Degree project

Evaluating Differences in Machine Learning Models: A Comparison of Cross-Validation and Train/Test Split in Breast Cancer Gene Expression Data

Bachelor Degree Project in Bioinformatics G2E

First Cycle 30 credits

Spring semester 2026

Student: Aneeqa Nadeem

Co-supervisor: Ryan Potter

Supervisor: Benjamin Ulfenborg

Examiner: Angelica Lindlöf

Abstract

Machine learning is widely used to analyse breast cancer gene expression data, but the reliability of model evaluation varies across studies. The choice of validation strategy can affect how trustworthy, reported model performance appears, making a direct comparison under identical conditions valuable. This study examined whether five-fold cross-validation produces different performance estimates compared with a repeated independent train/test split, using *ESR2* (estrogen receptor beta gene) expression as the classification target. *ESR2* was chosen because its expression is generally low and its clinical importance in breast cancer is still not fully understood, unlike the related gene *ESR1*, which is well established. Gene expression data were obtained from the Sweden Cancerome Analysis Network – Breast (SCAN-B) cohort (GSE202203), comprising 3,207 primary breast cancer tumour samples profiled across 19,644 genes. Samples were classified as *ESR2*-high or *ESR2*-low using a tertile cutoff, producing a balanced dataset of 2,116 samples. Three classifiers were evaluated: LASSO-regularised logistic regression, Random Forest, and a tuned support vector machine (SVM). Preprocessing, including low-expression filtering, log2 transformation, variance-based feature selection, and z-score scaling, was performed exclusively on training data to prevent information leakage. Test accuracy ranged from 0.813 to 0.825 and AUC from 0.8903 to 0.8982 across models. The difference in mean AUC between five-fold cross-validation and the train/test split was small for all three models: 0.0012 for LASSO, −0.0012 for Random Forest, and 0.0033 for SVM. Unpaired t-tests showed that none of these differences were statistically significant (LASSO $p = 0.767$, Random Forest $p = 0.751$, SVM $p = 0.449$). Overall, five-fold cross-validation and a repeated train/test split produced broadly similar performance estimates in this dataset, providing no strong evidence of systematic evaluation bias. For this particular dataset and experimental setup, five-fold cross-validation and repeated independent train/test splitting produced comparable estimates of model performance.

Popular Science Summary

Every year, hundreds of thousands of women are diagnosed with breast cancer. One of the big challenges in cancer research is figuring out why some tumours respond well to treatment while others don't. Researchers can now measure the activity of thousands of genes in a tumour at the same time, which gives a huge amount of information about that person's cancer. The problem is that this much data is difficult to make sense of on its own. Machine learning is one way to deal with this, since it allows a computer to learn patterns from existing patient data and use those patterns to make predictions about new patients.

This study focused on a gene called *ESR2*, which makes a protein involved in how cells respond to the hormone oestrogen. A related gene, *ESR1*, is already well understood and plays an important role in deciding how some breast cancer patients are treated. Much less is known about *ESR2*, and some studies suggest it may also be linked to how a tumour behaves, although the evidence so far has been mixed. In this study, machine learning models were trained to sort patients into two groups: those with high *ESR2* activity and those with low *ESR2* activity, based on their gene expression data.

Before a machine learning model can be trusted, it has to be tested properly. There is more than one way to do this. One way is a bit like a practice exam, where the data are split into smaller chunks and the model is tested on each chunk in turn. Another way is more like a final exam, where the model is trained on one part of the data and then tested on a separate part it has not seen before. This study compared these two testing methods to see whether they gave different estimates of model performance.

Both methods were tested using gene expression data from more than 3,000 breast cancer patients in Sweden, using three different machine learning models. The results were similar across all three models and both testing methods. The differences between the two approaches were small and did not change the overall conclusions. For this dataset and sample size, the results suggest that either testing approach could be used without changing the overall conclusions.

List of abbreviations

AUC	Area Under the Curve
CV	Cross-Validation
ERα	Estrogen Receptor alpha (protein)
ERβ	Estrogen Receptor beta (protein)
ESR1	Estrogen Receptor alpha (gene)
ESR2	Estrogen Receptor beta (gene)
GEO	Gene Expression Omnibus
GSE202203	GEO Series accession number
LASSO	Least Absolute Shrinkage and Selection Operator
ML	Machine Learning
NCBI	National Center for Biotechnology Information
RNA-seq	RNA Sequencing
ROC	Receiver Operating Characteristic
SCAN-B	Sweden Cancerome Analysis Network – Breast
SVM	Support Vector Machine
TPM	Transcripts Per Million

Table of Content

1 Introduction 5

2 Aim and Objectives 7

3 Materials and Methods 8

3.1 Dataset 8
3.2 Extraction of *ESR2* Expression 8
3.3 Log2 Transformation of *ESR2* Values 8
3.4 *ESR2* Group Assignment and Dataset Preparation 8
3.5 Removal of *ESR2* to Prevent Data Leakage 8
3.6 Data Splitting 8
3.7 Preprocessing on Training Data 9
3.8 Final Data Formatting 9
3.9 Regularized Logistic Regression Classification 9
3.10 Random Forest Classification 9
3.11 Support Vector Machine (SVM) Classification 9
3.12 Cross Validation 10
3.13 Statistical Analysis 10
3.14 Use of AI Tools 11

4 Results 11

4.1 ESR2 Expression Distribution and Group Assignment 11
4.2 Preprocessing and Feature Selection 12
4.3 Model Performance on Independent Test Set 12
4.4 Cross-Validation Performance 13
4.5 Cross-Validation vs Train/Test Performance 14
4.6 Statistical Significance of Evaluation Difference 15

5 Discussion 15

5.1 Limitations 16

6 Conclusions 18

7 Future Perspectives 18

8 Ethics and Societal Relevance 19

References 20

1 Introduction

Machine learning has become an increasingly important tool in biomedical research, particularly in the study of complex diseases such as cancer (Jordan & Mitchell, 2015; Kourou et al., 2015). With advances in next-generation sequencing, transcriptomics, and proteomics, researchers now have access to substantial amounts of biological data (Manzoni et al., 2018). The challenge is no longer data collection but rather transforming these data into meaningful and reliable findings (Jordan & Mitchell, 2015). Breast cancer gene expression datasets illustrate this problem well, typically containing thousands of molecular variables measured across hundreds or even thousands of patient samples (Kourou et al., 2015). This scale makes them difficult to analyse using traditional statistical approaches (Kourou et al., 2015). Consequently, machine learning methods have become widely applied in this field because of their capacity to handle large, high-dimensional datasets where patterns are not easily detectable using conventional methods (Jordan & Mitchell, 2015).

In cancer research, machine learning models are commonly applied for tumour classification, prognosis prediction, and biomarker discovery (Tran et al., 2021; Kourou et al., 2015). Gene expression profiles are especially useful because they provide insight into the molecular mechanisms driving tumour behaviour and disease progression (Kourou et al., 2015). At the same time, cancer datasets present several methodological challenges. They are biologically heterogeneous, often imbalanced, and typically contain far more variables than patient samples (Kourou et al., 2015). Under these conditions, models can readily overfit, learning noise rather than meaningful biological relationships (Hastie et al., 2009).

Overfitting occurs when a model memorises patterns specific to a given dataset rather than learning relationships that generalise to new patient data (Hastie et al., 2009). As a result, a model may appear highly accurate during training yet perform poorly when tested on independent samples (Hastie et al., 2009). This is why model evaluation is so critical in biomedical machine learning. The choice of evaluation method can substantially alter how trustworthy the final conclusions appear (Kohavi, 1995).

The two most common validation strategies are the train/test split and cross-validation (Kohavi, 1995). In a train/test split, the data is divided into separate training and testing subsets, and performance is measured on data the model has not seen (Kohavi, 1995). Cross-validation repeatedly partitions the dataset into folds, training and testing the model several times before averaging the results (Kohavi, 1995). This tends to produce more stable estimates than a single split, particularly when sample sizes are small, although with a larger dataset the two approaches would be expected to agree more closely (Kohavi, 1995). Both strategies are commonly used in transcriptomic studies, which is why a direct comparison under identical conditions addresses a more practically relevant question than testing against less common alternatives (Vabalas et al., 2019; Kohavi, 1995). Cross-validation does not necessarily produce the same performance estimates under all circumstances (Varma & Simon, 2006), and two models trained on the same data can appear to differ simply because of how the folds were constructed (Varma & Simon, 2006).

Another important issue is data leakage. If steps like feature selection or scaling are done using the whole dataset before splitting the data, information from the validation data can affect the training process and make the model look better than it really is (Kaufman et al., 2012). This can happen with both validation methods used in this study, so preprocessing was done carefully to avoid it.

Machine learning has been applied to breast cancer gene expression data for decades. Sørlie et al. (2003) used hierarchical clustering to identify distinct breast cancer subtypes with different clinical outcomes, establishing that transcriptomic profiles carry genuine prognostic information.

Parker et al. (2009) built on this with PAM50, a 50-gene classifier that assigns tumour samples to molecular subtypes and remains widely used in clinical research. More recent studies have applied regularised regression, random forests, and support vector machines to similar datasets, but reported classification accuracies vary considerably depending on the dataset, feature selection strategy, and evaluation method used (Kourou et al., 2015; Tran et al., 2021). Vabalas et al. (2019) showed that inconsistent validation strategies account for much of this variability in biomedical machine learning, making it difficult to determine whether performance differences between studies reflect genuine model quality or simply differences in how models were evaluated.

Despite the widespread use of both validation approaches, relatively few studies have directly compared them under the same experimental conditions using large-scale breast cancer gene expression datasets (Vabalas et al., 2019). Many published studies evaluate these methods separately or rely on simulated data rather than applying both strategies within the same modelling framework (Vabalas et al., 2019). As a result, it is not always possible to determine whether observed differences in reported model performance reflect genuine model characteristics or simply the choice of evaluation strategy (Kohavi, 1995).

This issue extends beyond methodology. In cancer research, unreliable evaluation methods can lead to misleading conclusions regarding the accuracy or clinical utility of a model, and even modest differences in validation strategy can affect how trustworthy a model appears to both researchers and clinicians (Kapoor & Narayanan, 2023). Machine learning models are increasingly being explored for applications in diagnosis, prognosis, and treatment selection (Tran et al., 2021; Kourou et al., 2015).

Gene expression data for this study were obtained from the SCAN-B cohort (GSE202203), a large uniformly processed dataset of 3,207 primary breast cancer tumour samples with RNA sequencing data across 19,644 genes (Dalal et al., 2022). The relatively uniform processing of the SCAN-B dataset helps reduce technical variation between samples. This is important because differences in data quality or processing can affect performance estimates regardless of the evaluation strategy used (Dalal et al., 2022). To investigate these validation strategies in a biologically relevant context, this study used *ESR2* expression as the classification target. *ESR2* encodes estrogen receptor beta (ERβ), a nuclear receptor that binds oestrogen and controls transcription of target genes (Choi, 2022). ERβ is similar in structure to ERα (encoded by ESR1), but it interacts with different co-regulators, which can affect different genes and lead to different effects on cell behaviour (Choi, 2022). ERα has an established role in breast cancer biology and in decisions about endocrine treatment, whereas the clinical significance of ERβ is less clearly established. Some studies link higher *ESR2* expression to better prognosis in basal-like subtypes, but findings are inconsistent (Dalal et al., 2022). This uncertainty, alongside the gene's generally low expression across breast tumours, is what drew this study to it as a classification target.

This study set out to examine how the choice of validation strategy affects estimates of classification performance in high-dimensional breast cancer gene expression data. Here, evaluation difference is defined as the difference in estimated classification performance, measured using accuracy and AUC, between cross-validation and a single train/test split. To capture this, both strategies were repeated across several random data partitions rather than relying on a single run. A second aim was to examine whether different machine-learning models produce different levels of stability in their performance estimates. Comparing train/test splitting and cross-validation under otherwise identical conditions allows the influence of the evaluation method on reported performance to be examined. Prior work suggests that cross-validation tends to give more stable performance estimates than a single train/test split, especially in small or high-dimensional datasets (Kohavi, 1995). Following this, cross-validation was expected to show less variability across repeated random partitions than the train/test split. No clear direction was

expected in terms of which method would produce higher performance scores, since both were applied to the same models and data under otherwise identical conditions. It was expected that the choice of model would affect the stability of the results, with Random Forest potentially showing more variation than the two linear models, LASSO and SVM. To reduce the risk of information leakage, any preprocessing step that depended on the data, such as feature selection and scaling, was carried out using only the training data and then applied to the test data. The same rule was followed within each cross-validation fold, with preprocessing parameters estimated from the training portion and then applied to the validation portion. This is important because, in biomedical machine learning, the reliability of a model's results depends not only on the model itself, but also on how its performance is evaluated (Kohavi, 1995; Kapoor & Narayanan, 2023).

2 Aim and Objectives

The aim of this study was to examine whether five-fold cross-validation and an independent train/test split produce different estimates of classification performance, a question raised by prior work on validation bias (Kohavi, 1995; Varma & Simon, 2006), by repeating both methods across several random data partitions, when classifying breast cancer samples according to *ESR2* expression level.

Objectives

- To classify breast cancer patient samples into *ESR2* high and *ESR2* low groups using LASSO regularised logistic regression, Random Forest, and Support Vector Machine, applied to RNA sequencing data from the SCAN-B cohort.
- To evaluate model performance using train/test split and five-fold cross-validation under identical experimental conditions, with preprocessing applied within the training data only.
- To quantify evaluation bias as the difference in accuracy and AUC between cross-validation and an independent train/test split, across all three classifiers.
- To determine whether the performance differences between the two evaluation methods are statistically significant using Welch two-sample t-test.

3 Materials and Methods

3.1 Dataset

All analyses were performed in R (version 4.6.1; R Core Team, 2026). Gene expression data were obtained from the NCBI Gene Expression Omnibus (GEO) database under accession number GSE202203 (Dalal et al., 2022). The dataset comes from the SCAN-B (Sweden Cancerome Analysis Network – Breast) study and contains RNA sequencing data from 3,207 primary breast cancer tumour samples across 19,644 genes. The TPM normalised matrix provided by the SCAN-B study was used for all downstream analyses, as TPM normalisation adjusts for sequencing depth and gene length, making values comparable across samples within this dataset (Wagner et al., 2012; Dalal et al., 2022). Data were loaded using the readr package (version 2.2.0; Wickham et al., 2023).

The following R packages were used: glmnet (version 5.0; Friedman et al., 2010) for regularised logistic regression, randomForest (version 4.7-1.2; Liaw & Wiener, 2002) for random forest classification, e1071 (version 1.7-17; Meyer et al., 2023) for support vector machine classification, caret (version 7.0-1; Kuhn, 2008) for cross-validation, pROC (version 1.19.0.1; Robin et al., 2011) for AUC calculation, and ggrepel (version 0.9.8; Slowikowski, 2023) for plot label placement.

3.2 Extraction of *ESR2* Expression

Expression values were extracted from the TPM normalised matrix across all 3,207 patient samples. The distribution of *ESR2* expression was then explored using summary statistics and visual inspection to decide on a suitable grouping strategy before classification.

3.3 Log2 Transformation of *ESR2* Values

ESR2 values were log2(x+1) transformed before visualisation and group assignment, since RNA-seq data are typically right-skewed and the transformation avoids undefined values at zero expression (Conesa et al., 2016). These transformed values were used only for group assignment. A separate log2(x+1) transformation was applied to the full gene expression matrix later, during preprocessing (Section 3 .7).

3.4 *ESR2* Group Assignment and Dataset Preparation

ESR2 expression values were grouped using a tertile-based strategy, following the approach used in the original SCAN-B study (Dalal et al., 2022). A median split was avoided given the right-skewed distribution, so cutoffs were instead calculated at the 33rd and 67th percentiles, using expression values from all 3,207 samples before the train/test split. Samples below the lower cutoff formed the ESR2 low group (n = 1,058), and those above the upper cutoff formed the ESR2 high group (n = 1,058). The middle tertile (n = 1,091) was excluded, since those samples sat too close to the classification boundary for confident labelling and would have introduced ambiguity into the class labels (Dalal et al., 2022). This left 2,116 samples for model development.

3.5 Removal of *ESR2* to Prevent Data Leakage

The *ESR2* gene was removed from the feature matrix once class labels had been assigned. Leaving it in would have allowed models to access the target variable directly during prediction, which would artificially inflate performance estimates (Kaufman et al., 2012). The final feature matrix contained 19,643 genes across 2,116 samples.

3.6 Data Splitting

The dataset was split into training and test sets using a stratified 80/20 ratio, implemented with the createDataPartition() function from the caret package. This gave a training set of 1,694 samples (847 ESR2 high, 847 ESR2 low) and an independent test set of 422 samples (211 per

class), with stratification keeping both classes equally represented across subsets (Kuhn & Johnson, 2013). The train/test split was repeated 25 times using seeds 1 through 25, and five-fold cross-validation was repeated using five random seeds (42, 123, 456, 789, 999), giving 25 fold-level results in total. Both were repeated to see how much the performance estimates might vary by chance, and unless stated otherwise, reported metrics reflect the mean and standard deviation across these 25 values. The test set was held back entirely during preprocessing and model development, with cross-validation applied only within the training data (Kaufman et al., 2012).

3.7 Preprocessing on Training Data

All preprocessing ran on the training data only, to prevent leakage into the test set (Kaufman et al., 2012). Genes with TPM values above 1 in fewer than 20% of training samples were removed as uninformative, leaving 14,787 genes (Bourgon et al., 2010). The remaining genes were log2(x+1) transformed to reduce the skewness typical of RNA sequencing data (Conesa et al., 2016), and the 500 genes with the highest variance across the training samples were then selected, consistent with approaches commonly used in transcriptomic classification studies (Sørlie et al., 2003; Parker et al., 2009).

Gene expression values were then standardised using z-score scaling, to prevent genes with larger numerical ranges from dominating the models (Kuhn & Johnson, 2013), with all scaling parameters estimated from the training data only.

3.8 Final Data Formatting

The processed training and test matrices were transposed so that rows represented samples and columns represented genes. Class labels were converted into factor variables for the binary classification task.

3.9 Regularized Logistic Regression Classification

LASSO regularised logistic regression was fitted using cv.glmnet() from the glmnet package (Friedman et al., 2010), with alpha = 1 to reduce overfitting in the high-dimensional feature space (Tibshirani, 1996). The optimal lambda was selected through cv.glmnet()'s internal cross-validation for tuning only, not as a performance estimate, and predictions on the test set used lambda.min. Model performance was evaluated using classification accuracy and AUC, with training and test accuracy compared to check for overfitting.

3.10 Random Forest Classification

A Random Forest classifier was fitted using the randomForest package (Liaw & Wiener, 2002), chosen because it handles high-dimensional gene expression data well and can capture complex relationships between features (Breiman, 2001; Díaz-Uriarte & Alvarez de Andrés, 2006). The model used 500 decision trees (ntree = 500). The mtry parameter, which controls how many genes are considered at each split, was tuned rather than left at its default. This was done using the train() function from the caret package, testing mtry values of 2, 7, 31, 125, and 500 with 5-fold cross-validation on the training data, with each tuning model built using 200 trees to keep computation time manageable. Once the mtry value with the highest mean ROC was identified, the final Random Forest model was fitted using that value with 500 decision trees (Kuhn, 2008). Performance was evaluated on the independent test set rather than internal OOB estimates.

3.11 Support Vector Machine (SVM) Classification

SVM classification used the svm() function from the e1071 package (Meyer et al., 2023), with a linear kernel chosen for its computational efficiency in high-dimensional transcriptomic settings (Cortes & Vapnik, 1995). The cost parameter was tuned by testing five candidate values (0.00005, 0.0001, 0.0005, 0.001, 0.005) using 5-fold cross-validation on the training data. Across the 25

repeated splits, the best cost was usually 0.0005, while 0.001 was selected in 4 of the 25 splits. Probability estimation was enabled to allow AUC calculation, and performance was evaluated on the independent test set using classification accuracy and AUC, with training and test accuracy compared to assess overfitting.

3.12 Cross Validation

Five-fold cross-validation was repeated using five different random seeds (42, 123, 456, 789, and 999). For each seed, all 2,116 samples were divided into five folds, with each fold used once for validation while the other four trained the model, giving one accuracy and AUC value per fold. Every individual fold result was kept rather than averaged within each seed, giving 25 fold-level results per model across the five seeds, later compared against the 25 results from the repeated train/test split. Five folds were used as a practical balance between computational time and stable performance estimates, with roughly 423 samples in each validation fold (Kohavi, 1995).

As with the train/test split, every preprocessing step and hyperparameter decision inside each fold came from that fold's training portion alone: the low-expression filter, the 500 most variable genes, the z-score parameters, and the best lambda for LASSO were all determined from the fold's training data and applied unchanged to the validation fold, so nothing from the validation fold could shape any modelling choice (Kaufman et al., 2012; Varma & Simon, 2006).

Model settings matched the train/test split throughout, with mtry and SVM cost re-tuned inside each training fold using the same procedures described in Sections 3.10 and 3.11.

3.13 Statistical Analysis

Classification performance was assessed using accuracy, sensitivity, specificity, and AUC. LASSO does not output a class label directly but a probability between 0 and 1, so a 0.5 threshold was used to convert this into HIGH or LOW: if the probability was above 0.5 the sample was called HIGH, otherwise LOW. This is the default threshold used in binary classification. Random Forest and SVM give class labels directly, and these were used for accuracy, sensitivity, and specificity, while the predicted probabilities from these models were used to calculate AUC. Sensitivity and specificity were calculated from the confusion matrix, comparing predicted labels to true labels, with HIGH ESR2 expression set as the positive class:

$$\text{Accuracy} = \left(\frac{TP + TN}{TP+TN+FP+FN}\right)$$

$$\text{Sensitivity} = \left(\frac{TP}{TP+FN}\right)$$

$$\text{Specificity} = \left(\frac{TN}{TN+FP}\right)$$

where TP, TN, FP, and FN stand for true positives, true negatives, false positives, and false negatives (Altman & Bland, 1994). AUC was calculated from the ROC curve, which plots sensitivity against 1 − specificity at different thresholds, and reflects how well each model could separate HIGH from LOW ESR2 expression overall, not just at one cutoff (Hanley & McNeil, 1982). Accuracy, sensitivity, and specificity were calculated using the confusionMatrix function from the caret package, and AUC using the pROC package (Kuhn, 2008; Robin et al., 2011).

For cross-validation, each fold's accuracy and AUC value was kept as a separate observation rather than being averaged per seed, then compared against the repeated train/test split results to check whether the two evaluation methods gave different performance estimates. An unpaired Welch two-sample t-test was used separately for each model for accuracy and AUC, comparing the 25 values from cross-validation against the 25 values from the train/test split, since the two

sets could not be paired one-to-one. All tests were two-sided at a significance level of 0.05, with no correction for multiple comparisons, since the analysis was exploratory rather than confirmatory.

3.14 Use of AI Tools

Claude (Anthropic) was used to help edit and clarify written sections of this thesis, including parts of this chapter. All experimental design, analysis, code, interpretation, and scientific content were produced independently by the author.

4 Results

4.1 ESR2 Expression Distribution and Group Assignment

ESR2 expression values were taken from the TPM-normalised data for all 3,207 breast cancer samples and log2-transformed using log2(x + 1). The 33rd and 67th percentiles of these values were used as cutoffs for the *ESR2*-low and *ESR2*-high groups, giving a lower cutoff of 0.022 and an upper cutoff of 0.149 (around 0.015 and 0.109 TPM on the raw scale). Samples below the lower cutoff were put in the *ESR2*-low group (n = 1,058), and samples above the upper cutoff in the *ESR2*-high group (n = 1,058). The 1,091 samples in between were excluded, as their expression was too intermediate to reliably assign to either group. Note that because several samples had identical expression values right at the cutoffs, the two groups did not end up exactly equal in size even though the cutoffs came from the 33rd and 67th percentiles. This left 2,116 samples for the classification task (Figure 1).

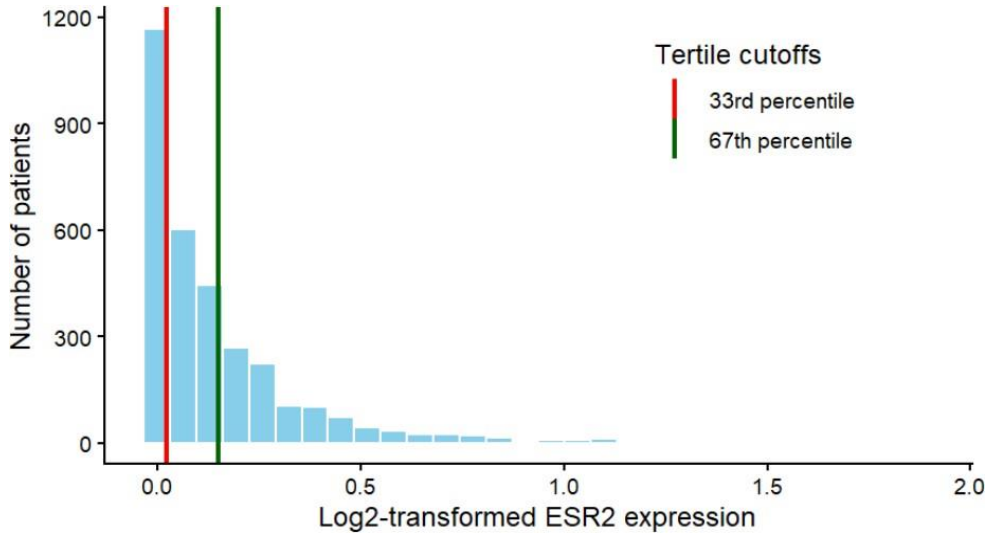

Figure 1. Distribution of $\log_2$-transformed *ESR2* expression in the 3,207 breast cancer samples. The red and green vertical lines show the lower (33rd percentile) and upper (67th percentile) tertile used to divide the samples into *ESR2*-low and *ESR2*-high groups. Samples within the middle tertile were excluded from the classification analysis.

4.2 Preprocessing and Feature Selection

For each 80/20 train/test split, genes with low expression were removed, leaving approximately 14,778–14,792 genes out of the original 19,643. A gene was kept if its TPM value was greater than 1 in at least 20% of the training samples. The data were then $\log_2$-transformed and the 500 most variable genes were selected using the training data. The selected genes were then scaled using the means and standard deviations calculated from the training set, and the same scaling parameters were applied to the test set. This resulted in a training dataset of 1,694 samples and 500 genes, and a test dataset of 422 samples with the same 500 genes.

4.3 Model Performance on Independent Test Set

All three models were trained on the training data and then evaluated on the corresponding test sets. This process was repeated 25 times using different random seeds, with an 80/20 split between the training and test data. The classification performance of each model is summarized in Table 1.

Table 1. Performance of the LASSO, Random Forest, and support vector machine (SVM) models across 25 repeated 80/20 train-test splits. Test Accuracy is the proportion of test samples that were correctly classified as ESR2-high or ESR2-low, and Test AUC is the area under the ROC curve on the test set. Values are presented as mean ± standard deviation (SD) for test accuracy and test AUC.

Model	Test Accuracy (mean ± SD)	Test AUC (mean ± SD)
LASSO	0.8192 ± 0.0116	0.8931 ± 0.0101
Random Forest	0.8251 ± 0.0128	0.8982 ± 0.0108
SVM	0.8130 ± 0.0168	0.8903 ± 0.0103

Across the 25 repeated train/test splits, Random Forest came out on top for both mean test accuracy (0.8251 ± 0.0128) and mean test AUC (0.8982 ± 0.0108). LASSO was close behind, with a mean test accuracy of 0.8192 ± 0.0116 and a mean AUC of 0.8931 ± 0.0101. SVM performed slightly worse than the other two, with a mean test accuracy of 0.8130 ± 0.0168 and a mean AUC of 0.8903 ± 0.0103. Random Forest reached perfect training accuracy (1.000) in every single split, but its test accuracy dropped to 0.8251, a clear sign of overfitting to the training data. The ROC curves for the three models are shown in Figure 2. The curves were quite similar, with only small differences in AUC between the models.

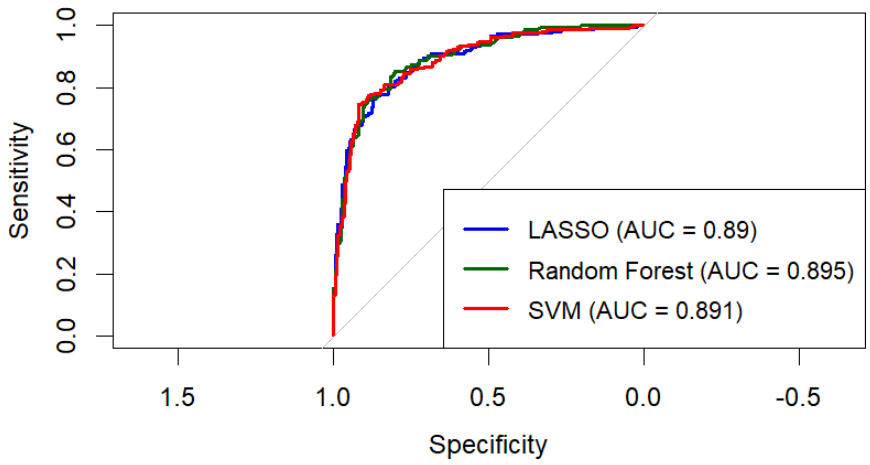

Figure 2. ROC curves for the LASSO, Random Forest, and support vector machine (SVM) models from one representative train/test split (seed 25). The diagonal line shows the performance expected from random classification. The AUC values shown are from this split, while the mean AUC values from all 25 repeated splits are presented in Table 1.

4.4 Cross-Validation Performance

Repeated five-fold cross-validation was performed using all 2,116 samples retained after *ESR2* group assignment. Five different random seeds were used, resulting in 25 outer validation folds in total. In each fold, approximately 80% of the samples were used for training and the remaining 20% were used for validation. The cross-validation results are summarised in Table 2. Random Forest achieved the highest mean accuracy of 0.8268 and mean AUC of 0.8970. LASSO achieved a mean accuracy of 0.8205 and a mean AUC of 0.8943, while SVM achieved a mean accuracy of 0.8153 and a mean AUC of 0.8936. Overall, the three models showed similar performance, with only small differences in their mean accuracy and AUC values. The standard deviations for accuracy and AUC were low for all three models, suggesting relatively stable performance across the repeated cross-validation runs. For accuracy, LASSO showed the lowest variation (SD = 0.0205) and Random Forest the highest (SD = 0.0225), although all three values were fairly close. For AUC, Random Forest showed the lowest variation (SD = 0.0154) and SVM the highest (SD = 0.0189). During SVM hyperparameter tuning, a cost of 0.0005 was selected in most of the outer folds, while a cost of 0.001 was selected in some folds. This suggests that the selected SVM cost was generally consistent across the cross-validation procedure.

Table 2. Classification performance of the LASSO, Random Forest, and support vector machine (SVM) models based on repeated five-fold cross-validation. The mean and standard deviation of accuracy and AUC were calculated across all 25 individual fold-level results (5 seeds x 5 folds) for each model.

Model	Mean Accuracy	Accuracy SD	Mean AUC	AUC SD
LASSO	0.8205	0.0205	0.8943	0.0176
Random Forest	0.8268	0.0225	0.8970	0.0154
SVM	0.8153	0.0213	0.8936	0.0189

4.5 Cross-Validation vs Train/Test Performance

The results from the repeated train/test split and repeated five-fold cross-validation are shown in Table 3. For LASSO, the mean accuracy increased slightly from 0.8192 to 0.8205, while the mean AUC increased from 0.8931 to 0.8943. For Random Forest, the mean accuracy increased from 0.8251 to 0.8268, while the mean AUC decreased slightly from 0.8982 to 0.8970. SVM showed the largest change, with mean accuracy rising from 0.8130 to 0.8153 and mean AUC from 0.8903 to 0.8936.

Table 3. Mean test accuracy and AUC from the repeated 80/20 train/test split and repeated 5-fold cross-validation, with the difference between the two methods for each model. Repeated Train/Test Accuracy and Repeated Train/Test AUC are the mean values across the 25 train/test splits; CV Accuracy and CV AUC are the mean values across the 25 cross-validation folds (5 folds × 5 seeds). Accuracy Difference and AUC Difference are CV values minus train/test values, so a positive number means cross-validation gave a higher estimate.

Model	Repeated Train/Test Accuracy	CV Accuracy	Accuracy Difference	Repeated Train/Test AUC	CV AUC	AUC Difference
LASSO	0.8192	0.8205	+0.0013	0.8931	0.8943	+0.0012
Random Forest	0.8251	0.8268	+0.0017	0.8982	0.8970	− 0.0012
SVM	0.8130	0.8153	+0.0023	0.8903	0.8936	+0.0033

For accuracy, cross-validation came out slightly higher for all three models, though the differences were small: 0.0013 for LASSO, 0.0017 for Random Forest, and 0.0023 for SVM. SVM had the largest gap and LASSO the smallest. For AUC, the results were less consistent. Cross-validation was slightly higher for LASSO (0.0012) and SVM (0.0033), but slightly lower for

Random Forest (−0.0012). SVM again showed the biggest difference, while LASSO and Random Forest were both close to zero.

4.6 Statistical Significance of Evaluation Difference

For each model, an unpaired Welch two-sample t-test was used to compare the 25 accuracy and AUC values from cross-validation against the 25 values from the repeated train/test split. None of the six comparisons were statistically significant at the 0.05 level. These results should still be interpreted with some caution, since the 25 cross-validation estimates are not fully independent because they come from repeated splits of the same dataset. The actual differences in mean performance were also very small. For accuracy, the differences between cross-validation and repeated train/test evaluation ranged from 0.0013 to 0.0023, while the AUC differences ranged from −0.0012 to 0.0033. Overall, there was no clear evidence of a large or consistent difference between the two evaluation methods in this dataset. This does not mean that the methods are statistically equivalent, only that no meaningful difference was detected here.

5 Discussion

ESR2 was chosen as the classification target based on the biological rationale outlined in the introduction (Dalal et al., 2022; Choi, 2022). Overall, *ESR2* expression was quite low across the whole cohort. The cutoff used for the "high" group only worked out to a raw TPM of around 0.11. That raises a fair question: is the difference between the *ESR2*-high and *ESR2*-low groups actually biologically meaningful, or is it just small variation within an already low-expression gene? Since all three models were able to separate the groups fairly consistently, the results suggest that the remaining gene-expression profiles contain information that can distinguish the two ESR2-defined groups. But whether that difference actually matters biologically, given how low the expression is overall, is something that would need to be looked at further, ideally using protein-level data such as immunohistochemistry.

All three models performed similarly on the test set (Kourou et al., 2015). AUC ranged from 0.8903 for SVM up to 0.8982 for Random Forest, with LASSO close behind at 0.8931. Because the scores were so close, the gene-expression profiles seem to contain a strong enough signal to distinguish ESR2-high and ESR2-low tumours that the choice of model did not matter much. This lines up with other transcriptomic studies, which have found that different models tend to perform similarly when the underlying biological signal is strong (Statnikov et al., 2008).

Random Forest had the best test AUC (0.8982), but it also showed the clearest sign of overfitting. It reached 100% training accuracy every time, while its test accuracy dropped to 0.8251. This is fairly typical of random forests. The 100% training accuracy, compared with the noticeably lower test accuracy, suggests that the model fitted the training data very closely but was less accurate when making predictions on new, unseen samples (Díaz-Uriarte & Alvarez de Andrés, 2006; Hastie et al., 2009; Breiman, 2001). Averaging across the trees helped keep generalisation reasonable, but the gap between training and test accuracy is a good reminder that training accuracy alone does not say much about how well a model will actually generalise, especially for tree-based models with high-dimensional data like this.

LASSO's accuracy was fairly stable across the repeated cross-validation runs, with a standard deviation of 0.0205. This was close to the other two models, Random Forest at 0.0225 and SVM at 0.0213, so none of the three stood out as clearly more variable than the others. This is likely because LASSO's penalty shrinks weak or uninformative coefficients toward zero, helping limit overfitting in a high-dimensional setting like this one (Tibshirani, 1996) which was fitted here using the glmnet package (Friedman et al., 2010). With 500 gene features and only 1,694 training samples, that kind of regularisation is a sensible fit for the task.

SVM's tuned cost parameter also stayed fairly consistent across runs. Across the 25 repeated splits, a cost of 0.0005 was selected in 21 runs, while 0.001 was selected in the other 4 runs. This consistency suggests the ideal decision boundary did not shift much depending on which samples ended up in training, and that the tuning process was not just reacting to noise. Since a linear kernel is relatively simple, cost was really the only thing that needed tuning, which probably helped keep SVM's performance stable, which fits with how SVM was originally designed to find the best linear boundary between classes (Cortes & Vapnik, 1995), even though it ended up a bit behind Random Forest.

Looking at the variation across both evaluation methods, Random Forest's accuracy SD was 0.0128 under the train/test split but 0.0225 under cross-validation, so it was actually a bit more variable under cross-validation, not less. This does not match the original expectation that averaging across folds would smooth out some of that variability for tree-based models. It is possible this just reflects the particular folds and samples used in each run, rather than any real difference in how stable Random Forest is between the two evaluation methods.

The main goal of this study was to check whether five-fold cross-validation gives different performance estimates compared with a single independent train/test split, using ESR2 expression as the classification target. The differences between the two methods were small for all three models: cross-validation AUC came out slightly higher than test AUC for LASSO (by 0.0012) and SVM (by 0.0033), but slightly lower for Random Forest (by 0.0012). The direction of the difference was not even consistent across models, which points more toward noise than a real difference. The unpaired t-tests also found none of these differences to be statistically significant (LASSO $p = 0.767$, Random Forest $p = 0.751$, SVM $p = 0.449$), though these p-values should be interpreted cautiously, since the cross-validation values are not completely independent. Overall, the results did not show any clear difference between cross-validation and repeated train/test evaluation in terms of the performance estimates.

This makes sense given the size of the dataset. With roughly 423 samples per validation fold, there was probably enough data for fairly stable estimates either way (Kohavi, 1995). For a breast cancer gene expression dataset of this size and with balanced classes, cross-validation and a single train/test split seem to give close enough results that it likely would not matter much in practice which one is used, consistent with the idea that validation choice matters more in smaller or less consistent datasets (Vabalas et al., 2019).

Cross-validation is generally expected to be a little optimistic, since each fold's training set is smaller than the full training set (Arlot & Celisse, 2010). LASSO and SVM followed that pattern here, but Random Forest did not, coming out slightly lower under cross-validation instead. Given how small all three differences were, well under 0.01 AUC units, this looks more like noise than any real, consistent effect in either direction. A difference this size is unlikely to change which model gets picked in a research or clinical setting. There could still be a small real effect that this study was not powered to detect, though comparing 25 values on each side gives more power to catch it than the earlier five-repetition version did.

Overall, this study found no clear evidence that five-fold cross-validation gave different performance estimates compared with the train/test split. The limited number of cross-validation seeds is discussed further below.

5.1 Limitations

There are a few limitations to this study. First, only one dataset was used, SCAN-B, which is made up mostly of Swedish patients. The sample size (3,207) is large, which helps with statistical power, but the results might not apply to other populations, treatment backgrounds, or sequencing platforms. Testing this on other datasets would be needed before saying these findings hold more generally.

ESR2 expression was also quite low across the whole cohort. The cutoff for the high group corresponded to a raw TPM of around 0.11, which is actually below the TPM of 1 that is normally used as a sign of active transcription. So, while the tertile split does separate patients into two groups based on relative expression, it is not clear whether this difference is biologically meaningful at the protein level, since that can't really be judged from expression data alone.

The ESR2 tertile cutoffs were calculated using the full cohort before splitting the data into training and test sets. This means that the test samples were also used when defining the outcome groups, so the test set was not completely independent in terms of how the outcome was defined, which represents a form of information leakage (Kaufman et al., 2012). However, all model preprocessing steps, including gene filtering, variance selection, and scaling, were based only on the training data. Since the tertile cutoffs were calculated from more than 3,000 samples, this is unlikely to have had a large practical effect on the results. Ideally, future studies should define the ESR2 expression groups using an independent reference cohort before evaluating the model, since even small leaks like this can undermine confidence in reported performance (Kapoor & Narayanan, 2023).

For Random Forest, the mtry parameter was tuned using a grid of five values through caret, with five-fold cross-validation used to pick the best one in each split. This is a fairly small grid compared to testing every possible mtry value, so it is possible a wider search would have found a slightly better setting.

The train/test split was repeated 25 times, while cross-validation was repeated using five random seeds, with each seed giving five fold-level results. This kept the computation time manageable while still giving a reasonable number of values to compare. Using more cross-validation seeds would likely have made the comparison more robust, but this was not done here due to time constraints.

The two evaluation methods were compared using an unpaired Welch t-test since the 25 cross-validation results and the 25 train/test split results couldn't be directly matched up. They came from different procedures, the cross-validation results were based on 5 seeds with 5 folds each, while the train/test results came from 25 independent splits, so a paired test didn't really make sense here. That said, the 25 cross-validation results aren't fully independent either, since the folds from different seeds were built from the same pool of samples, which could affect how well the Welch t-test's independence assumption actually holds. A more rigorous way to handle this would be nested cross-validation, where tuning and evaluation happen inside the same loop (Varma & Simon, 2006). What was done here is similar in spirit, just not fully nested.

A few things could be looked at in future work. Testing on other breast cancer datasets, ideally from different populations or sequencing platforms, would show whether these results generalise. Implementing full nested cross-validation would also make the comparison between evaluation strategies more rigorous, since tuning and evaluation would happen inside the same loop rather than being handled somewhat separately, as was done here (Varma & Simon, 2006).

It would also be worth trying a wider range of mtry values for Random Forest and testing different feature selection thresholds, to see if that improves performance beyond what was found here. Comparing these models to something like a neural network could also be useful, since that would allow non-linear patterns in the data to be captured that these models might miss. Lastly, linking the *ESR2* expression groups to actual clinical outcomes, like survival or response to treatment, would help show whether the differences found here have any real prognostic value.

6 Conclusions

This study looked at whether five-fold cross-validation gives a different picture of model performance compared to a repeated train/test split, using LASSO, Random Forest, and SVM to classify breast cancer samples by *ESR2* expression. Cross-validation gave slightly higher accuracy for all three models, and slightly higher AUC for LASSO and SVM, although Random Forest's AUC was marginally lower under cross-validation. The differences were small across all three models and did not follow the same pattern. None of them were statistically significant in the t-tests, although this should be interpreted with some caution because the cross-validation values are not fully independent. Overall, with 2,116 samples, there was no meaningful sign that cross-validation overestimated performance, and both evaluation methods gave very similar results.

The three models also performed quite similarly on the test set, with AUC ranging from about 0.890 to 0.898. This suggests the difference between *ESR2*-high and *ESR2*-low tumours was strong enough that it didn't matter much which model was used. Random Forest had the best accuracy and AUC overall, but it also overfit clearly, reaching 100% training accuracy in every split. LASSO was more consistent between its training and test performance, likely due to its built-in regularisation.

Overall, this suggests that for this dataset and modelling setup, it probably doesn't matter much whether cross-validation or a single train/test split is used to evaluate a model, at least not in terms of the final performance numbers. That doesn't mean data leakage stops being a concern, since careful preprocessing was still needed throughout this pipeline, but it does suggest that worries about cross-validation being overly optimistic may matter less once the sample size is large. Testing this again with more repeated runs, a fully nested cross-validation setup, or a different dataset altogether would help confirm whether this holds up beyond ESR2 and beyond SCAN-B.

7 Future Perspectives

A few things could be done to build on this study. One important next step would be testing the classification approach and the evaluation comparison on other breast cancer cohorts, ideally ones from different populations and using different sequencing platforms. This matters because models trained on a single cohort don't always perform as well when tested on external data, so checking whether these results hold up elsewhere would be a useful next step (Vabalas et al., 2019; Kapoor & Narayanan, 2023).

It would also be worth implementing a fully nested cross-validation setup. In this study, hyperparameter tuning and the final performance evaluation weren't run entirely inside the same loop, so there's a small chance this affected how fair the comparison between the two evaluation strategies really was. Full nesting is generally seen as the more rigorous way to get a reliable performance estimate in this kind of biomedical machine learning work (Varma & Simon, 2006).

There's also more that could be done on the modelling side. Trying a wider range of mtry values for Random Forest, along with different feature selection thresholds, could show whether performance improves beyond what was achieved here. Comparing these results to a deep learning model could be useful too, since models like that can pick up non-linear patterns in gene expression data that LASSO and standard Random Forest aren't really built to capture (Tran et al., 2021).

Lastly, it would be worth looking at whether the *ESR2* expression groups actually relate to clinical outcomes like survival or how patients respond to treatment. This would help show whether the transcriptomic differences found in this study are just a statistical pattern or something with real clinical relevance.

8 Ethics and Societal Relevance

The gene expression data used in this study come from the SCAN-B study (GSE202203), collected under ethical approval from the Regional Ethical Review Board in Lund, Sweden, and released publicly through the NCBI Gene Expression Omnibus. No new patient data were collected, and no personally identifiable information was accessed. The dataset was used in accordance with the terms of its public release, and no additional ethical approval was required for this secondary analysis.

Breast cancer predominantly affects women, and the SCAN-B cohort is almost entirely female. The models trained here may not perform equally well across all patient subgroups, including male breast cancer patients or those from different ethnic backgrounds. Any future clinical use of these methods would need to demonstrate equitable performance across groups before deployment. This is particularly important given that machine learning models trained on predominantly European cohorts may not generalise reliably to patients from other populations.

Machine learning tools are increasingly being explored for cancer diagnosis and prognosis, and how those tools are evaluated matters for patient safety. Overly optimistic performance estimates from poorly implemented cross-validation could result in models entering clinical practice before their true generalisation performance is known (Morley et al., 2020). For this dataset and modelling setup, five-fold cross-validation gave very similar performance estimates to the train/test approach. There was no clear evidence that cross-validation introduced any meaningful bias in this case.

Reproducibility is an ethical issue too. All analyses were run in R with fixed random seeds, and the code is available for inspection. In clinical bioinformatics, methodological errors can have real consequences for patient care, which is why transparent and reproducible pipelines matter.

References

Arlot, S., & Celisse, A. (2010). A survey of cross-validation procedures for model selection. *Statistics Surveys*, 4, 40–79. https://doi.org/10.1214/09-SS054

Bourgon, R., Gentleman, R., & Huber, W. (2010). Independent filtering increases detection power for high-throughput experiments. *Proceedings of the National Academy of Sciences*, 107(21), 9546–9551. https://doi.org/10.1073/pnas.0914005107

Breiman, L. (2001). Random forests. *Machine Learning*, 45, 5-32. https://doi.org/10.1023/A:1010933404324

Choi, Y. (2022). Estrogen Receptor β Expression and Its Clinical Implication in Breast Cancers: Favorable or Unfavorable? *Journal of Breast Cancer*, 25(2), 75–93. https://doi.org/10.4048/jbc.2022.25.e9.

Conesa, A., Madrigal, P., Tarazona, S., Gomez-Cabrero, D., Cervera, A., McPherson, A., Szcześniak, M. W., Gaffney, D. J., Elo, L. L., Zhang, X., & Mortazavi, A. (2016). A survey of best practices for RNA-seq data analysis. *Genome Biology*, 17(1), 13. https://doi.org/10.1186/s13059-016-0881-8

Cortes, C., & Vapnik, V. (1995). Support-vector networks. *Machine Learning*, 20(3), 273–297. https://doi.org/10.1007/BF00994018

Dalal, H., Dahlgren, M., Gladchuk, S., Brueffer, C., Gruvberger-Saal, S. K., & Saal, L. H. (2022). Clinical associations of *ESR2* (estrogen receptor beta) expression across thousands of primary breast tumors. *Scientific Reports*, 12, 4696. https://doi.org/10.1038/s41598-022-08210-3

Díaz-Uriarte, R., & Alvarez de Andrés, S. (2006). Gene selection and classification of microarray data using random forest. *BMC Bioinformatics*, 7(1), 3. https://doi.org/10.1186/1471-2105-7-3

Friedman, J., Hastie, T., & Tibshirani, R. (2010). Regularization paths for generalized linear models via coordinate descent. *Journal of Statistical Software*, 33(1), 1–22. https://doi.org/10.18637/jss.v033.i01

Hastie, T., Tibshirani, R., & Friedman, J. (2009). *The elements of statistical learning: Data mining, inference, and prediction* (2nd ed.). Springer. https://doi.org/10.1007/978-0-387-84858-7

Jordan, M. I., & Mitchell, T. M. (2015). Machine learning: Trends, perspectives, and prospects. Science, 349(6245), 255–260. https://doi.org/10.1126/science.aaa8415

Kapoor, S., & Narayanan, A. (2023). Leakage and the reproducibility crisis in machine-learning-based science. Patterns, 4(9), 100804. https://doi.org/10.1016/j.patter.2023.100804

Kaufman, S., Rosset, S., Perlich, C., & Stitelman, O. (2012). Leakage in data mining: Formulation, detection, and avoidance. ACM Transactions on Knowledge Discovery from Data, 6(4), 1–21. https://doi.org/10.1145/2382577.2382579

Kohavi, R. (1995). A study of cross-validation and bootstrap for accuracy estimation and model selection. *Proceedings of the 14th International Joint Conference on Artificial Intelligence*, 2, 1137–1145.

Kourou, K., Exarchos, T. P., Exarchos, K. P., Karamouzis, M. V., & Fotiadis, D. I. (2015). Machine learning applications in cancer prognosis and prediction. Computational and Structural Biotechnology Journal, 13, 8–17. https://doi.org/10.1016/j.csbj.2014.11.005

Kuhn, M. (2008). Building predictive models in R using the caret package. Journal of Statistical Software, 28(5), 1–26. https://doi.org/10.18637/jss.v028.i05

Kuhn, M., & Johnson, K. (2013). *Applied predictive modeling*. Springer. https://doi.org/10.1007/978-1-4614-6849-3

Liaw, A., & Wiener, M. (2002). Classification and regression by randomForest. *R News*, 2(3), 18–22.

Manzoni, C., Kia, D. A., Vandrovcova, J., Hardy, J., Wood, N. W., Lewis, P. A., & Ferrari, R. (2018). Genome, transcriptome and proteome: The rise of omics data and their integration in biomedical sciences. Briefings in Bioinformatics, 19(2), 286–302. https://doi.org/10.1093/bib/bbw114

Meyer, D., Dimitriadou, E., Hornik, K., Weingessel, A., & Leisch, F. (2023). e1071: Misc functions of the Department of Statistics, Probability Theory Group (R package version 1.7-17) [Computer software]. https://CRAN.R-project.org/package=e1071

Morley, J., Machado, C. C. V., Burr, C., Cowls, J., Joshi, I., Taddeo, M., & Floridi, L. (2020). The ethics of AI in health care: A mapping review. *Social Science & Medicine*, *260*, 113172. https://doi.org/10.1016/j.socscimed.2020.113172

Parker, J. S., Mullins, M., Cheang, M. C. U., Leung, S., Voduc, D., Vickery, T., Davies, S., Fauron, C., He, X., Hu, Z., Quackenbush, J. F., Stijleman, I. J., Palazzo, J., Marron, J. S., Nobel, A. B., Mardis, E., Nielsen, T. O., Ellis, M. J., Perou, C. M., & Bernard, P. S. (2009). Supervised risk predictor of breast cancer based on intrinsic subtypes. Journal of Clinical Oncology, 27(8), 1160–1167. https://doi.org/10.1200/JCO.2008.18.1370

R Core Team. (2026). *R: A language and environment for statistical computing* [Computer software]. R Foundation for Statistical Computing. https://www.R-project.org/

Robin, X., Turck, N., Hainard, A., Tiberti, N., Lisacek, F., Sanchez, J. C., & Müller, M. (2011). pROC: An open-source package for R and S+ to analyze and compare ROC curves. BMC Bioinformatics, 12(1), 77. https://doi.org/10.1186/1471-2105-12-77

Slowikowski, K. (2023). *ggrepel: Automatically position non-overlapping text labels with ggplot2* (Version 0.9.8) [Computer software]. https://CRAN.R-project.org/package=ggrepel

Sørlie, T., Tibshirani, R., Parker, J., Hastie, T., Marron, J. S., Nobel, A., Deng, S., Johnsen, H., Pesich, R., Geisler, S., Demeter, J., Perou, C. M., Lønning, P. E., Brown, P. O., Børresen-Dale, A. L., & Botstein, D. (2003). Repeated observation of breast tumor subtypes in independent gene expression data sets. Proceedings of the National Academy of Sciences, 100(14), 8418–8423. https://doi.org/10.1073/pnas.0932692100

Statnikov, A., Wang, L., & Aliferis, C. F. (2008). A comprehensive comparison of random forests and support vector machines for microarray-based cancer classification. BMC Bioinformatics, 9(1), 319. https://doi.org/10.1186/1471-2105-9-319

Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. Journal of the Royal Statistical Society: Series B, 58(1), 267–288. https://doi.org/10.1111/j.2517-6161.1996.tb02080.x

Tran, K. A., Kondrashova, O., Bradley, A., Williams, E. D., Pearson, J. V., & Waddell, N. (2021). Deep learning in cancer diagnosis, prognosis and treatment selection. Genome Medicine, 13(1), 152. https://doi.org/10.1186/s13073-021-00968-x

Vabalas, A., Gowen, E., Poliakoff, E., & Casson, A. J. (2019). Machine learning algorithm validation with a limited sample size. PLOS ONE, 14(11), e0224365. https://doi.org/10.1371/journal.pone.0224365

Varma, S., & Simon, R. (2006). Bias in error estimation when using cross-validation for model selection. BMC Bioinformatics, 7(1), 91. https://doi.org/10.1186/1471-2105-7-91

Wagner, G. P., Kin, K., & Lynch, V. J. (2012). Measurement of mRNA abundance using RNA-seq data: RPKM measure is inconsistent among samples. Theory in Biosciences, 131(4), 281–285. https://doi.org/10.1007/s12064-012-0162-3

Wickham, H., Hester, J., & Bryan, J. (2023). *readr: Read rectangular text data* (Version 2.2.0) [Computer software]. https://CRAN.R-project.org/package=readr