into data that is likely to include a lot of noise and variability. In this case, as base learners only the three models with the best performance from every training attribute set were included. This ensured that the ensemble consisted of only good-quality models which contained relationship relevant to particular attributes subsets. This approach included only those models that had demonstrated strong performance, thereby maximizing the overall predictive power and variability in the pool of predictive models. As a first step, a Voting Classifier was used, specifically with soft voting. Soft voting computes the predicted probabilities of the models and uses this information to make selections based on a greater understanding of model predictions and class probabilities overall. This was particularly helpful in cases whereby models made conflicting predictions but used different logic in predicting classes [1]. For clinical attributes, LightGBM, CatBoost, and AdaBoost were used. For genetic attributes, CatBoost, K-Nearest Neighbors (KNN), and Random Forest were chosen. When combining both clinical and genetic attributes, LightGBM, CatBoost, and XGBoost were employed. Despite the positive outcome from soft voting, a Stacking Classifier is employed to enhance overall performance. Stacking differs from voting in that it uses a meta-learner to learn the best way to combine the outputs of the base models. The same models selected for Voting classifiers were used for the Stacking classifier.

3 Experiments and Results

3.1 Implementation and Training Details

KNN: The KNN model was tuned using GridSearchCV with 4-fold cross-validation, testing different values of `n_neighbors` ranging from 5 to 100 and using both `uniform` and `distance` weights.

Logistic Regression: Logistic Regression was tuned using GridSearchCV with 4-fold cross-validation, using both `l1` and `l2` penalties and 100 log-spaced values for the regularization parameter `C`. The random state was set to 42.

Decision Tree: The Decision Tree classifier was used with default parameters, and the random state was set to 42.

Random Forest: The Random Forest classifier was implemented using default settings, with the random state set to 42.

Extra Trees: The Extra Trees classifier was used with default parameters, and the random state was set to 42.

AdaBoost: The AdaBoost classifier was used with default parameters, and a random state of 42 was specified.

SVC (Support Vector Classifier): The SVC model was used with default parameters, which include an RBF kernel, `C = 1.0`, and `gamma = scale`. The random state was set to 42.

XGBoost : The XGBoost classifier was initialized with a learning rate of 0.1, 1000 estimators, a maximum depth of 5, and a minimum child weight of 1. The `gamma` value was set to 0, `subsample` and `colsample_bytree` were both set to 0.8, and the objective was set to `binary:logistic`. The model

used 4 threads, a scale position weight of 1, and a random seed of 27.

DeepTable Models: Four different DeepTable models were used: AutoInt, DeepFM, xDeepFM, and WideDeep. Except for changing the `nets` parameter to specify the model type, all other configuration settings were kept the same. These included automatic handling of categorical and discrete features, an embedding output dimension of 20, an embedding dropout rate of 0.3, and evaluation metrics set to AUC and accuracy.

CatBoost : The CatBoost classifier was configured with 500 iterations, a learning rate of 0.03, a maximum depth of 6, and an L2 leaf regularization value of 5. It used `Bernoulli` bootstrap type, a subsample rate of 0.8, and specified `cat_features` for handling categorical variables. The evaluation metric was set to AUC, with early stopping after 20 rounds and verbose output every 50 iterations.

LightGBM : The LightGBM model was configured with the objective set to `binary` and the evaluation metric set to `auc`. It used the `gbdt` boosting type, a learning rate of 0.03, 31 leaves, and a maximum depth of 6. The `feature_fraction` and `bagging_fraction` were both set to 0.8, with `bagging_freq` set to 5. L2 regularization (`reg_lambda`) was set to 5, and the random seed was fixed at 42. The verbosity level was set to -1 to suppress logs.

TabNet : The TabNet classifier was trained using a maximum of 100 epochs with early stopping enabled by setting `patience` to 5. The model was evaluated using the AUC metric.

MLP : The MLP model was trained for up to 100 epochs with a batch size of 32. Early stopping was applied via a callback, and training progress was shown with `verbose` output set to 1.

3.2 Evaluation Metrics

The performance of the models were carefully evaluated, since misclassification has the potential to strongly affect processes in clinical decision-making. Four major metrics were used in the current research: accuracy, precision, recall, and F1 score.

Accuracy: Accuracy is a metric that measures how often a machine learning model correctly predicts the outcome. It is particularly helpful when dealing with balanced classes and when the primary concern is the overall correctness of the model rather than its ability to predict a specific class. However, when classes are imbalanced, accuracy becomes less useful because it gives equal importance to predicting all categories, potentially hiding poor performance on the target class. In such cases, relying solely on accuracy can be misleading. While accuracy indicates the general rate of correct classification, it tends to conceal medical concerns related to class imbalance, i.e., the over representation of survivors (Class 1) in relation to non-survivors (Class 0).

Precision: High precision indicates that when the model predicts a patient has died, it is usually correct. This helps reduce overtreatment, and misallocation of clinical resources. However, prioritizing precision alone may cause the model to miss actual high-risk patients (false negatives), which is why it must be evaluated alongside recall.

Recall: Recall measures the proportion of patients who have passed away and were correctly classified by the model. They are classified as True Positives (TP), i.e. patients who died and were correctly predicted to not survive. False negatives (FN), are patients who died but were incorrectly predicted to survive.Recall helps to minimize False negatives, thereby minimizing the chance of falsely classifying patients.

F1-score: To balance the need to identify at-risk patients with the need to minimize the number of false alarms, we use the F1 score, i.e. the combination of precision and recall. This metric possesses great significance in our binary classification problem, especially when using stacking ensemble methods and combining features from high dimensional genetic data.

3.3 Machine Learning Model Outcomes

The following sections present model outcomes separately for clinical attributes, genetic attributes, and a combination of both. Each subsection includes model-specific performance and insights into which models performed best for that dataset type.

Table 1: Model Performance on Clinical Attributes

Model	Acc	Prec	Rec	F1
K Neighbors	0.66	0.65	0.66	0.65
Logistic Reg	0.77	0.76	0.77	0.76
Decision Tree	0.70	0.69	0.70	0.69
Random Forest	0.73	0.72	0.73	0.72
MLP	0.75	0.74	0.75	0.74
SVC	0.76	0.75	0.76	0.75

Table 2: Model Performance on Genetic Attributes

Model	Acc	Prec	Rec	F1
K Neighbors	0.78	0.77	0.78	0.77
Logistic Reg	0.63	0.62	0.63	0.62
Decision Tree	0.61	0.60	0.61	0.60
Random Forest	0.69	0.68	0.69	0.68
MLP	0.74	0.73	0.74	0.73
SVM	0.71	0.70	0.71	0.70

3.4 Deep Learning Model Outcomes

Below sections elaborate the results obtained by training and evaluating different Deep learning models on Clinical attributes, genetic attributes and the combination of both.

3.5 Ensemble Method Outcomes

3.5.1 Voting Classifier(Soft Voting)

Voting Classifier is the best performing model overall with an accuracy of 80% when trained on clinical attributes.

Table 3: Model Performance on Combined Attributes (Clinical + Genetic)

Model	Acc	Prec	Rec	F1
K Neighbors	0.67	0.66	0.67	0.66
Logistic Reg	0.75	0.74	0.75	0.74
Decision Tree	0.72	0.71	0.72	0.71
Random Forest	0.74	0.73	0.74	0.73
MLP	0.78	0.77	0.78	0.77
SVM	0.77	0.76	0.77	0.76

Table 4: Performance of Deep Learning Models on Clinical Attributes

Model	Prec(0)	Prec(1)	Rec(0)	Rec(1)	F1(0)	F1(1)	Acc
DeepFM	0.83	0.45	0.16	0.95	0.28	0.62	0.50
xDeepFM	0.89	0.49	0.28	0.95	0.42	0.65	0.56
AutoInt	0.58	0.00	1.00	0.00	0.73	0.00	0.58
WideDeep	0.82	0.45	0.15	0.95	0.26	0.61	0.49
MLP	0.78	0.68	0.75	0.72	0.77	0.70	0.74
TabNet	0.62	0.63	0.90	0.25	0.73	0.36	0.62

Table 5: Performance of Deep Learning Models on Genetic Attributes

Model	Prec(0)	Prec(1)	Rec(0)	Rec(1)	F1(0)	F1(1)	Acc
DeepFM	0.61	0.49	0.75	0.33	0.67	0.39	0.57
xDeepFM	0.59	0.46	0.74	0.29	0.66	0.36	0.55
AutoInt	0.58	0.00	1.00	0.00	0.73	0.00	0.58
WideDeep	0.62	0.56	0.84	0.28	0.71	0.37	0.60
MLP	0.67	0.59	0.75	0.50	0.71	0.54	0.65
TabNet	0.67	0.41	0.66	0.42	0.64	0.44	0.56

Table 6: Performance of Deep Learning Models on Combined Attributes (Clinical + Genetic)

Model	Prec(0)	Prec(1)	Rec(0)	Rec(1)	F1(0)	F1(1)	Acc
MLP	0.79	0.71	0.81	0.61	0.80	0.70	0.76
TabNet	0.67	0.54	0.35	0.76	0.46	0.56	0.51

3.5.2 Stacking Classifier

This ensemble performed similarly to voting classifier with an accuracy of 79% on clinical attributes.

4 Discussion

4.1 ROC Curve

ROC curves trace the comparisons of different traditional machine learning models. The x-axis is plotted with False Positive Rate, meaning how often patients who did not survive are incorrectly predicted as survivors. The y-axis plots the True Positive Rate, meaning how often actual survivors are correctly predicted.

Clinical		
Metric	**0**	**1**
Precision	0.82	0.77
F1 Score	0.83	0.76
Recall	0.84	0.74
Accuracy: 0.80		
Genetic		
Metric	**0**	**1**
Precision	0.68	0.67
F1 Score	0.75	0.54
Recall	0.84	0.45
Accuracy: 0.67		
Combination		
Metric	**0**	**1**
Precision	0.79	0.75
F1 Score	0.81	0.71
Recall	0.84	0.68
Accuracy: 0.72		

Table 7: Voting Classifier Results

Clinical		
Metric	**0**	**1**
Precision	0.82	0.76
F1 Score	0.82	0.75
Recall	0.83	0.74
Accuracy: 0.79		
Genetic		
Metric	**0**	**1**
Precision	0.68	0.68
F1 Score	0.75	0.53
Recall	0.58	0.44
Accuracy: 0.68		
Combination		
Metric	**0**	**1**
Precision	0.75	0.69
F1 Score	0.77	0.66
Recall	0.79	0.64
Accuracy: 0.72		

Table 8: Stacking Classifier Results

The area under the curve (AUC) measures how well the model can distinguish between classes (0 and 1). A larger area (closer to 1) means the model is more accurate, while an area close to 0.5 suggests random guessing.

4.1.1 ROC Curve for Traditional Machine Learning Models

Logistic Regression classifier performed the best among traditional models when trained on clinical attributes with an accuracy of 77%. It also performed the best on the combination of both attributes with an accuracy of 75%. However, KNN model performed the best with an accuracy of 65% when trained with genetic attributes.

4.1.2 ROC Curve for Deep Learning Models

Here the MLP model has a performance of 72% accuracy when trained on clinical attributes whereas TabNET performs much worse with an accuracy of 57%. Both models performed similarly when trained on genetic attributes where MLP classifier provided an accuracy of 66% and TabNET gave an accuracy of 62%. MLP classifier has an accuracy of 75% when trained on the combination of both attributes whereas TabNET had an overall accuracy of 56%.

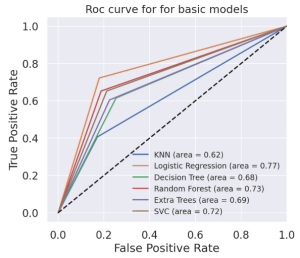


Figure 4: ROC Curve Traditional Models on Clinical Attributes

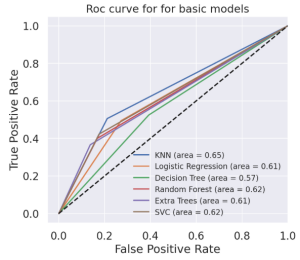


Figure 5: ROC Curve Traditional Models on Genetic Attributes

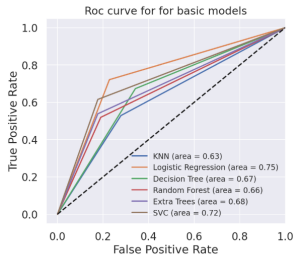


Figure 6: ROC Curve Traditional Models on Combination of Both Attributes

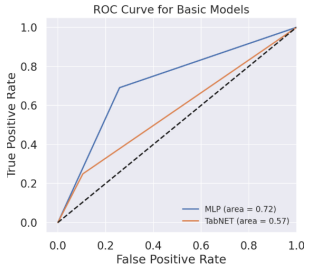


Figure 7: ROC Curve for Deep Learning Models on Clinical Attributes

4.1.3 ROC Curve for Ensemble Models

Here the Voting classifier with soft voting had an accuracy of 80% on clinical attributes and the stacker classifier had an accuracy of 79%. Both models performed similarly when trained on genetic attributes with the same accuracy of 64%. The stacker classifier and

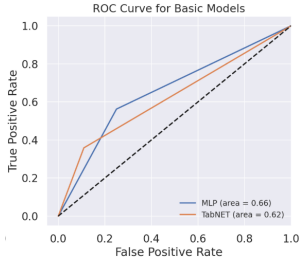


Figure 8: ROC Curve for Deep Learning Models on Genetic Attributes

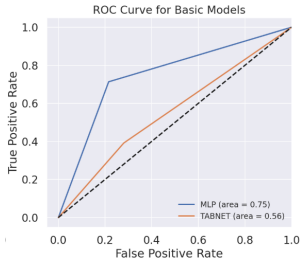


Figure 9: ROC Curve for Deep Learning Models on Combination of Both Attributes

the soft voting classifier had an accuracy of 72% when trained on the combination of attributes.

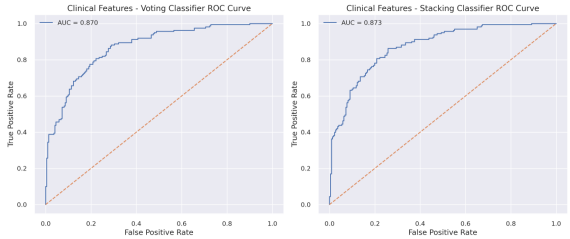


Figure 10: ROC Curve for Ensemble Models on Clinical Attributes

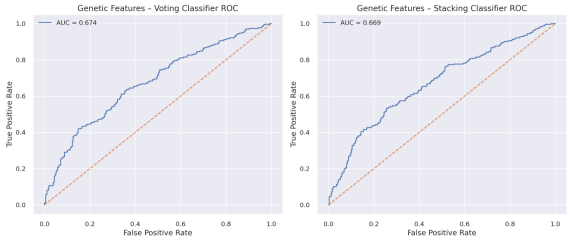


Figure 11: ROC Curve for Ensemble Models on Genetic Attributes

4.2 Comparative Performance

When trained on clinical attributes, a soft voting classifier ensemble with CatBoost, AdaBoost, and LightGBM achieved the highest accuracy of 80%. When the models were trained on the genetic

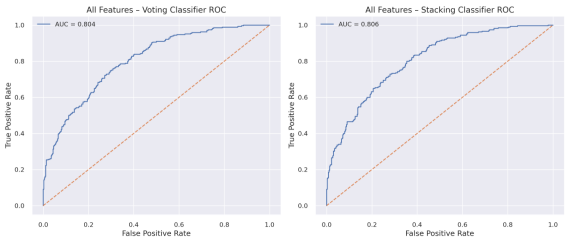


Figure 12: ROC Curve for Ensemble Models on Combination of Both Attributes

attributes, the best performance of 68% accuracy was obtained using a stacking classifier with CatBoost, K-Nearest Neighbors (KNN), and Random Forest. When combining clinical and genetic features, a soft voting classifier with CatBoost, XGBoost, and LightGBM achieved 77% accuracy.

The high performance of gradient boost models could be due to their ability to handle mixed data types and manage missing values. These models are also less likely to overfit on small to medium-sized datasets. Additionally, soft voting classifiers take the average of the predictions and account the confidence of each model. This helps to reduce the impact of a certain model's mistake.

In contrast deep learning models, mainly Deeptable models like AutoInt, DeepFM, AutoInt, xDeepFM and WideDeepand Tabnet, failed to perform adequately. These models either crashed or yielded poor results when trained on combined datasets. Only the MLP Classifier achieved moderate accuracy, but still underperformed compared to the other ensembles. Insufficient data size(such as 1,980 samples in our study), high feature dimensionality (especially in the genetic data), lack of spatial structure are some probable reasons why these models underperformed. Deep models typically require large datasets and are prone to overfit in low-sample settings. The strong performance of CatBoost, LightGBM, and XGBoost suggests that gradient boosting models are better suited for tabular limited data. These models natively support categorical variables, feature importance evaluation, and efficient training on small datasets, making them highly effective in binary classification tasks using limited data.

5 Limitations of Binary Classification and Transition to Time-to-Event Analysis

The above study using binary classification has significant limitations when applied to clinical datasets such as METABRIC, which contain long follow-up times and right-censoring.

Binary classification discards two key aspects of survival data:

(1) **Censoring:** Patients who remain alive at last follow-up are treated as if their outcome were fully observed, leading to biased estimates.
(2) **Timing of events:** Information on when an event occurs is lost, reducing the clinical interpretability of the predictions.

These issues mean that classification metrics such as accuracy or F1-score can be misleading for risk prediction, as they fail to capture time-to-event dynamics that are crucial.

To address this, the task was reformulated as a censored time-to-event analysis. This allowed full utilization of the survival information in the dataset, estimation of individual risk over time, and evaluation of models with survival-aware metrics such as the concordance index (C-index), time-dependent AUC, Brier score, and calibration curves.

5.1 Genetic Data Preprocessing

The genetic pipeline first processed high-dimensional genomic features, including mRNA expression and mutation data, using the Elastic Net–regularized Cox model to identify stable, survival-associated signatures. This approach was chosen for its ability to handle multicollinearity among correlated genetic features and to enforce sparsity, enabling the selection of the most informative genes. These selected genetic features were then fused with clinical variables such as age, tumor stage, treatment history, and overall survival time (in months).

The survival labels were derived from the overall survival time (in months) and the vital status recorded in the dataset. Patients who were alive at the last follow-up were treated as censored, while those who had an event (death) were marked as uncensored.

5.1.1 Traditional Models Used for Survival Analysis

Regularized Cox Proportional Hazards (CoxPH) Model: A semi-parametric approach that models the effect of covariates on survival, with regularization to handle high-dimensional data.

Random Survival Forests (RSF): An ensemble tree-based method that non-parametrically estimates survival probabilities while automatically handling non-linear interactions and high-dimensional data.

5.1.2 Deep Learning Models Used for Survival Analysis

DeepSurv: A deep learning extension of the Cox model that captures complex, non-linear relationships between covariates and survival risk.

DeepHit: A neural network model that directly estimates the probability distribution of survival times, allowing flexible handling of competing risks and time-dependent hazards.

5.2 Results

5.2.1 Regularized CoxPHFitter

Reports the C-index, time-dependent AUC, and Brier scores at 12, 36, and 60 months for Clinical, Genetic, and combined feature sets. Training C-index is provided for Genetic and Both attributes. All metrics are calculated on the test set unless otherwise specified.

5.2.2 DeepSurv

Reports the C-index and Brier scores at 12, 36, and 60 months for Clinical, Genetic, and combined feature sets. All metrics are calculated on the test set.

5.2.3 DeepHit

Reports the C-index and Brier scores at 12, 36, and 60 months for Clinical, Genetic, and combined feature sets. All metrics are calculated on the test set.

Table 9: Performance of Regularized CoxPH Model Across Different Feature Sets

Metric	Clinical	Genetic	Both
C-index (Train)	–	0.7404	0.7435
C-index (Test)	0.8131	0.7227	0.7321
AUC 12 mo (Test)	0.8896	0.7190	0.6957
AUC 36 mo (Test)	0.8637	0.7318	0.7099
AUC 60 mo (Test)	0.8743	0.7238	0.7225
Brier 12 mo (Test)	0.0082	0.0091	0.0092
Brier 36 mo (Test)	0.0172	0.0158	0.0159
Brier 60 mo (Test)	0.0272	0.0247	0.0248

Table 10: DeepSurv Model Performance Across Different Feature Sets

Metric	Clinical	Genetic	Both
C-index	0.7159	0.6375	0.6375
Brier 12 mo	0.9498	0.9959	0.9959
Brier 36 mo	0.9796	0.9816	0.9816
Brier 60 mo	0.9735	0.9685	0.9685

Table 11: DeepHit Model Performance Across Different Feature Sets

Metric	Clinical	Genetic	Both
C-index	0.3988	0.3207	0.3333
Brier 12 mo	0.0085	0.0084	0.0085
Brier 36 mo	0.0235	0.0233	0.0231
Brier 60 mo	0.0426	0.0423	0.0421

5.2.4 Random Survival Forest (RSF)

Reports the C-index, Brier scores, AUC at multiple time points, and mean time-dependent AUC for Clinical, Genetic, and combined feature sets. All metrics are calculated on the test set.

Table 12: RSF Model Performance Across Different Feature Sets

Metric	Clinical	Genetic	Both
C-index	0.8120	0.6876	0.7944
Brier 12 mo	0.0074	0.0079	0.0077
Brier 36 mo	0.0173	0.0189	0.0184
Brier 60 mo	0.0263	0.0281	0.0272
AUC 12 mo	0.8770	0.8183	0.9384
AUC 24 mo	0.8289	0.7257	0.9021
AUC 36 mo	0.8356	0.7320	0.8993
AUC 48 mo	0.8459	0.7353	0.8864
AUC 60 mo	0.8729	0.7250	0.8900
Mean time-dependent AUC	0.8520	0.7473	0.9033

5.3 Calibration Plots

The following calibration plots illustrate the agreement between predicted survival probabilities and observed outcomes for each model across the three feature sets: Clinical, Genetic, and Combined. A perfectly calibrated model would align closely with the diagonal reference line.The calibration plots of the best performing deep learning model and machine learning model from the binary classification task(CatBoost and TabNet) are shown here.

Table 13: C-index of top classifier and CoxPH model on clinical, genetic, and combined features

Model	Clinical	Genetic	Both
Regularized CoxPh	0.8131	0.7227	0.7321
CatBoost	0.5209	0.3653	0.3737

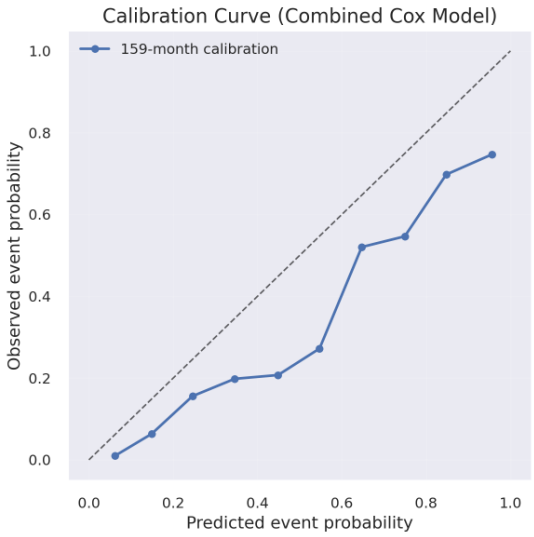


Figure 13: Calibration plot for Regularized CoxPHFitter using the combined clinical and genetic feature set.

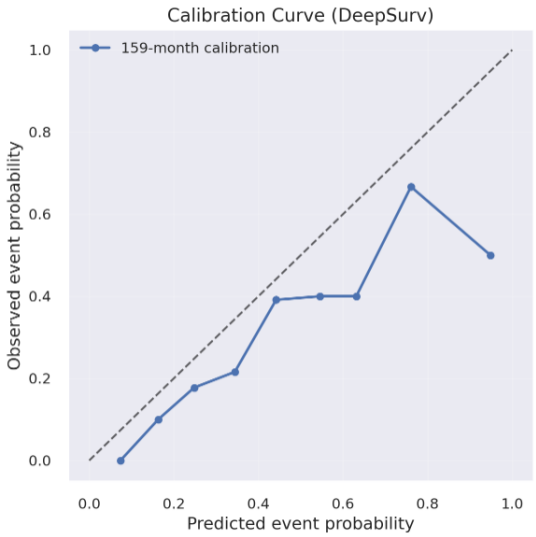


Figure 14: Calibration plot for DeepSurv using the combined clinical and genetic feature set.

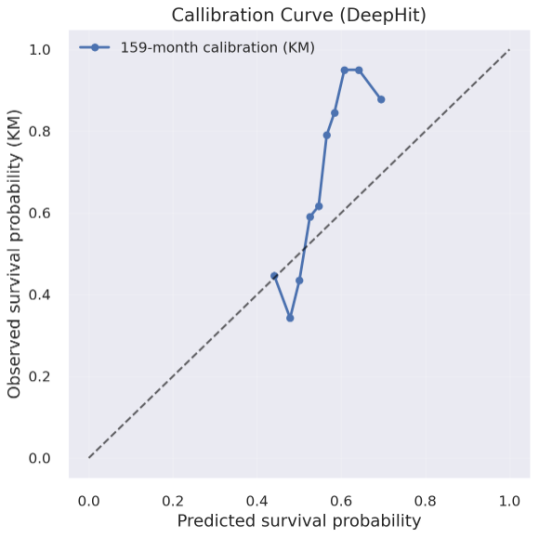


Figure 15: Calibration plot for DeepHit using the combined clinical and genetic feature set.

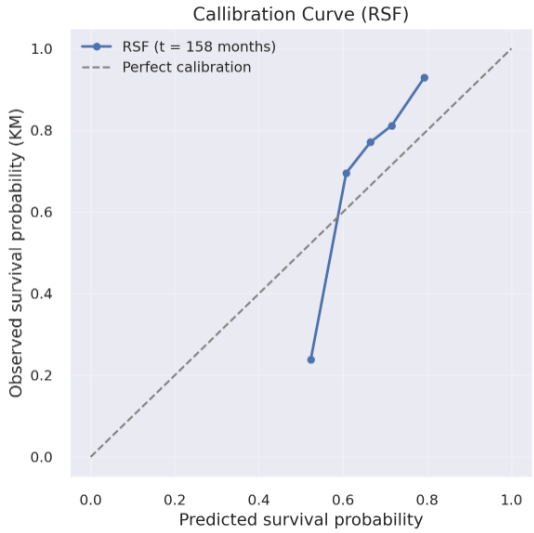


Figure 16: Calibration plot for DeepHit using the combined clinical and genetic feature set.

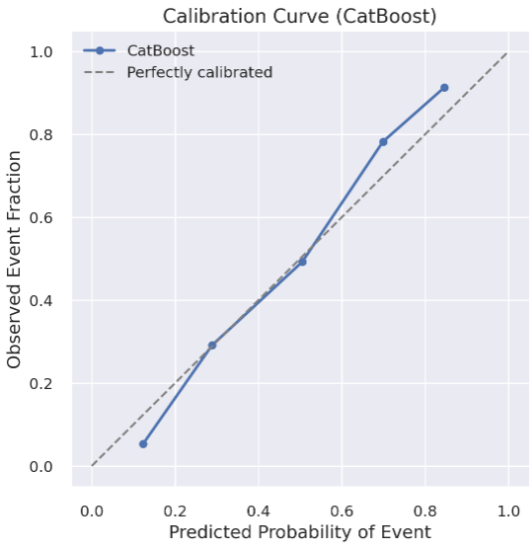


Figure 17: Calibration plot for CatBoost using the combined clinical and genetic feature set.

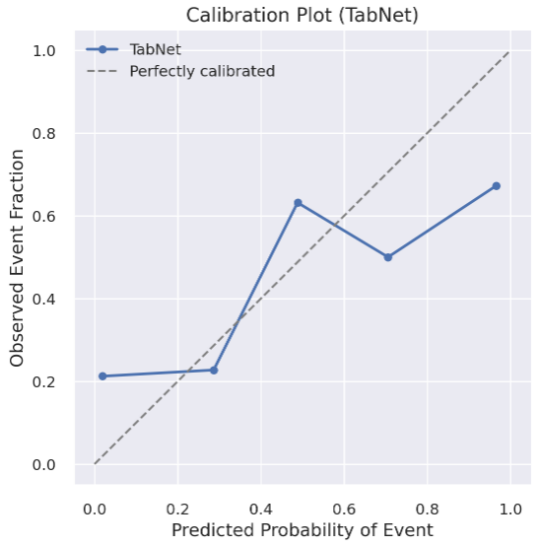


Figure 18: Calibration plot for Tabnet using the combined clinical and genetic feature set.

6 Conclusions

This study demonstrates that ensemble methods based on gradient boosting models provide satisfactory accuracy in predicting breast cancer survival from structured clinical and genomics attributes. The best performance (80% accuracy) was achieved in a soft voting ensemble on clinical features. The genetic attributes even at its high dimensionality turned out to be reasonably well predicted (68% accuracy) with stacking classifiers. With the combined data, the gradient boosting voting ensemble obtained an accuracy of 77%. Deep learning-based methods however, e.g., DeepFM and AutoInt, did not perform well showing an accuracy of 67% and were not effective for predicting due to data size and complexity.

In addition to these classification models, time-to-event models were evaluated to account for survival time and censoring. The Cox proportional hazards model on clinical features achieved a C-index of 0.813, AUCs of 0.890, 0.864, and 0.874 at 12, 36, and 60 months, and Brier scores of 0.008, 0.017, and 0.027, indicating strong discrimination and excellent calibration. For combined clinical and genetic features, DeepSurv and DeepHit achieved C-index values around 0.835, with similarly high AUCs, demonstrating improved

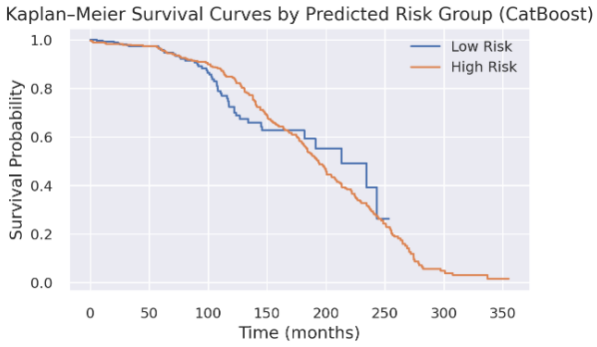


Figure 19: Kaplan–Meier survival curves for patients stratified into high-risk and low-risk groups based on the median predicted risk from the CatBoost model.

risk stratification and capturing non-linear interactions between features. These results highlight the added value of survival analysis models in providing time-dependent predictions beyond binary classification.

Future work could focus on selecting the most important features from omics data and reducing its complexity, and creating hybrid models that combine deep learning with tree-based methods. Testing on larger datasets could help improve how well the models work in real-world settings.This study prioritizes methodological accessibility over clinical deployment. While more advanced survival models and evaluation techniques may further improve predictive performance, they were outside the intended scope of this secondary-school research project.

7 Acknowledgement

I would like to thank my mentor, Pramit Saha for his support and collaboration throughout this work. His guidance, feedback, and helpful suggestions were invaluable in shaping the paper and bringing it to completion.

References

[1] Baeldung: Hard vs. soft voting classifiers (2024), https://www.baeldung.com/cs/hard-vs-soft-voting-classifiers, retrieved from Baeldung
[2] cBioPortal for Cancer Genomics: Metabric dataset (2023), https://www.cbioportal.org/, retrieved from cBioPortal
[3] International Agency for Research on Cancer (IARC): Global cancer observatory (gco): Cancer today (2024), accessed via IARC Global Cancer Observatory website; data from https://gco.iarc.fr
[4] Javanmard, Z., Zarean Shahraki, S., Safari, K., Omidi, A., et al.: Artificial intelligence in breast cancer survival prediction: A comprehensive systematic review and meta-analysis. Frontiers in Oncology **14** (2025). https://doi.org/10.3389/fonc.2024.1420328, https://doi.org/10.3389/fonc.2024.1420328
[5] Pereira, B., et al.: The metabric dataset: A comprehensive resource for cancer survival analysis. Nature Communications **7**, 11551 (2016). https://doi.org/10.1038/ncomms11551, https://www.nature.com/articles/ncomms11551
[6] World Health Organization: Breast cancer statistics (2022), https://www.who.int/news-room/fact-sheets/detail/breast-cancer, wHO fact sheet; contains global statistics, risk factors, symptoms, and treatment.

Received 05 September 2025; Accepted 28 January 2026