Beyond the Boundaries: Evaluating Classifier Performance in Breast Cancer Detection

Oran Frawley
Dept. of Mechanical Engineerig
University of Galway
Galway, Ireland
o.frawley2@universityofgalway.ie

***Abstract*—This report evaluates the performance of several classifiers – k-Nearest Neighbour, Naïve Bayes, Linear Discriminant Analysis and Quadratic Discriminant Analysis – on the Wisconsin Breast Cancer Detection dataset. Classifiers are evaluated using varying numbers of features to determine the effects the feature sets have on the classifier performance . (*Abstract*)**

I. Introduction

This report aims to investigate the effectiveness of a k-Nearest Neighbour (k-NN) classifier in comparison to several probabilistic classifiers for classifying instances as malignant or benign. The Wisconsin Breast Cancer Detection (WBCD) dataset, a commonly used dataset in machine learning and medical research, is deployed in this investigation. Further details of this dataset are discussed in Section II. C.

Section II also discusses in detail the various types of classifiers which are used in the investigation, along with the performance measures applied to evaluate them. Section III outlines the methods used for implementing each of the classifiers, including the methods of feature normalisation used. Section IV displays relevant tables and graphs, showcasing results from each classifier and discusses the performance of each classifier. Finally, Section V discusses the findings of the investigation and provides concluding remarks.

II. Background

A. Classifiers Used in the Investigation

For this investigation, a k-NN classifier will be compared to 3 probabilistic classifiers – Naïve Bayes (NB), Linear Discriminant Analysis (LDA) and Quadratic Discriminant Analysis (QDA). Each of the classifiers will be optimised for prediction using the WDBC dataset, and the optimised classifiers will be compared against each other.

K-NN is a widely used classification technique, which utilises the principle that close data points in feature space are often of the same class. When classifying a new instance, the algorithm takes the k closest data points from the training set on a predetermined distance metric, commonly Euclidian distance. The class is assigned based on a majority vote of these k closest points.

Naïve Bayes classifiers are based on Bayes' Theorem with the assumption that all components of the feature vectors which are used as input to the classifier are strongly independent given the class that the feature represents. The classifier works by generating a conditional probability for each of the classes supported by the classifier. Due to the simple nature of NB classifiers, they can be extremely fast compared to more sophisticated methods and they allow each distribution to be independently estimated as a one dimensional distribution, alleviating problems associated with the curse of dimensionality.[1][2]

LDA and QDA classifiers build upon the methodologies of the NB classifier, making a more general assumption that the features of x are modelled as a single multivariate normal distribution. As their names suggest, the classifiers have a linear and quadratic decision surface respectively and they form closed form solutions which are easily computed, and have no hyperparameters to tune.[3]

The use of LDA is relatively uncommon, due to the fact stipulation that the dataset must be linearly separable for successful LDA classification. For LDA classification, the mean value of each of the N features across all examples of that class in the training dataset to calculate μ_i. Training a QDA classifier requires one additional step – calculating the covariance matrix for the class using all examples of that class in the training dataset to yield the Σ matrix.

B. Measures of Classifier Performance

To evaluate the effectiveness of each classifier type, several performance metrics were applied, including confusion matrices, classification reports and Receiver Operating Characteristic (ROC) curves.

- **Confusion matrix**: A table which summarises the counts of true positives, true negatives, false positives and false negatives. The layout helps identify specific types of errors, as the classifier aims to minimise the number of false positives and false negatives.
- **ROC Curve**: Plots the true positive rate against the false positive rate, visually illustrating the relationship between the two metrics. The Area Under the Curve (AUC) summarises the ROC curve, with AUC values tending towards 1.0 indicating better performance.
- **Classification Report**: A report which provides detailed metrics such as precision, recall and F1-score for each class. The formulae for obtaining these values are shown below in Equations 1, 2 and 3.

$$Precision = \frac{TP}{TP+FP} \quad (1)$$

$$Recall = \frac{TP}{TP+FN} \quad (2)$$

$$F1\ Score = 2\frac{Precision \times Recall}{Precision+Recall} \quad (3)$$

Where TP, TN, FP, and FN are the number of true positive, true negative, false positive and false negative values respectively.

C. Dataset Description

The WBCD dataset is a commonly used dataset in machine learning from the University of California repository. It consists of 569 instances with 30 numerical features that have been computed from images of samples of breast mass. The 30 features are divided into three groups of 10 features describing various physical characteristics of each cell nucleus such as radius, texture, perimeter, area, smoothness, compactness, concavity, concave points, symmetry and fractal dimension. The first 10 features contain the mean values of these characteristics, the second 10 features contain the standard error and the third 10 contain the maximum values. The dataset contains no missing values, making it perfect for straightforward classification.

III. Methodology

Each classifier is implemented in a manner such that the number of features included can be adjusted to include the first 10 , 20 or 30 features.

A. k-NN Classifier

When implementing a k-NN classifier for the given dataset, a 20% test data split is used, as is used by Mert et al.[4]. 5-fold cross validation is used to determine the optimal k-value, this is the standard for a dataset of this size as 10-fold may result in the folds that are too small to for accurate results. The feature sets are normalised, due to the nature of the dataset – the mean, standard deviation and max value features are on a completely different scale. Confusion matrices and classification reports are generated for k-NN classifiers using just the mean features, using the mean and standard deviation features and using mean, standard deviation and max values features for comparison. Only results for the test data (20%) are displayed, as this data is unseen it is a more useful measure of classifier accuracy.

B. Naïve Bayes Classifier

The specific NB classifier used in this investigation is a Gaussian Naïve Bayesian classifier, which assumes that an appropriate event model for all features is a normal distribution. The probability density function for this type of classifier is as follows:

$$P(x_j = v|C_i) = \frac{1}{\sqrt{2\pi\sigma_i^2}} e^{-\frac{(v-\mu_i)^2}{2\sigma_i^2}} \quad (4)$$

Where x_j is the parameter for a given class i, μ_i is the mean of all the x_j values of class i and σ_i is the sample variance of all the x_j values for examples of class i.

Confusion matrices, classification reports and ROC curves are generated for the first 10, 20 and 30 features, same as in the k-NN classifier for comparison and analysis.

C. LDA and QDA Classifiers

Both LDA and QDA classifier are easily implemented using their respective scikit functions. It uses the 'svd' solver is used for both algorithms by default, as it does not rely on the calculation of the covariance matrix. For LDA, two svd's are computed, one of the centred input matrix X and one of the class-wise mean vectors. As in the k-NN and NB classifiers, confusion matrices, classification reports and ROC curves are generated for the first 10, 20 and 30 features for comparison and analysis.

D. Feature Normalisation

Due to the nature of the dataset, where the first 10 features are mean values, second 10 are standard deviations and third 10 are max values, it is necessary to scale the dataset in order to use all features in the classification. Scikit-learn's StandardScaler is used for this scaling. This function standardizes the features of a dataset by removing the mean and scaling to unit variance using z-scaling.[5]

IV. Results

This section provides tables and graphs which describe the performance of each classifier, with varied numbers of features included.

A. k-NN Classifer

Separate classifiers are implemented using the first 10, 20 and 30 features, with 5-fold cross validation for each one. Optimal k-values are found to be: 7, 4 and 4 for the classifier using first 10, 20 and 30 features respectively. In the results, each classifier is denoted as 'First 10', 'First 20' and 'First 30', corresponding to the number of features included in the classifier. Table I shows the confusion matrix of each, denoting malignant and benign as *M* and *B* respectively. Table II shows the classification reports of each classifier, and Figure 1 shows their respective ROC curves, also displaying their AUC.

TABLE I. Confusion Matrices for k-NN Classifiers

		Predicted					
		First 10		***First 20***		***First 30***	
		M	*B*	*M*	*B*	*M*	*B*
Actual	*M*	41	2	43	0	41	2
	B	1	70	3	68	3	68

TABLE II. Classification Reports for k-NN Classifiers

		Precision	**Recall**	**f-1 Score**	**Support**
First 10	**Benign**	0.98	0.95	0.96	43
	Malignant	0.97	0.99	0.98	71
	Accuracy			0.97	114
	Macro Avg	0.97	0.97	0.97	114
	Weighted Avg	0.97	0.97	0.97	114
First 20	**Benign**	0.93	1.00	0.97	43
	Malignant	1.00	0.96	0.98	71
	Accuracy			0.97	114
	Macro Avg	0.97	0.98	0.97	114
	Weighted Avg	0.98	0.97	0.97	114
First 30	**Benign**	0.93	0.95	0.94	43
	Malignant	0.97	0.96	0.96	71
	Accuracy			0.96	114
	Macro Avg	0.95	0.96	0.95	114
	Weighted Avg	0.96	0.96	0.96	114

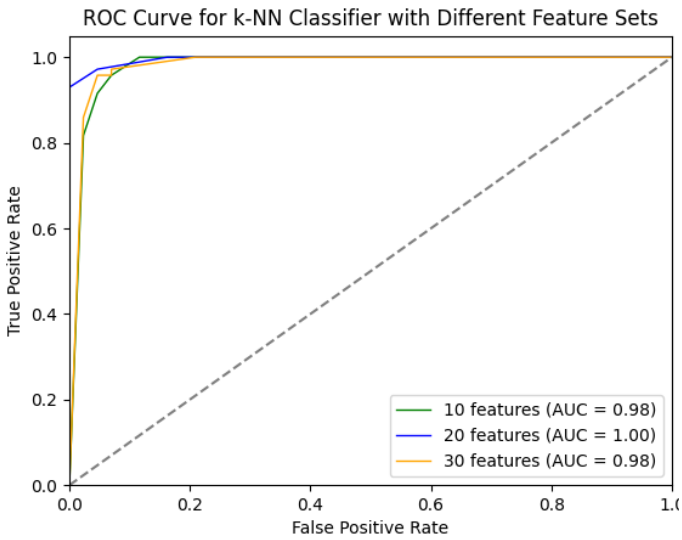

Fig. 1. ROC Curve for k-NN Classifiers

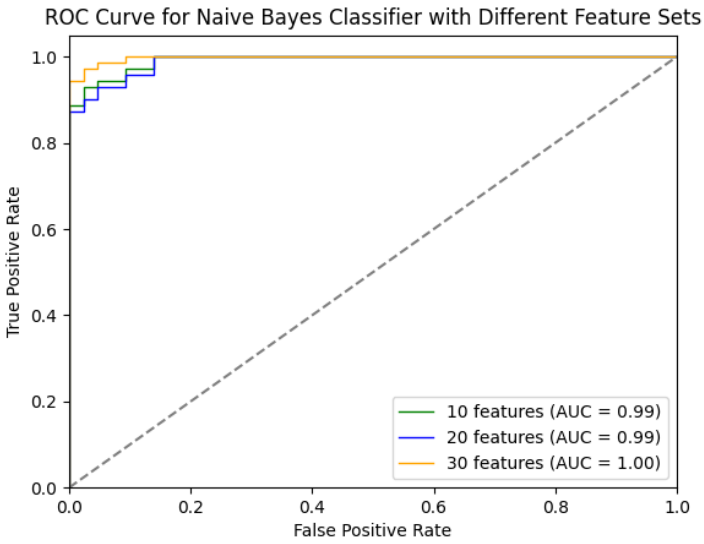

Fig. 2. ROC Curves for NB Classifiers

B. *Naïve Bayes Classifier*

Separate classifiers are also implemented using the NB classifier, as in the k-NN classifier. NB classifiers don't require hyperparameter tuning, therefore the implementation of the various classifiers is straightforward. Table III shows the confusion matrices for each classifier, with the same nomenclature as the previous section. Table IV shows the classification reports for each classifier and Figure 2 shows their ROC curves and AUC values.

TABLE III. CONFUSION MATRICES FOR NB CLASSIFIERS

		Predicted					
		First 10		***First 20***		***First 30***	
		M	*B*	*M*	*B*	*M*	*B*
Actual	*M*	38	5	37	6	40	3
	B	2	69	3	68	1	70

TABLE IV. CLASSIFICATION REPORTS FOR NB CLASSIFIERS

		Precision	**Recall**	**f-1 Score**	**Support**
First 10	**Benign**	0.95	0.88	0.92	43
	Malignant	0.93	0.97	0.95	71
	Accuracy			0.94	114
	Macro Avg	0.94	0.93	0.93	114
	Weighted Avg	0.94	0.94	0.94	114
First 20	**Benign**	0.93	0.86	0.89	43
	Malignant	1.00	0.96	0.94	71
	Accuracy			0.92	114
	Macro Avg	0.92	0.91	0.91	114
	Weighted Avg	0.92	0.92	0.92	114
First 30	**Benign**	0.98	0.93	0.95	43
	Malignant	0.96	0.99	0.96	71
	Accuracy			0.96	114
	Macro Avg	0.97	0.96	0.96	114
	Weighted Avg	0.97	0.96	0.96	114

C. *Linear Discriminant Analysis*

Separate LDA classifiers are also implemented as with the other classifiers. The implementation of this algorithm is straightforward, as outlined in Section III.C. Table V shoes the confusion matrices for each classifier, Table VI shows their classification reports and Figure 3 shows their ROC curves and AUC values.

TABLE V. CONFUSION MATRICES FOR LDA CLASSIFIERS

		Predicted					
		First 10		***First 20***		***First 30***	
		M	*B*	*M*	*B*	*M*	*B*
Actual	*M*	39	4	37	6	39	4
	B	4	67	2	69	1	70

TABLE VI. CLASSIFICATION REPORTS FOR LDA CLASSIFIERS

		Precision	**Recall**	**f-1 Score**	**Support**
First 10	**Benign**	0.91	0.91	0.91	43
	Malignant	0.94	0.94	0.94	71
	Accuracy			0.93	114
	Macro Avg	0.93	0.93	0.93	114
	Weighted Avg	0.93	0.93	0.93	114
First 20	**Benign**	0.95	0.86	0.90	43
	Malignant	0.92	0.97	0.95	71
	Accuracy			0.93	114
	Macro Avg	0.93	0.92	0.92	114
	Weighted Avg	0.93	0.93	0.93	114
First 30	**Benign**	0.97	0.91	0.94	43
	Malignant	0.95	0.99	0.97	71
	Accuracy			0.96	114
	Macro Avg	0.96	0.95	0.95	114
	Weighted Avg	0.96	0.96	0.96	114

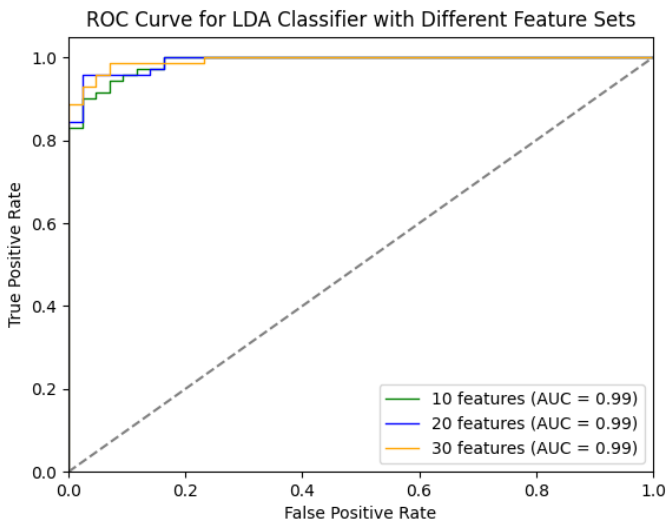

Fig. 3. ROC Curves for LDA Classifiers

D. Quadratic Discriminant Analysis

As with every other classifier type, separate QDA classifiers are implemented, which is again straightforward, with no need for hyperparameter tuning. Table VII shows the confusion matrices for each classifier, Table VIII shows their classification reports and Figure 4 shows their ROC curves and AUC values.

TABLE VII. CONFUSION MATRICES FOR QDA CLASSIFIERS

		Predicted					
		First 10		***First 20***		***First 30***	
		M	*B*	*M*	*B*	*M*	*B*
Actual	*M*	39	4	39	4	41	2
	B	3	68	4	67	3	68

TABLE VIII. CLASSIFICATION REPORTS FOR QDA CLASSIFIERS

		Precision	Recall	f-1 Score	Support
First 10	**Benign**	0.93	0.91	0.92	43
	Malignant	0.94	0.96	0.95	71
	Accuracy			0.94	114
	Macro Avg	0.94	0.93	0.93	114
	Weighted Avg	0.94	0.94	0.94	114
First 20	**Benign**	0.91	0.91	0.91	43
	Malignant	0.94	0.94	0.94	71
	Accuracy			0.93	114
	Macro Avg	0.93	0.93	0.93	114
	Weighted Avg	0.93	0.93	0.93	114
First 30	**Benign**	0.93	0.95	0.94	43
	Malignant	0.97	0.96	0.96	71
	Accuracy			0.96	114
	Macro Avg	0.95	0.96	0.95	114
	Weighted Avg	0.96	0.96	0.96	114

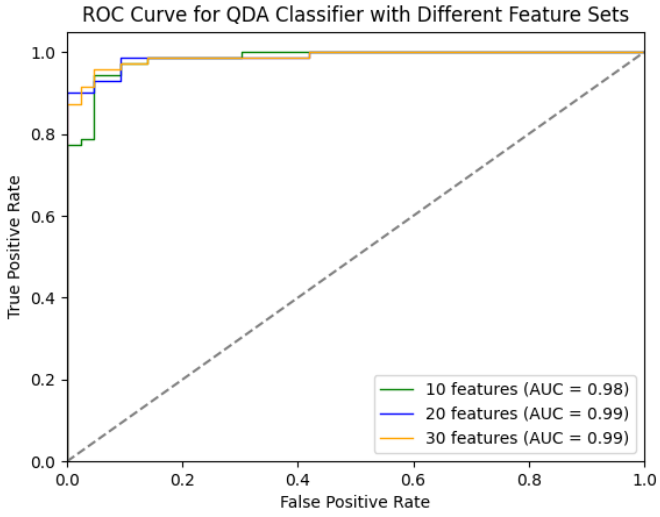

Fig. 4. ROC Curves for QDA Classifiers

E. Discussion of Results

The various confusion matrices, classification reports and ROC curves provide a valuable insight into the performance of each type of classifier when using different numbers of features.

The results reveal patterns in classifier performance as the feature sets are expanded from 10 to 20 to 30. The k-NN classifier performed consistently well across all feature sets, achieving high results on all performance metrics, with accuracies of 97% for both 10 and 20 features, and only dropping to 96% with 30 features.

The NB and LDA classifiers perform poorest of all the classifiers, however both show improved performance when using all 30 features, with an accuracy of 96% each. This increased performance suggests that the NB classifier utilised the complete dataset effectively. The QDA classifier also performs optimally with 30 features, likely due to the classifier's ability to effectively capture nonlinear boundaries.

V. CONCLUSION

K-NN, Naïve Bayes, Linear Discriminant Analysis and Quadratic Discriminant Analysis classifiers are developed and implemented for a dataset containing 569 instances in Python 3.12. The number of features used in each classifier is varied and they key findings of the investigation are:

- The K-NN classifier provides the most consistent results across all feature sets tested showing very high accuracy scores across all feature sets tested.
- The probabilistic classifiers all showed greatest accuracy when using all 30 features, with the QDA outperforming the other two classifiers when using fewer features.
- K-NN would be recommended for use with this dataset due to its consistently high accuracy, however each of the probabilistic classifiers compete well with k-NN when using all 30 features

REFERENCES

[1] Online, "Section 1.9 Naïve Bayes" Scikit-learn User Guide. https://scikit-learn.org/1.5/modules/naive_bayes.html

[2] Shashmi Karanam, Curse of Dimensionality – A "Curse" to Machine Learning, https://towardsdatascience.com/curse-of-dimensionality-a-curse-to-machine-learning-c122ee33bfeb

[3] Online, "Section 1.2. Linear and Quadratic Discriminant Analysis" Scikit-learn User Guide. https://scikit-learn.org/1.5/modules/lda_qda.html

[4] Mert, Ahmet & Kilic, Niyazi & Bilgili, Erdem & Akan, Aydin. (2015). Breast Cancer Detection with Reduced Feature Set. Computational and Mathematical Methods in Medicine. Article ID 265138. 11 pages. 10.1155/2015/265138

[5] Online, "StandardScaler" Scikit-learn User Guide. https://scikit-learn.org/dev/modules/generated/sklearn.preprocessing.StandardScaler.htm