Breast Cancer Diagnosys Prediction

Harvard Data Science Capstone Project for Breast Cancer Prediction

Author --- Amol Chaudhari

Date --- 22/06/2025

Contents

1 Introduction

The incidence of new cancer cases was estimated at 18 million in 2018 (Bray et al. 2018) and, based on current trends, is projected to increase to approximately 28 million by 2040 (CRUK 2019). In women, breast cancer is the most commonly diagnosed cancer and the leading cause of cancer mortality (Bray et al. 2018).

Despite increasing mortality rates globally (GBD 2015 Mortality and Causes of Death Collaborators 2016), survival rates have improved in high income countries such as the United Kingdom (The Nuffield Trust 2020) largely due to improvements made in detection and early diagnosis as well as in the treatment and management of the disease (Coleman et al. 2010; Walters et al. 2013).

Breast cancer screening programmes have been introduced in countries such as the UK in order to improve detection and early diagnosis and have been shown to improve breast cancer mortality (Marmot et al. 2013) but remain controversial not least because they can drive over-diagnosis and unnecessary biopsy and/or treatment (Nehmat and Nehmat 2017). Regardless, the use of screening to drive earlier diagnosis certainly places even greater emphasis on the ability to diagnose early signs of malignancy via biopsy accurately (Vetto et al. 1995; Ginsburg et al. 2020).

Fine needle aspiration (FNA) is a type of biopsy that remains a popular procedure for diagnosing breast cancer because it is relatively non-invasive, inexpensive and simple to administer (Lukasiewicz et al. 2017) despite continued controversy because of the small amount of breast tissue that is sampled that can be inadequate for accurate diagnosis (Casaubon 2020), resulting in substantially lower sensitivity, specificity and reproducibility versus the more invasive core needle biopsy procedure (Lukasiewicz et al. 2017). False negatives result in underdiagnosis and increases mortality risk while false positives result in overdiagnosis and risk of exposure to unnecessary treatments. A review of the literature on studies of core biopsies specifically in breast cancer found false negative rates ranging from 0% to 13% based on expert assessment by pathologists (Dillon et al. 2005). Machine learning algorithms thus have a high bar to be considered of clinical utility as an alternative to this expert evaluation by pathologists.

The Wisconsin breast cancer (diagnostic) database is a set of labelled multivariate data that has been used for classification based machine learning many times since it was first used in the 1990s (Street, Wolberg, and Mangasarian 1993; Wolberg, Street, and Mangasarian 1994; Wolberg 1995) and subsequently donated to the UCI machine learning repository (Dua and Graff 2017). The objective of this project was to use this data set to train different algorithms in order to accurately diagnosis breast cancer based on a prediction as to whether a given sample of cells was from a malignant (cancerous) or benign (non-cancerous) tumour mass.

2 Exploratory Analysis

2.1 Dataset overview

The Breast Cancer Wisconsin (Diagnostic) data set was downloaded from the UCI machine learning repository (Dua and Graff 2017). It is a data.table, data.frame consisting of 32 columns and 569 rows. The first and second columns provide the ID number and diagnosis for each patient respectively.

There are 569 unique patient ID numbers, confirming that each row represents a sample from a unique patient. The diagnosis column includes character strings that classify whether the samples were diagnosed as benign (B) or malignant (M). The data set consists of 357 (63%) benign samples and 212 (37%) malignant samples.

This imbalance between benign and malignant samples means that overall accuracy is likely not to be a sufficient measure of a predictive algorithm in isolation as it may mask the false negative rate, or type II error, i.e. the proportion of malignant samples misdiagnosed as benign. As such, examination of the confusion matrix, and the sensitivity or recall in particular, is critical during model development.

Table 1: Description of nuclear features

Feature	Description
Radius	Mean of distances from center to points on the perimeter of individual nuclei
Texture	Variance (standard deviation) of grey-scale intensitities in the component pixels
Perimeter	Perimeter length of each nucleus
Area	Area as measured by counting pixels within each nucleus
Smo othness	Local variation in radius lengths
Compactness	Combination of perimeter and area using the formula: (perimeter^2 / area - 1.0)
Concavity	Number and severity of concavities (indentations) in the nuclear contour
Concave Points	Number of concavities in the nuclear contour
Symmetry	Symmetry of the nuclei as measured by length differences between lines perpendicular to the major axis and the cell boundary
Fractal Dim	Fractal dimension based on the 'coastline approximation' - 1.0

Each of the features described are such that larger values will typically indicate a higher likelihood of malignancy given that they reflect larger cells and/or more irregular shapes.

Tables 2-4 summarise the numeric data for each of the features included in the data set, showing clearly that the range and magnitude of values for each feature varies considerably and would benefit from normalisation (centering and scaling) prior to further visualisation and use in developing predictive algorithms.

Figure 1 provides a facet wrap of density plots for each of the features grouped by diagnosis. The axes have been omitted to simplify the figure. This data visualisation makes it clear that (a) the data are relatively normally distributed and don't require transformation, (b) malignant samples are, on average, larger in size and more abnormal in shape, than benign samples and (c) malignant samples have a greater variance in data than benign samples.

Figure 1: Feature density plots by diagnosis

Table 2: Mean scores

radius_m	texture_m	perimeter_m	area_m	smoothness_m	compactness_m	concavity_m	concave_points_m	symmetry_m	fractal_dim_m
Min.:6.98	Min.:9.7	Min.:43.8	Min.:144	Min.:0.0526	Min.:0.019	Min.:0.000	Min.:0.0000	Min.:0.106	Min.:0.0500
1stQu.:11.70	1stQu.:16.2	1stQu.:75.2	1stQu.:420	1stQu.:0.0864	1stQu.:0.065	1stQu.:0.030	1stQu.:0.0203	1stQu.:0.162	1stQu.:0.0577
Median:13.37	Median:18.8	Median:86.2	Median:551	Median:0.0959	Median:0.093	Median:0.062	Median:0.0335	Median:0.179	Median:0.0615
Mean:14.13	Mean:19.3	Mean:92.0	Mean:655	Mean:0.0964	Mean:0.104	Mean:0.089	Mean:0.0489	Mean:0.181	Mean:0.0628
3rdQu.:15.78	3rdQu.:21.8	3rdQu.:104.1	3rdQu.:783	3rdQu.:0.1053	3rdQu.:0.130	3rdQu.:0.131	3rdQu.:0.0740	3rdQu.:0.196	3rdQu.:0.0661
Max.:28.11	Max.:39.3	Max.:188.5	Max.:2501	Max.:0.1634	Max.:0.345	Max.:0.427	Max.:0.2012	Max.:0.304	Max.:0.0974

Table 3: Worst scores

radius_w	texture_w	perimeter_w	area_w	smoothness_w	compactness_w	concavity_w	concave_points_w	symmetry_w	fractal_dim_w
Min.:7.9	Min.:12.0	Min.:50.4	Min.:185	Min.:0.0712	Min.:0.027	Min.:0.000	Min.:0.0000	Min.:0.156	Min.:0.0550
1stQu.:13.0	1stQu.:21.1	1stQu.:84.1	1stQu.:515	1stQu.:0.1166	1stQu.:0.147	1stQu.:0.114	1stQu.:0.0649	1stQu.:0.250	1stQu.:0.0715
Median:15.0	Median:25.4	Median:97.7	Median:686	Median:0.1313	Median:0.212	Median:0.227	Median:0.0999	Median:0.282	Median:0.0800
Mean:16.3	Mean:25.7	Mean:107.3	Mean:881	Mean:0.1324	Mean:0.254	Mean:0.272	Mean:0.1146	Mean:0.290	Mean:0.0839
3rdQu.:18.8	3rdQu.:29.7	3rdQu.:125.4	3rdQu.:1084	3rdQu.:0.1460	3rdQu.:0.339	3rdQu.:0.383	3rdQu.:0.1614	3rdQu.:0.318	3rdQu.:0.0921
Max.:36.0	Max.:49.5	Max.:251.2	Max.:4254	Max.:0.2226	Max.:1.058	Max.:1.252	Max.:0.2910	Max.:0.664	Max.:0.2075

Table 4: Standard error scores

radius_se	texture_se	perimeter_se	area_se	smoothness_se	compactness_se	concavity_se	concave_points_se	symmetry_se	fractal_dim_se
Min.:0.112	Min.:0.36	Min.:0.76	Min.:7	Min.:0.00171	Min.:0.0023	Min.:0.000	Min.:0.0000	Min.:0.0079	Min.:0.00089
1stQu.:0.232	1stQu.:0.83	1stQu.:1.61	1stQu.:18	1stQu.:0.00517	1stQu.:0.0131	1stQu.:0.015	1stQu.:0.0076	1stQu.:0.0152	1stQu.:0.00225
Median:0.324	Median:1.11	Median:2.29	Median:25	Median:0.00638	Median:0.0204	Median:0.026	Median:0.0109	Median:0.0187	Median:0.00319
Mean:0.405	Mean:1.22	Mean:2.87	Mean:40	Mean:0.00704	Mean:0.0255	Mean:0.032	Mean:0.0118	Mean:0.0205	Mean:0.00379
3rdQu.:0.479	3rdQu.:1.47	3rdQu.:3.36	3rdQu.:45	3rdQu.:0.00815	3rdQu.:0.0324	3rdQu.:0.042	3rdQu.:0.0147	3rdQu.:0.0235	3rdQu.:0.00456
Max.:2.873	Max.:4.88	Max.:21.98	Max.:542	Max.:0.03113	Max.:0.1354	Max.:0.396	Max.:0.0528	Max.:0.0790	Max.:0.02984

2.2 Datawrangling

The ID column is not required for this project and was removed. The diagnosis column was reclassified as categorical data using the base 'as.factor()' function, with 2 levels, one for benign masses (B) and the other for malignant masses (M).

Finally, the data set was reorganised into a list including the outcomes (diagnoses) and a matrix of all of the predictors (feature measurements for each sample). Converting the numeric feature data into a matrix allows for more versatility and efficiency in both further data processing and predictive model development.

2.3 Split data into training/test sets

Prior to normalising the data and exploring distance and clustering of samples and features, the data-set was split into train and test sets in an 80:20 split using the caret function 'createData-Partition' (Kuhn 2019). This was critical to avoiding any influence of the test data (for example, in calculating the column means and standard deviations used to centre and scale the data) which could result in over-fitting the model and over-estimating the predictive accuracy of the algorithm.

The balance of classes was consistent between the train set (malignant = 37.2%) and test set (malignant = 37.4%). Further data exploration was conducted with the train set only.

2.4 Exploring train set samples

The train set was normalised using the sweep function firstly to centre each data point (x) around zero by subtracting the sample mean ($\bar{x}$) by column and then to scale each data point by dividing by the sample standard deviation (S) by column, yielding z-scores (1).

$$z = \frac{(x - \bar{x})}{S} \quad (1)$$

One way of measuring the variance between samples is to calculate the Euclidean distance, in this case in 569 (the number of samples) dimensional space. The Euclidean distance between two observations, x1 and x2, is defined below (2).

$$d(x1, x2) = \sqrt{\sum_{i=1}^{569} (x1_{,i} - x2_{,i})^2} \quad (2)$$

The average distance between all samples included in the train set was 6.99. Benign samples were closer to each other (6.39) than to malignant samples (8.53). Of note, benign samples were also closer to each other than malignant samples (8.02) were from each other, indicating greater variance in the measured features in malignant cells. The heatmap shown in Figure 2 provides a visualisation of the distances between samples. A sequential palette ("Blues" from the RColorBrewer package, see Neuwirth (2014)) was selected to highlight the difference between those pairings that are further from each other (dark colours) and those that are close together (light colours). The heatmap shows that samples are relatively well clustered by class (green: benign, red: malignant) and that benign samples are closest to each other (light blue hue in the bottom right quadrant) and furthest from malignant samples (more dark blue hue in the top right and bottom left quadrants).

2.5 Exploring train set features

Various techniques are available for machine learning without the supervision of the outcomes (diagnosis). These are known as unsupervised methods and can be used to cluster, select or extract features to include in the prediction algorithm. The benefit of extracting an individual feature, for example based on low variance across samples or high correlation with one or more other features is to reduce noise and processing time as well as minimising the risk of over-fitting the algorithm.

2.5.1 Variancewithinfeatures

The nearZeroVar function within the caret package demonstrated that none of the features in the train set had either zero or near zero variance. The lowest percentage of unique values for any feature was 77.533 and the mean of all features was 91.601. Moreover, the mean frequency ratio was 1.81 with 80% of all features having a frequency ratio score of 2 or less. This analysis supports the inclusion of all features in the algorithm based on their within feature variance.

2.5.2 Hierarchicalclustering

The train set was transposed so that the features were moved from the columns of the matrix to the rows and the dist function operated to calculate the Euclidean distance between each feature. Hierarchical clustering was performed using the hclust function from the stats package and resulted in the dendrogram shown in Figure 3 (colour coded to show 7 clusters).

The features that are closest together are those that measure nuclear size, including radius, perimeter and area. Those features that measure shape rather than size are further apart from each other,

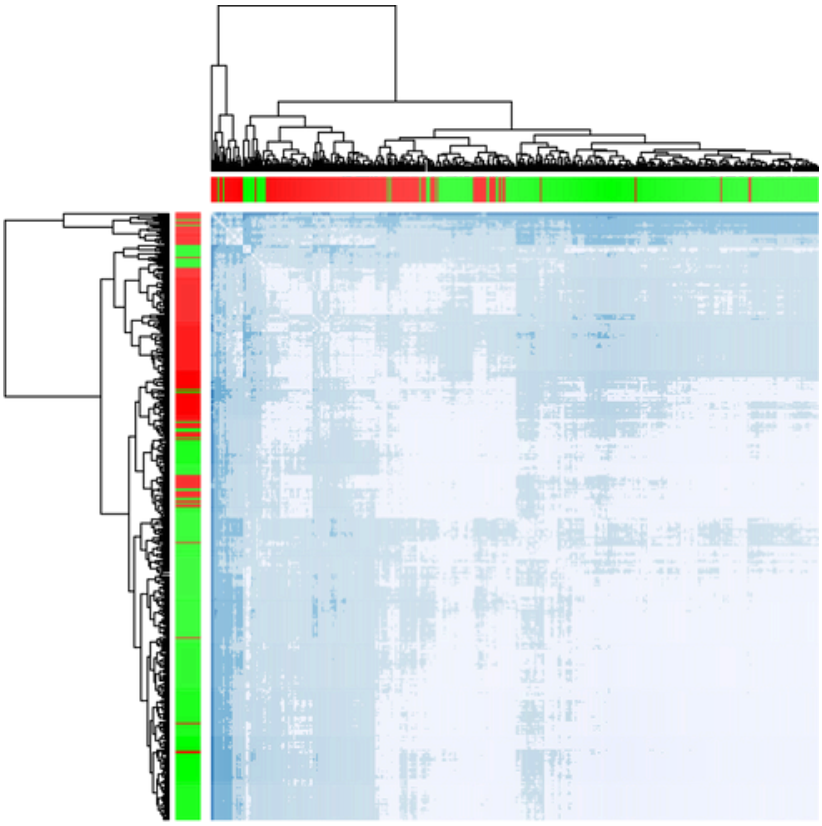


Figure 2: Heatmap of distance between benign (green) and malignant (red) samples

although concavity, compactness and the number of concave points are relatively near to each other. Fractal dimension, smoothness and symmetry are the features with the greatest distance, both from each other and the other features.

2.5.3 Correlation between features

In addition to exploring the variance within features and the distance between features, it is helpful to understand the degree of correlation between features before deciding which features ton include in the development of a training algorithm. In particular, unsupervised methods for predictive modelling can benefit from excluding features that are highly correlated with each other from the training set. Here the Pearson correlation coefficient (3) was used to measure correlation, r between two features, x1 and x2.

$$r_{x1,x2} = \frac{\sum_{i=1}^{n}(x_{1,i} - \bar{x}_1)(x_{2,i} - \bar{x}_2)}{\sqrt{\sum_{i=1}^{n}(x_{1,i} - \bar{x}_1)^2}\sqrt{\sum_{i=1}^{n}(x_{2,i} - \bar{x}_2)^2}} \quad (3)$$

The heatmap in Figure 4 plots the absolute correlation coefficients between each of the 30 features included in the data-set. A sequential palette (“RdPu” from the RColorBrewer package, see

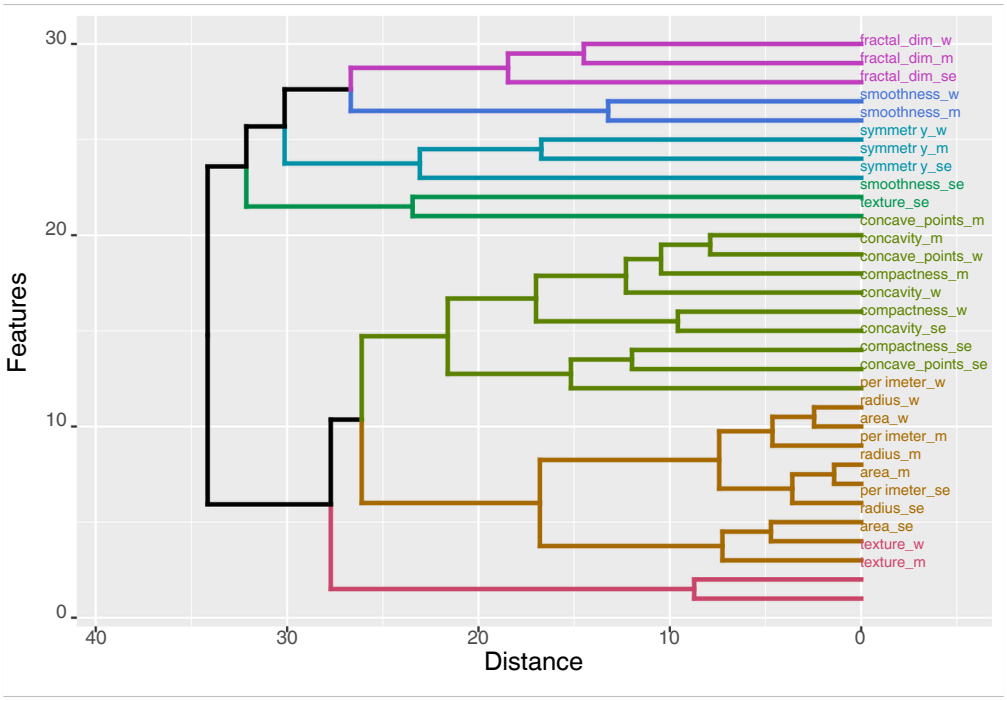


Figure 3: Dendrogram of hierarchical clusters of features

Neuwirth (2014)) was selected to highlight the difference between those pairings that are highly positively or negatively correlated (dark colours) and those that are not very correlated with each other (light colours).

Overall, there is a low level of correlation between features; indeed, the mean correlation coefficient of features in the train set is 0.43. The greatest correlation is between the various measures of cell size, namely radius, perimeter and area and, to a lesser extent between some of the measures of cell shape, namely fractal dimension, symmetry and smoothness.

Ten features have a correlation of 0.9 or more, namely concavity_m, concave_points_m, perimeter_w, radius_w, perimeter_m, area_w, radius_m, perimeter_se, area_se, and texture_m. Excluding these features from unsupervised methods of developing the predictive algorithm may be b eneficial.

2.5.4 PrincipalComponentAnalysis

Principal component analysis (PCA) is a technique for transforming data-sets in order to reduce dimensionality without reducing the number of features by identifying the principal components which explain as much of the data variance as possible. PCA can be used to improve visualisation of multidimensional data and, potentially, to improve the predictive accuracy of classification models.

Table 5 shows the standard deviation (Eigenvalues), proportion of variance and cumulative proportion of variance for the first 10 principal components. The first principal component (PC1) accounts for 45% of the total variance within the data-set, the first two components account for almost 64% of the cumulative variance and the first 10 principal components account for more than 95% of the cumulative variance within the data-set.

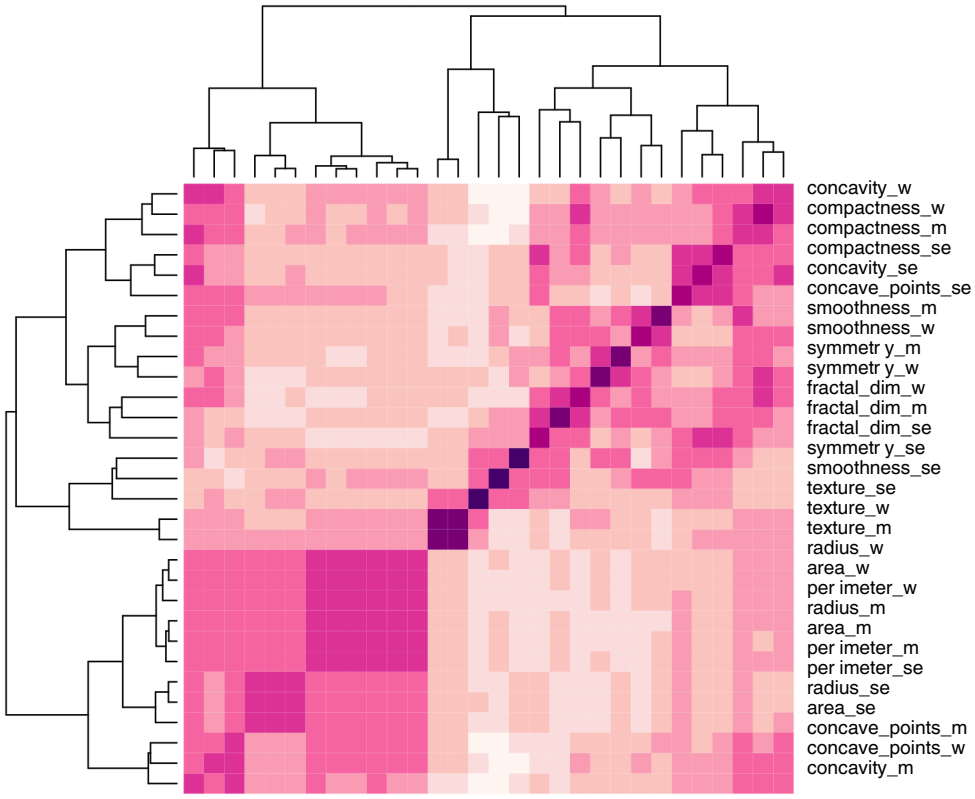


Figure 4: Heatmap of correlation between features

Table 5: First 10 Principal Components

	PC1	PC2	PC3	PC4	PC5	PC6	PC7	PC8	PC9	PC10
Standard deviation	3.692	2.354	1.654	1.406	1.243	1.103	0.844	0.706	0.616	0.612
Proportion of Variance	0.454	0.185	0.091	0.066	0.052	0.041	0.024	0.017	0.013	0.012
Cumulative Proportion	0.454	0.639	0.730	0.796	0.848	0.888	0.912	0.929	0.941	0.954

Figure 5 is a series of box plots for each of the first 10 principal components grouped by diagnosis. In most cases the spread is greater for malignant masses than for benign masses. PC1 is the only component for which the interquartile ranges do not overlap. Principal component analysis does not take into account the classification of data, in this case the diagnosis assigned to each sample.

Figure 6 is a two-dimensional scatter plot of the first two principal components, data-points coloured
red if classified as benign and blue if classified as malignant. The graph shows that the malignant data-points are more spread out than the benign data-points and that more of the variance can be accounted for on the x-axis (PC1) than on the y-axis (PC2). Ellipses help to visualise this even better, firstly with a larger ellipse for malignant data-points than for benign data-points and considerable separation of data by classification, despite some overlap. This analysis support the use of PCA in algorithm development to predict diagnosis from this data-set.

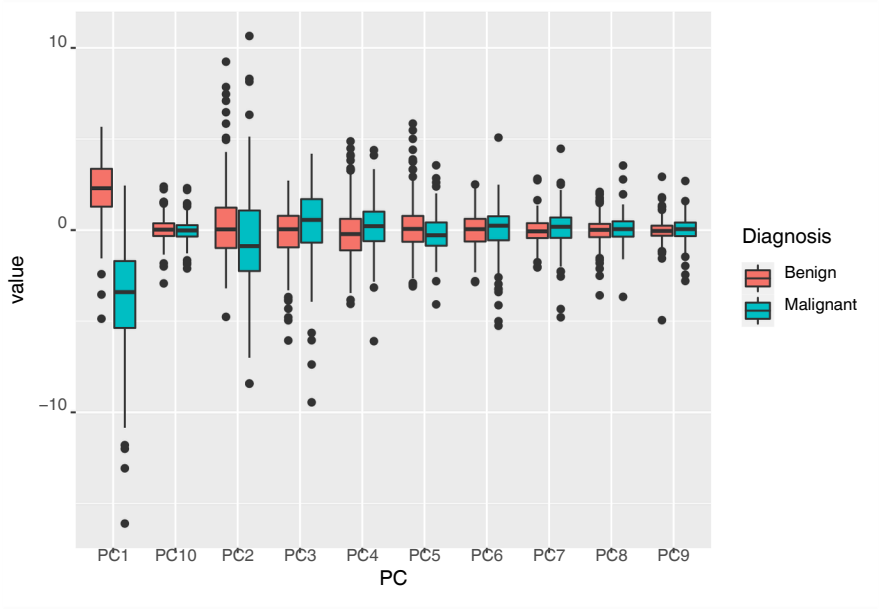


Figure 5: Box plots of top 10 PCs by diagnosis

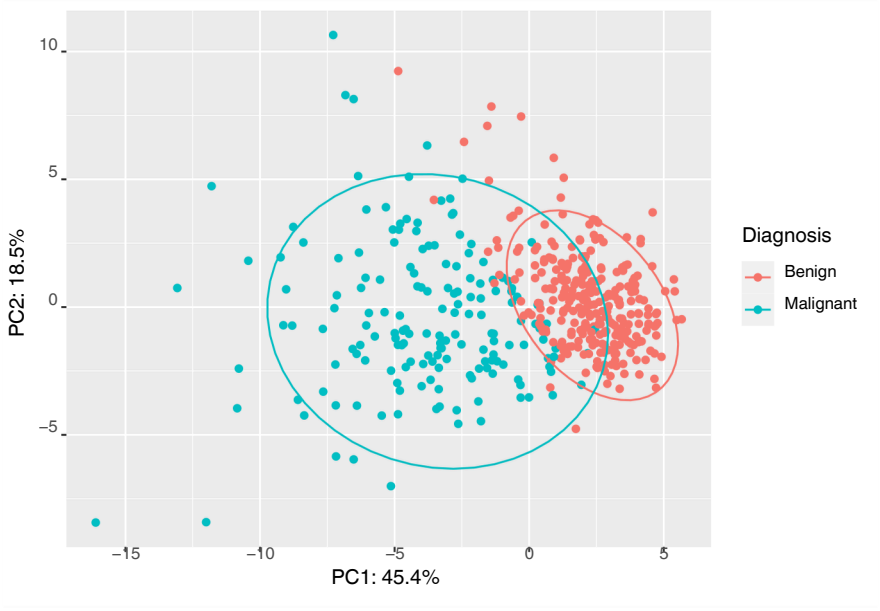


Figure 6: Scatter plot of PC1 and PC2 by diagnosis

3 Methods

3.1 Pre-pro cessing

The exploratory analysis of the Wisconsin breast cancer (diagnostic) data-set revealed patterns across both samples and features that support the use of machine learning techniques to develop predictive algorithms, including the use of both supervised and unsupervised models.

Prior to testing any models, it was necessary to normalise the test data to reflect changes made to the train data, i.e. to centre and scale the data using the column means and standard deviations calculated from the train set. Taking this approach allowed for the test set to be normalised in a way that was consistent with the train set but without allowing the test data itself to influence the training of the algorithm.

An empty data frame was generated in which to store key performance metrics for each model developed, namely the overall accuracy of the model (4), the sensitivity, or true positive rate (5), the specificity, or true negative rate (6).

$$\text{Accuracy} = \frac{\text{TruePositive}+\text{TrueNegative}}{\text{TruePositive}+\text{TrueNegative}+\text{FalsePositive}+\text{FalseNegative}} \quad (4)$$

$$\text{Sensitivity} = \frac{\text{TruePositive}}{\text{TruePositive}+\text{FalsePositive}} \quad (5)$$

$$\text{Specificity} = \frac{\text{TrueNegative}}{\text{TrueNegative}+\text{FalsePositive}} \quad (6)$$

The F1 score, a harmonic mean of precision, or positive predictive value, and sensitivity (7) is another measure of a model's accuracy and was also included. To aid analysis, the false negative rates (8) and false positive rates (9) were also computed.

$$\text{F1score} = \frac{2(\text{TruePositive})}{2(\text{TruePositive})+\text{FalsePositive}+\text{FalseNegative}} \quad (7)$$

$$\text{FalseNegativeRate} = \frac{\text{FalseNegative}}{\text{FalseNegative}+\text{TruePositive}=1-\text{Sensitivity}} \quad (8)$$

$$\text{FalsePositiveRate} = \frac{\text{FalsePositive}}{\text{FalsePositive}+\text{TrueNegative}=1-\text{Specificity}} \quad (9)$$

Cross-validation is an important technique used to measure performance of a model without recourse to the test data-set, allowing the test set to be reserved for the final hold-out test of each model and minimising the risk of over-fitting. It is also a useful technique for tuning parameters for those models that require it (e.g., to tune the number of neighbours, k, to include in a k-nearest neighbours model). The caret package provides a convenient method for cross-validation that can be defined in advance, using the 'trainControl' function and applied to each model as required (Kuhn 2019). Here, the train control parameters were set to apply 10-fold cross-validation, repeated ten times.

For the purposes of measuring performance within the resamples only the final predictions and summary performance metrics (accuracy and kappa scores) were saved based on the tuned parameters where applicable. Kappa scores are another measure of the agreement between observed (po) and expected values (pe) and, unlike overall accuracy, take account of the chance that a prediction (or observed value) will match the true (or expected) value (10). Kappa scores may be negative, where the model performs less well than chance and, generally speaking, a kappa score of more than 0.8 is considered to represent very strong agreement (Landis and Koch 1977).

$$\kappa = \frac{po - pe}{1 - pe} \tag{10}$$

3.2 Random sampling

The first model to be tested involved random sampling with equal probability for each outcome, i.e. equivalent to a coin toss. Given the imbalance in prevalence of benign and malignant samples in the data-set, a second model was built with weighted random sampling, where the prevalence of each class of samples was used to define the probability of each outcome within the random sample.

3.3 Unsupervisedlearning

3.3.1 k-meansclustering

In most cases, where labels exist to support the use of supervised models, this is the preferred option given these models will invariably yield more accurate predictions. The exploratory analysis showed that the Euclidean distance (2) between features may be predictive of outcome and that hierarchical clustering does allow for unsupervised clustering of features based on separation of data achieved based on their distance from each other. It also showed that features are relatively distinct from one another but that a few have high correlation with one another.

k-means clustering is another form of unsupervised modelling where the number of clusters is defined in advance. k-means clustering is an attractive option for large data-sets because it is a relatively simple approach to clustering and as such is computationally faster than hierarchical clustering. Given that the clustering is informed by distance, it is more effective when data are continuous, as they are in this data-set, than when the data are categorical. The first step is to develop a k-means prediction function which calculates the distance between data points and cluster centres by iteration in order to assign each row of data (i.e. each sample) to one of k clusters, based on minimum distance to the cluster centre. Here, the number of clusters (or centres) was defined as two as there are two classes of outcome to be predicted.

Two version of the k-means model were developed. The first used the normalised data from the full train data-set. The second selected out those features which were highly correlated, i.e. the ten features where the correlation coefficient exceeded the 0.9 cutoff defined in the exploratory analysis. k-means clustering places greater weight on larger clusters and variables that are highly correlated (that form a large cluster) may therefore carry greater weight in the prediction algorithm (Sambandam 2003; Biesiada and Duch 2007; Chormunge and Jena 2018).

In each case, the number of random sets to be chosen was set at 25 using the ‘nstart’ argument within the ‘kmeans’ function. This approach helps to introduce stability to the model outcome given that it can be sensitive to the fact that the initial centre is chosen at random (Irizarry 2020).

3.4 Supervised learning

3.4.1 Generativemodelling

Generative models are supervised machine learning techniques that model how the entire data-set, including both predictors and outcomes are distributed and use the joint probability distribution in order to predict the conditional probability of one outcome or another. The most general generative model is the Naive Bayes model which is based on the Bayes rule, where $f_{\mathbf{X}|Y=1}$ and $f_{\mathbf{X}|Y=0}$ are the distribution functions of the predictor $\mathbf{X}$ with binary outcomes, Y = 1 and Y = 0 (11).

$$p(\mathbf{x})=Pr(Y=1|\mathbf{X}=\mathbf{x})=\frac{f_{\mathbf{X}|Y=1}(\mathbf{x})Pr(Y=1)}{f_{\mathbf{X}|Y=0}(\mathbf{x})Pr(Y=0)+f_{\mathbf{X}|Y=1}(\mathbf{x})Pr(Y=1)} \quad (11)$$

The Naive Bayes model assumes that all features within the data-set are equally important and independent (Kurama 2020). Whilst this is a naive assumption that is unlikely to be true for a given set of data, it is typically good enough for the purposes of classification (Kurama 2020). The 'nb' method provided in the caret package includes three tuning parameters (Kuhn 2019) each of which was tuned using resampling during cross-validation of this model: laplace smoothing (0 or 1), kernel distribution (true or false) and adjustment (0 or 1).

Other generative models include linear discriminative analysis (LDA) and quadratic discriminative analysis (QDA). LDA has the benefit of serving to reduce the dimensionality of the data (similarly to PCA) and to classify the data for predictive purposes. LDA assumes that the data are normally distributed and that the correlation structure is the same for all classes (Döring 2018).

On the other hand QDA assumes that the distributions are multivariate normal, and cannot be used

3.4.2 Discriminativemodelling

3.4.2.1 Logistic Regression Logisticregressionisthemostcommonlyusedformofgeneralised linear model (GLM). Linear regression assumes that the predictor, X, and the outcome Y , follow a bivariate normal distribution such that the conditional expectation, i.e. the expected outcome Y for a given predictor X, fits the regression line (12).

$$p(x)=Pr(Y=1|X=x)=\beta 0+\beta 1x \quad (12)$$

Logistic regression is an extension of linear regression, where g is a function that transforms the probability, p, to log odds ($g(p) = \log \frac{p}{1-p}$) such the conditional probability can be modelled as below (13). A logistic regression model was developed using the caret package to train the normalised train set before predicting outcomes in the normalised test set. In addition, the model was also run incorporating the outputs from PCA via caret's pre-processing functionality. Dimension reduction is most useful in highly dimensional data but there is some evidence in the literature that dimension

reduction via PCA can also improve the predictive accuracy of models such as logistic regression (Hsu, Huang, and Chen 2014; Sabharwal and Anjum 2016).

(13)

3.4.2.2

$$g\{Pr(Y = 1|X = x)\} = \beta 0 + \beta 1 x$$

Nearestneighbourmodel Thek-Nearestneighbourmodel(kNN)isasimpleapproach to supervised machine learning that assumes proximity equates to similarity, once again measuring the Euclidean distance (2) between two points in multidimensional data. Unlike hierarchical and k-means clustering, the KNN model is a form of supervised learning, i.e. it relies on and makes use of the diagnosis labels in the training set in order to predict diagnosis in an unlabelled test set.

Whereas in k-means clustering, the k represents the number of clusters, or centres, within the data, in the kNN model , k represents the number of neighbours for any given data-point. As with the use of bins in smoothing, larger values of k result in smoother estimates. k is a tuning parameter within the train function for the kNN model (Kuhn 2019), and cross-validation within the train set was used to tune a value for k between 1 and 30 in increments of 2 to optimise the model before using it to predict outcome in the test set.

3.4.2.3

$$\text{Gini}(j) = \sum_{k=1}^{K} \hat{p}_{j,k} (1 - \hat{p}_{j,k})$$

(14)

One of the key challenges with decision trees is that they are prone to over-training and can therefore be unstable to changes in training data. Random forests address this challenge by effectively creating an ensemble (‘forest’) of multiple decision trees via bootstrap aggregation, and averaging the predictions from each of these trees to form a final prediction. Here, the ‘rf’ method within the caret package was used and the number of randomly selected predictors to include in each decision tree was tuned within the train set using the ‘mtry’ tuning parameter via cross-validation (Kuhn 2019). Other random forest methods also allow for tuning of the minimum number of data-points to include in each decision tree but were not utilised here.

3.4.2.4 Neuralnetworkmodel Anotheroptionfordealingwithmultidimensionaldata,and particularly with non-linearity, is neural network modelling. As such, this type of algorithm has found particular application in dealing with complex machine learning tasks such as image pro- cessing, particularly with the multi-layer neural networks. The more layered the network, however,

the more computational resource it requires and the greater the risk of over-fitting to a training data set (Lawrence and Giles 2000; Hayashi, Sakata, and Gallant 1990). The simplest forms of neural network, known as single-layer neural networks, are only equipped to deal with linear data.
These algorithms take multidimensional inputs (xi), apply a weighting (wi) before summing them in order to classify the output yi (15).

$$y_i = \sum i \, wixi \tag{15}$$

Neural networks can also be established for unsupervised learning but, as this project has labelled data, a supervised approach was developed using the single hidden layer neural network method, 'nnet' that is available within the caret package (Kuhn 2019). Unlike the simplest neural network this model has three layers and, consequently, can handle non-linear data. Whilst tuning parameters to optimise the number of hidden units (size) and weight decay (decay) are available with this method, they were not deployed here.

3.4.3 Ensemblemodel

Ensembles are combinations of individual model predictions that seek to improve both stability and accuracy of the final result (Wichard 2006), just as the random forest algorithm uses combinations of individual decision trees. There is no established convention for selecting which models to include in the ensemble. One approach is to establish a performance cutoff within the training sets, via cross-validation, in order to avoid selection based on performance in the test set (Wichard 2006; Irizarry 2020).
Here a decision was made to simply include only those algorithms that utilised supervised learning, given their inherent advantage where labelled data are available. Thus, an ensemble was created by combining the predictions from each of the naive Bayes, linear discriminant analysis, quadratic discriminant analysis (with and without principal component analysis), logistic regression (with and without PCA), k-nearest neighbours, random forest and neural network models. The final prediction from the ensemble was determined by majority vote.

4 Results

4.1 Overallperformance

Table 6 provides the key performance metrics for each of the models tested in this project. Overall, the results confirm the expectation established during the exploratory analysis, i.e. that specificity will be easier to achieve with this data than sensitivity. Six models achieved a specificity of 1, but only two of these achieved the same level of sensitivity.

Unsurprisingly, guessing the outcome through random sampling was the least accurate method of predicting diagnosis, with an overall accuracy of only 0.55 and slightly better sensitivity (0.60) than specificity (0.51). Clearly, with 17 out of 43 malignant samples being incorrectly classified as benign and 35 out of 72 benign samples being incorrectly classified as malignant, this is not an effective method for predicting diagnosis.

Weighting the sampling to account for the prevalence of malignant samples within the data-set did improve specificity as anticipated (i.e. reduced the false positive rate from 49% to 40%) but dramatically reduced the sensitivity to 0.23 (i.e. increased the false negative rate from 40% to 77%) and as a result, in fact, reduced the overall accuracy to 0.46.

Table 6: Key performance metrics for each model

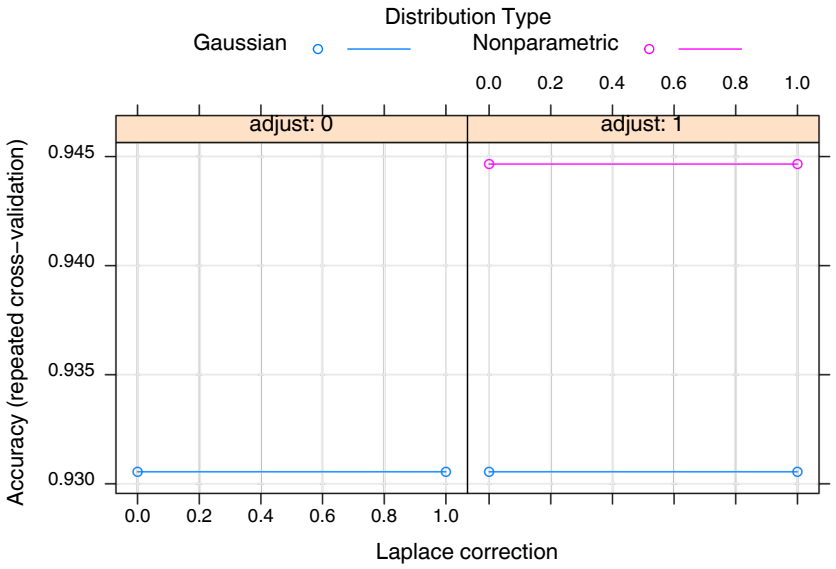
Method	Accuracy	Sensitivity	Specificity	F1	FNR	FPR
Random sampling						
Randomsample	0.55	0.60	0.51	0.50	40%	49%
Weightedrandomsample	0.46	0.23	0.60	0.24	77%	40%
Unsupervised models						
K-meansclustering	0.89	0.86	0.90	0.85	14%	10%
K-means(withouthighlycorrelatedfeatures)	0.76	0.63	0.83	0.66	37%	17%
Generative models						
NaiveBayes	0.92	0.93	0.92	0.90	7%	8%
LinearDiscriminantAnalysis	0.96	0.88	1.00	0.94	12%	0%
QuadraticDiscriminantAnalysis	0.97	0.98	0.96	0.95	2%	4%
QuadraticDiscriminantAnalysis(withPCA)	0.97	0.95	0.99	0.96	5%	1%
Discriminative models						
Logisticregression	0.97	0.91	1.00	0.95	9%	0%
Logisticregression(withPCA)	0.98	0.98	1.00	0.98	2%	0%
KNearestNeighbour	0.97	0.92	1.00	0.96	7%	0%
RandomForest	0.95	0.98	1.00	0.97	5%	3%
NeuralNetwork	0.97	0.93	1.00	0.96	0%	0%
Ensemble						
Ensemble	1.00	1.00	1.00	1.00	0%	0%

Note:
FNR = false negative rate; FPR = false positive rate; PCA = principal component analysis

This exercise highlights the importance of balance within the data-set to the performance of classification models. It also reinforces the importance of the F1 score in evaluating performance of models dealing with imbalanced data-sets. Random sampling and weighted random sampling resulted in F1 scores of 0.50 and 0.24 respectively.

k-means clustering improved accuracy, with an F1 score of 0.85 but still not to an acceptable level, with a false negative rate of 14% and a false positive rate of 10%. Of note, removing the highly correlated features (i.e. those with a correlation coefficient above 0.9) from the data-set reduced the accuracy of the k-means clustering model, yielding a reduced F1 score of 0.66 and reducing both the specificity and, in particular, the sensitivity of the model.

The use of supervised learning techniques substantially improved the accuracy of predictions, with each subsequent model achieving an overall accuracy of more than 0.9. Amongst the generative models, Naive Bayes achieved an overall accuracy of 0.92 and an F1 score of 0.90 with a balance of sensitivity and specificity. Of note, the Naive Bayes model performed best when the usekernel parameter was set to TRUE and the associated adjustment was set to 1, which indicates that a normal (Gaussian) distribution is not the best way to estimate the conditional probabilities (see Figure 7).



Linear discriminant analysis and quadratic discriminant analysis improved on the performance of the Naive Bayes model, with F1 scores of 0.94 and 0.95 respectively. The LDA model achieved a specificity of 1.00 (i.e. FPR of 0%) but this was offset by reduced sensitivity (0.88), i.e. an unacceptable FNR of 12%.

The QDA model delivered a better balance between FNR (2%) and FPR (4%). Of note, dimension reduction through pre-processing the training data with PCA improved the specificity of the QDA model, reducing the FPR to 1% but it reduced the sensitivity, i.e. increased the FNR to 5%.

The discriminative models of supervised learning were the best performing models with this data-set. Logistic regression achieved an overall accuracy of 0.97. This was improved further to 0.99 with dimension reduction using PCA by improving the sensitivity of the model, achieving FNR and FPR of only 2% and 0% respectively.

The nearest neighbours model performed best when the number of neighbours, k, was defined as three (see Figure 8). On this basis, the overall accuracy of 0.97 but with lower sensitivity (0.93).

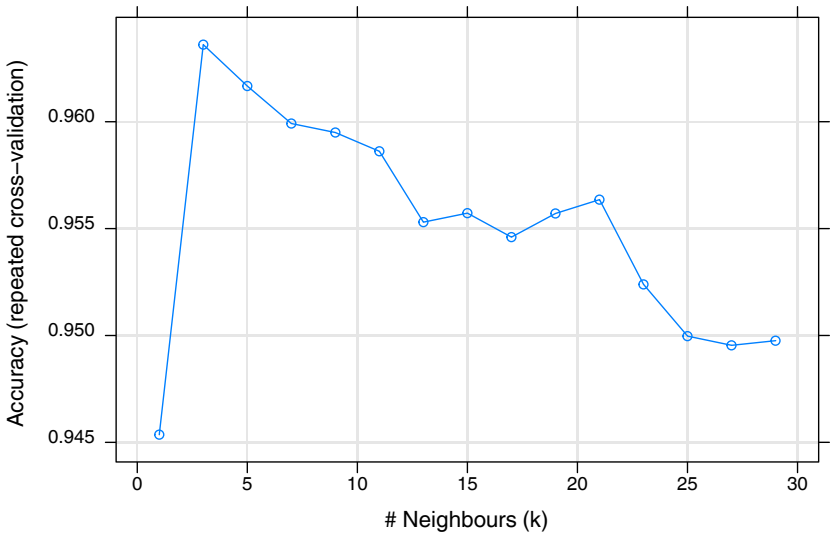


Figure 8: Tuning results for nearest neighbour model during cross-validation

The random forest model was not very sensitive to the number of randomly selected predictors included in each decision tree it was tuned for, but performed marginally best when mtry was 11 (see Figure 9), with overall accuracy of 0.97, sensitivity of 0.95 and specificity of 0.97.

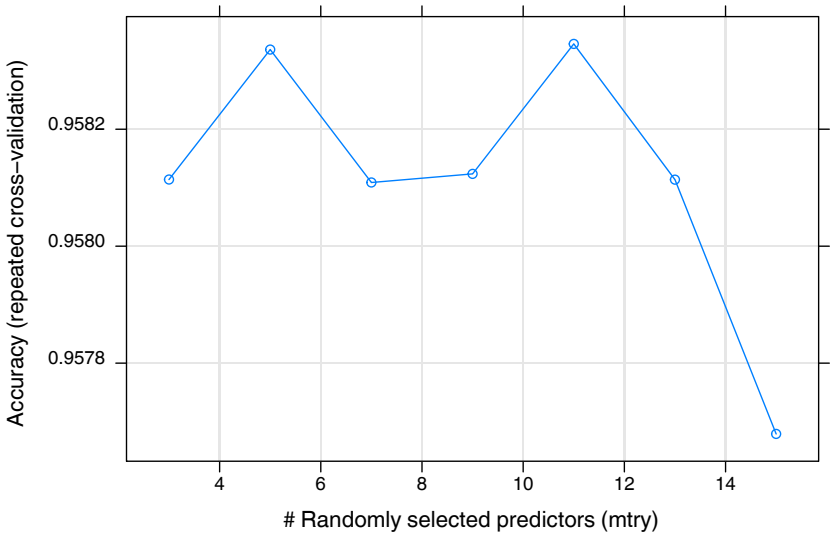
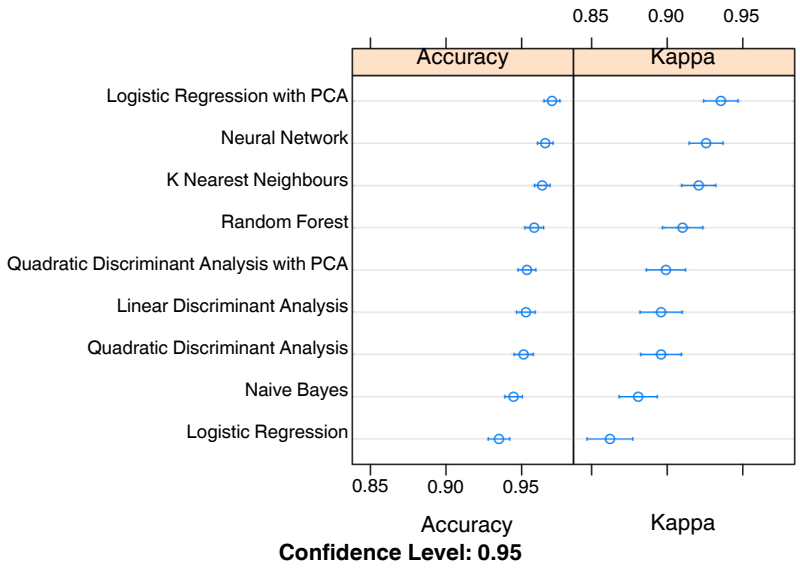


Figure 9: Tuning results for random forest model during cross-validation

The final individual model to be trained and tested was the neural network. This performed very well with an overall accuracy of 1.00 and an F1 score of 1.00. Finally an ensemble of all of the supervised learning models, including both generative and discriminative approaches, also delivered an overall accuracy of 1.00 and an F1 score of 1.00.

4.2 Cross-validation

provides dot plots summarising the accuracy and Kappa scores for each of the supervised learning models for which cross-validation was conducted, ranked in order of performance. Overall, these results are consistent with those observed in the test set predictions; the neural network was the highest performing model followed by logistic regression with PCA and the random forest mo del.



Confidence Level: 0.95

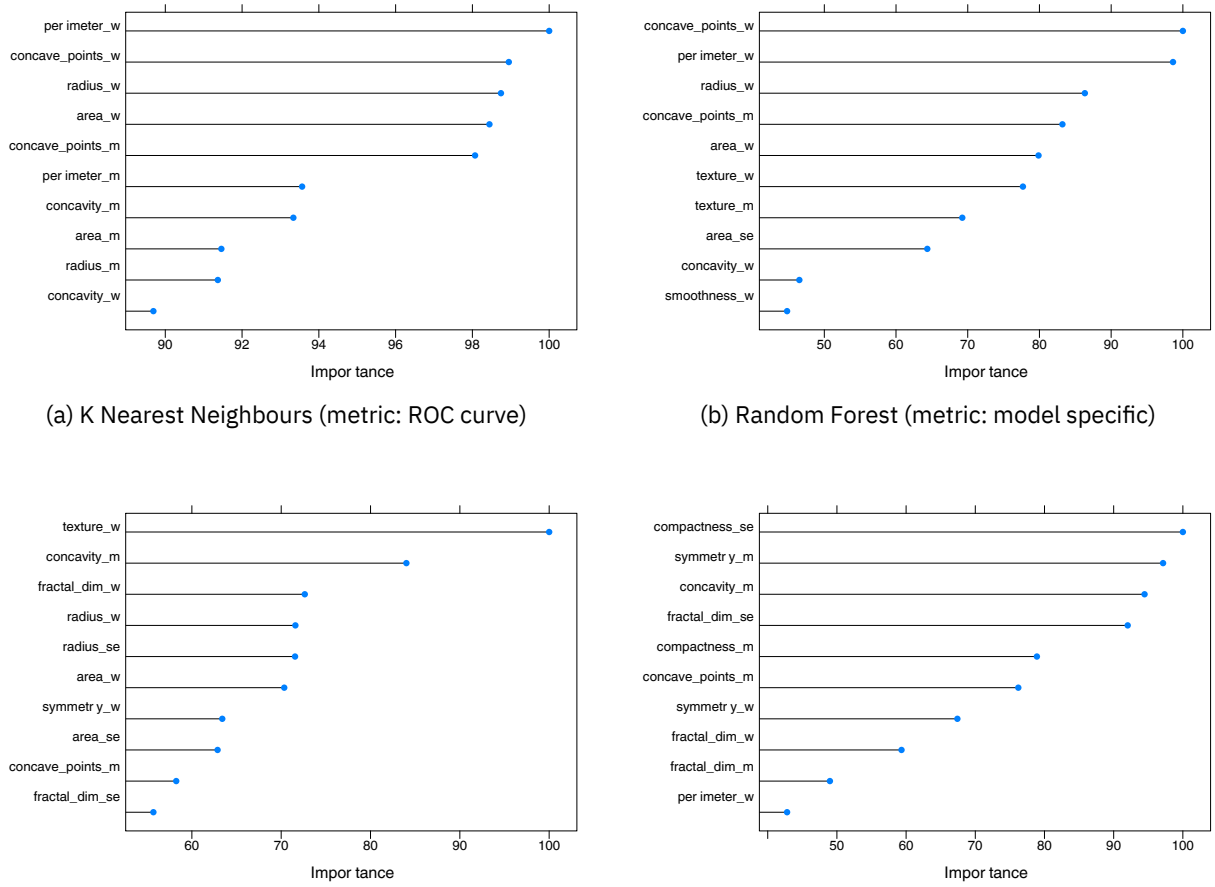
Mean accuracy scores were consistently above 0.9 for all models, with the lowest mean accuracy achieved by logistic regression (mean: 0.94) and the highest for the neural network model (mean: 0.97) and logistic regression with PCA (mean: 0.97).

Mean kappa scores were consistently lower but were still above 0.8 for all models, and above 0.9 for logistic regression with PCA (mean: 0.94) and for the nearest neighbours (mean: 0.92), random forest (mean: 0.91 and neural network (mean: 0.93 models. In all bar the Naive Bayes model, the accuracy scores achieved during cross-validation did not exceed those achieved in the final predictions run within the test set, indicating that over-fitting of algorithms during training was not a major concern, and providing confidence that model performance would be sustained beyond the current data-set.

Of note, pre-processing with principal component analysis was shown to improve the performance of quadratic discriminant analysis and, in particular, logistic regression. The latter achieved the lowest mean accuracy and kappa scores without PCA (mean accuracy: 0.94, mean kappa: 0.86) compared with the highest mean scores with PCA (mean accuracy: 0.97, mean kappa: 0.94).

4.3 Variableimportance

The caret package includes the ‘varImp’ function to compute the importance of each feature to the performance of the model during cross-validation within the training data-set (Kuhn 2019). The



(a) K Nearest Neighbours (metric: ROC curve)

(b) Random Forest (metric: model specific)

(c) Neural Network (metric: model specific)

(d) Logistic Regression (metric: model specific)

Figure 11: Variable importance

top ten features in order of variable importance for the nearest neighbours, random forest, neural network and logistic regression models are shown in Figure 11. The importance scores are scaled to aid comparison across models, such that the most important feature received a score of 100.

The worst performing discriminative model, logistic regression appears to be most reliant on mean and standard error scores for features related to nuclear shape in the top five variables of importance, namely, compactness_m, concavity_m, symmetry_m, compactness_se, and fractal_dim_se.

Looking at the best performing models, by comparison, the worst scores feature heavily in the top five variables of importance for the nearest neighbour model (4 out of 5), the random forest model (4 out of 5) and the neural network model (3 out of 5).

The top 5 most important features were the same in both the nearest neighbour model (concave_points_m, radius_w, perimeter_w, area_w, and concave_points_w) and the random forest model (concave_points_m, radius_w, perimeter_w, area_w, and concave_points_w). In contrast, the most important variables in the neural network model were concavity_m.

5 Discussion

In this Project ,the various supervised learning methods delivered high levels of overall accuracy and F1 scores reflecting good performance in terms of both sensitivity and specificity.
Discriminative models are generally preferred to generative models for developing classification algorithms and they performed better here, with the naive Bayes model in particular underperforming the other supervised learning algorithms.

Principal component analysis is a highly effective method for reducing the number of dimensions without a proportional loss in the variance within the data and can help improve the computational efficiency of models with very large data sets. Arguably, this particular data set, with 30 features and 569 samples is not large enough to need an improvement in efficiency. That said, the analysis demonstrated that dimension reduction can be achieved such that selecting the top third of principal components only lost 5% of the variance within the data. Two of the models were subjected to dimension reduction via PCA using pre-processing; in the case of quadratic discriminant analysis, specificity was improved but this was offset by reduced sensitivity. On the other hand, PCA maintained the perfect specificity and improved the sensitivity of logistic regression, thus improving overall accuracy to 0.99. Logistic regression is vulnerable to high levels of correlation between multiple variables as well as the effects of high dimensionality, both of which can be alleviated using PCA.

The nearest neighbours and random forest models were both tuned using tuning parameters available within the caret package in order to optimise their performance but were otherwise not subjected to additional data pre-processing. The nearest neighbours model performs less well with imbalanced data because of its reliance on distance between data points in order to discriminate between classes. In particular, this can reduce the sensitivity of the model to detecting the minority class as evidenced by the fact that this model had perfect specificity but a 7% false negative rate.

The neural network was the only individual model to correctly predicted the diagnosis for each of the samples in the test set. Neural networks are particularly strong with non-linear and complex data hence their popularity with image processing. Of note, the variable importance analysis showed the shape features were more important with this model whereas the size features dominated the nearest neighbour and random forest models. The apparent ability of the neural network to

iDespite the success of the neural network, there is merit in selecting the ensemble of supervised models as the preferred algorithm given that it also correctly predicted diagnosis and has the benefit of mitigating the risk of over-training with an individual model, making it more likely to provide reproducible results in different data-sets that include the same feature information.

The current dataset suffers from a number of inherent biases that represent possible limitations to the reproducibilty of the performance achieved here. The samples were collected from a single site and from a consecutive series of patients. Operator biases will have included those responsible for conducting the biopsies, digitising the images to measure each of the features and even the clinical diagnoses made to classify each sample as benign or malignant. The methods for capturing nuclear size and shape information in 1995 were relatively rudimentary and more advanced image processing techniques available today would complement the complex machine learning algorithms (such as convolutional neural networking) now available to capture differences between benign and malignant samples.

Finally, the growing knowledge of cancer biomarkers as well as the characterisation of circulating tumour cells has contributed to advances in the use of liquid biopsy as a less invasive clinical tool than conventional biopsy techniques (Alix-Panabières 2020). These advances provide a richness of data that is tailor made for machine learning (Mouratidis 2019; Savage 2020), not only for diagnosis of primary disease but also to measure prognostic risk, in order to shape treatment choice, and to detect recurrence of metastatic disease (De Rubis, Rajeev Krishnan, and Bebawy 2019).

6 Conclusions

An exploratory analysis of the data revealed that measures of both distance and correlation of nuclear features could be useful in both clustering and classifying individual samples. All of the models developed performed better than random sampling but supervised learning was more accurate than unsupervised learning, and discriminative models were more effective than generative models. The neural network was the most successful individual model, with perfect accuracy within the test data and this performance was matched with an ensemble of supervised models.

These results confirm the potential of machine learning to accurately predict diagnosis of breast cancer using samples obtained via fine needle aspiration biopsy, with high levels of both sensitivity and specificity.

Advances in both medical research and data science, such as developments in the use liquid biopsy and the refinement of image processing techniques, are likely to increase the clinical utility of machine learning to support early diagnosis of breast cancer and, ultimately, to improve patient outcomes.