MACHINE LEARNING FOR DISEASE TREATMENT RESPONSE PREDICTION

Chang Neng Chek, Peng Yang How, Ser Vin Chan, Alden Zheng Heng Sia, Wei Sheng Ooi

School of Computer Science, University of Nottingham, United Kingdom

ABSTRACT

The most prevalent cancer among women and the main cause of the rising death rate in the UK is breast cancer. Chemotherapy is commonly used in treatment strategies to reduce locally advanced tumours. However, this treatment is toxic and is not effective for everyone. This paper is a relative study on the implementation of multiple machine learning pipelines to predict the pathological complete response (PCR) and relapse-free survival (RFS) from the dataset obtained from the American College of Radiology Imaging Network. These predictions aim to provide better patient satisfaction and treatment to patients. Classification accuracy, f1-score, mean absolute error and root mean square error will be measured to compare the models. These implementations are coded in python. The results have shown that MLP with the LinearSVC method and KNN with the SBS method have the best performance for predictive analysis.

1. INTRODUCTION

Chemotherapy is a drug treatment which commonly used to reduce the size of tumour before surgery. However, chemotherapy also carries a risk of side effects as it can damage some healthy cell and it might not be effective to everyone. Therefore, this report aims to predict the pathological complete response (PCR) and relapse-free survival by using machine learning method based on clinically and magnetic resonance images (MRI) features.

2. RELATED WORK

Bhardwaj and Hooda apply nine types of different classification models which are Adabag, Adaboost, RF, Voted Perceptron Neural Network, SMO, MLP, Logistic, Bayes Net and Nave Bayes predict the PCR outcome [1]. The result shows that Neural Network has the lowest accuracy score which is below 70%. The Adabag and Adaboost obtain the highest accuracy which is 97.07%. Karim et al. used Linear, Huber, RANSAC and THeilSen Regression models to predict relapse-free survival (RFS) [2]. The result shows that the Huber Regression predicted well among them with an MAE of 1.069.

3. METHODOLOGY

3.1. Experimental Design

In this experiment, several machine learning pipelines will be proposed and implemented to determine and obtain the optimum models to predict PCR (classification) and RFS (regression). The missing value will be selected and removed in each row and column at the beginning stage after the dataset is loaded. Then the dataset will be split into two variables which are input (X) and output (Y). The input values will be normalized to reduce the ranges of the input variables. Then feature selection method will be applied to obtain and select the useful column to train the machine learning models. Different combinations of feature selection and machine learning models were carried out for a competition to determine and select the best pipeline that scores the best. SelectKBest with RF, SelectKBest with XGBoost Regressor, and Backward feature elimination with KNN-Regressor and Ridge Regressor will be implemented for the regression task. For the classification task, LinearSVC with MLP, forward feature selection with SVM, forward feature selection with DT and PCA with MLP pipelines will be implemented. The feature selection method mentioned will be used to select the most useful features and fit them into the model for training purposes. After that, the trained model will be evaluated with specific evaluation metrics to determine the performance.

3.2. Data cleaning & Pre-processing

In the dataset given, there is some missing value that replace by number “999”. The missing values will affect the performance of the model when predicting the result. Hence, there are two methods to handle it which are drop the missing value or filled with K-Nearest Neighbors (KNN) imputer. The KNN use the mean value of ‘k’ number of sample that are closer in the space to fill in the missing data point [3].

In this study, StandardScaler is chosen for the normalization method. StandardScaler standardizes features by subtracting the mean and scaling to unit variance as seen in eq. (1). This method can reduce the ranges of the input datasets or simplify when they are measured in several units

of measurement. The standard deviation, σ and mean, μ are determined and the component is scaled relying on [4] :

$$Z_{scaled} = \frac{(X-\mu)}{\sigma} \quad (1)$$

3.3. Feature selection

3.3.1. SelectKBest

SelectKBest is a univariate feature selection method that works by accessing the strength of the relationship of each input data to the target based on univariate statistical tests. It is a module in scikit learn library that chooses the k features with the best scores. The SelectKBest function seeks the most relevant features by combining the F-test and p-value scores to evaluate the effectiveness of the parameters. An estimate of the amount of linear dependence between two random variables will be added by the F-test. This makes it more reliable than the wrapper technique [5].

3.3.2. Sequential features selection (SFS)

SFS is a wrapper feature selection algorithm. The principle of SFS is sequentially iterated to remove (backward selection) or add (Forward selection) the feature variable that yield the highest accuracy by an estimator. Then, SFS select the set of feature variables for which the outcomes of the quantitative analysis feature selection are the lowest number of variables and the maximum quantitative analysis accuracy [6]. In this experiment, Sequential forward elimination will be used in classification task and sequential backward elimination will be used in regression task.

3.3.3. Principal components analysis (PCA)

PCA is a dimension reduction method that translates the variable through a linear function that maximize the variance and uncorrelated with each other. PCA calculate the covariance matrix and looking the eigenvectors and eigenvalues of its to identify the principal components. Consequently, the user can decide whether to keep the component or discard the low eigenvalues vectors [7].

3.3.4. Linear SVM for features selection

The objective of SVM is to find the optimal hyperplane to classify data point into difference classes. The optimal hyperplane is call support vector will be orientated by maximising the distance between the hyperplane on either side of the class. By using the SVM coefficient the main features can be identified and remove the feature that hold the less variance [8].

3.4. Model selection

3.4.1. Random Forest (RF)

In order to classify or predict the value, Random Forest is an ensemble learning approach that combines a large number T of decision trees, resulting in a decrease in variance compared to the single decision tree.

$$f_r^K{}_f(x) = \frac{1}{K}\sum_{k=1}^{K} T(x) \quad (2)$$

Equation 2 describes when RF obtain an (x) input vector, made up of the values of the different evidential features evaluated for a given training area. K regression tress are constructed by RF, which then average the outputs. The equation (1) is the RF regression predictor formed once K such trees $\{T(x)\}_1^K$ are grown [9].

3.4.2. XGBoost regression

XGBoost is an optimized distributed gradient boosting library. It applies a recursive binary splitting approach to select the best split at each step to reach the best model.

$$L^{(t)} = \sum_i l\left(y_{\text{pred}}^{(t)}, y_{\text{truth}}\right) + \sum_k \Omega(f_k) \quad (3)$$

Equation 3 illustrates the regularized objective of the XGBoost model at the t^{th} training step, where $l\left(y^{(t)}{}_{pred}, y_{truth}\right)$ represents the loss that relate to calculating the difference between the prediction of the simulated ground value and missing value.

$\Omega(f_k) = \gamma T + \frac{1}{2}\lambda \| w \|^2$ is a regularizer used to express the complexity of the k^{th} tree. For training instances, the L2 normalization of all leaf scores are denoted by $||w||^2$, where T stands for the number of leave. The level of conservatism is controlled by γ and λ when searching the tree.

3.4.3. K_Nearest Neighbour (KNN) Regressor

The KNN regressor is a semi-supervised le arning algorithm that based on learning data of which the closest distance to the object. In the KNN algorithm, testing data, training data and a k value must be predefined. Then, to find the distance of neighbours, the calculation process must be carried out based on hamming distance formula (2) or Manhattan distance (3). The new data will be grouped based on their distance to some of the closest neighbours. A value will be produced based on the similarity function to determine whether there are similarities between the new cases and those in the case base [2]. In the similarity function (4), n is number of attributes in each case, t is new case, and s is the value of the closeness of the case. In the hamming distance (5) and Manhattan distance (6), K is the number of attributes in each case, I is new case and W is the value of the proximity of the case.

$$\text{similarity}(T,S) = \frac{\sum_{i=1}^{n} f(T_i S_i)x}{w_i}$$

$$\text{similarity}(T,S) = \frac{\sum_{i=1}^{n} f(T_i S_i)x}{w_i} \quad (4)$$

$$D_n = \frac{\sum_i^K |I_i - W_i|}{\sum_i^K |I_i + W_i|} \quad D_n = \frac{\sum_i^K |I_i - W_i|}{\sum_i^K |I_i + W_i|} \quad (5)$$

$$D_n = \sum_i^K \frac{|I_i - W_i|}{k} D_n = \sum_i^K \frac{|I_i - W_i|}{k} \quad (6)$$

3.4.4. Ridge

Since ridge regression is a model tuning method, ordinary ridge regression (ORR) based on ridge regression is used to avoid data from multicollinearity. When it occurs, the variances are large, and the least squares are unbiased. It leads to a large difference between predicted values and actual values. The method implies adding a small positive constant (K) to the main diagonal elements of the information matrix (X ′ X). The links between the explanatory variables are decoded by the positive value which is the ridge parameter [3]. The k value in the matrix equation of ORR (4) below is not bigger than 0.

3.4.5. Multilayer perceptron (MLP)

The multilayer perceptron consists of a system of simple interconnected neurons, or nodes, which is a model representing a nonlinear mapping between an input vector and an output vector.

$$O(x) = G(b(1) + W(1)h(x)) \quad (7)$$

As show in eq 7, This method consists of weighting W (1) the input h(X) of the nodes with adding a bias vector b (1) and channels them through an activation function G. During the model training the weight and bias will keep iterate to minimise the loss function by using gradient descent.

3.4.6. Decision Tree.

A decision tree is a type of supervised machine learning that is used to categorise or make predictions depending on the answers to a prior series of questions. In a decision tree, predicting a class label for a record will begin at the tree's root. The values of the root property are then compared to the values of the record's attribute

3.5. Evaluation result

3.5.1. Datasets

In this experiment, a dummy public dataset is provided from The American College of Radiology Imaging Network. There are 400 overall rows and 120 columns in the dataset. Each row represents a patient in the dataset containing 10 clinical features and 107 imaged-based features.

3.5.2. Evaluation metrics

The method's performance was evaluated using k-fold cross-validation and evaluation metrics for regression and classification tasks. Mean Absolute Error (MAE) and Root Mean Square Error (RMSE) will be calculated for regression models for comparison. F1 score (F1) and classification accuracy will be calculated with k-fold cross-validation to determine the performance of classification models for comparison.

4. PARAMETER SETTINGS

Table 1: Classification Hyperparameter Tuning

Model	Hyper-parameter	Type	Search Space
Decision Tree Classifier	max_depth	Discrete	[None,1,3,5,7,9, 11,12]
	min_ samples_ leaf	Discrete	[1,2,3,4,5,6,7,8,9, 10]
	max_ features	Categorical	[None,"auto", "log2","sqrt"]
	max_leaf_ nodes	Discrete	[None,10,20,30,40,5 0,60,70,80,90]
	min_ samples_ split	Discrete	[2, 3, 4]
Logistic Regression	C	Continuous	[0.001, 0.01, 0.1, 1, 10, 100, 1000]
SVC	C	Continuous	[0.1, 1, 10, 100, 1000]
	kernel	Categorical	['rbf','poly', 'linear','sigmoid']
MLP	hidden_ layer_sizes	Discrete	[(50,50,50), (50,100,50), (100,)]
	activation	Categorical	['tanh', 'relu']
	solver	Categorical	['sgd', 'adam']
	alpha	Continuous	[0.0001, 0.05]
	learning_ rate	Categorical	['constant', 'adaptive']

Table 2: Regression hyperparameter Tuning

Model	Hyper-parameter	Type	Search Space
KNN Regressor	n_neighbors	Discrete	[1,2,3,4,5,6,7,8,9,10]
Ridge Regressor	alpha	Continuous	[0.1,1,10,100]
XGBoost Regressor	nthread	Discrete	[4]
	objective	Categorical	['reg:linear']
	learning_ rate	Continuous	[.01,0.05,.1]
	max_depth	Discrete	[1,5,6,7]
	min_child_we ight	Discrete	[4]
	subsample	Continuous	[0.7]
	colsample_ bytree	Continuous	[0.3,0.5,0.8]
	gamma	Discrete	[0,1,5]
	n_ estimators	Discrete	[500]
Random Forest Regressor	boostrap	Boolean	[True]
	max_depth	Discrete	[80,90,100,110]
	max_ features	Discrete	[2,3]
	min_sample_l eaf	Discrete	[3,4,5]
	min_ samples_ split	Discrete	[8,10,12]
	n_ estimators	Discrete	[100,200,300,1000]

The hyperparameters of each model above are tuned with the search space to obtain the best models. The grid search method is applied with 5-fold cross-validation to try every combination of values in each hyperparameter to find the best-performing combination in regression and classification tasks. To apply the grid search method, “GridSearchCV” is

used as a splitter to iteratively splits testing and training dataset as an array of indices to obtain the best models.

5. RESULTS

Table 3: Classification Results

Model	Feature Selection Method	Feature Selected	Classification Accuracy	F1-Score
SVM	**SFS**	10	0.75	0.25
DT	**SFS**	10	0.68	0.27
MLP	**LinearSVC**	7	0.77	0.28
MLP	**PCA**	10	0.72	0.14

Table 4: Regression Results

Model	Feature Selection Method	Feature Selected	MAE	RMSE
RF	SelectKBest	24	20.09	25.45
XGBoost	SelectKBest	24	20.22	25.31
KNN	SBS	26	19.31	25.39
Ridge	SBS	33	19.38	25.13

Tables 3 and 4 show the result obtained from the implementation of different combinations. The model, feature selection method, feature selected, and the result obtained from evaluation metrics are shown. The evaluation result is obtained with 5-fold of cross validation method by automatically split the dataset into 4 train sets with 1 test sets to evaluate the performance of the models.

In table 3, MLP with linearSVC feature selection method obtain the best classification accuracy score and F1-score with the least selected feature when training the model. In table 4, KNN with SBS feature selection method outstanding other models. It has the least MAE score. Although the feature selected in the KNN model and RMSE is not the smallest, the MAE score will be focused on in this experiment.

6. DISCUSSION

For Regression, KNN regressor has the best result. Since KNN regressor is typically implemented using an approximative closest neighbor search technique like KD-tree, it is affected by the curse of dimensionality. However, SBS decrease the dimensionality of dataset. Hence, KNN regressor can handle the data easily because KNN work well with low dimensionality and small dataset. Furthermore, unlike XGBoost, KNN regressor is less sensitive to outlier if the 'k_neighbors' is large enough. Since random forest uses many decisions tree, it can require a lot of data to train its model, but the dataset is not enough to feed the model. Ridge can handle the data when there is multicollinearity in data. Because the multiple independent variables in dataset have strong correlation with other independent variables apart from correlation with dependent variables, the result of ridge regressor close to the best model, KNN but lower than it.

For classification, the decision tree has the worst result because the dataset contains multiple classes and a limited number of training samples, decision trees are prone to errors. Smaller datasets with higher dimensions produce better results for SVM because it takes longer to process. However, the result of SVM is not higher than MLP. It is because the size of the hidden layer is large enough to fit the dataset to train the model. Besides, MLP is also suitable to apply to complex non-linear problems. Although MLP provides quick prediction after training, the processing time is difficult and time-consuming compared to the other models. In addition, the feature selection LinearSVC that is applied with the MLP model works effectively when comes to small datasets. It is also effective in high-dimensional spaces. However, this method is not suitable when the dataset is large and contains noise.

7. CONCLUSION

The primary goal of this research project is to develop the best predictive machine learning models for PCR and RFS prediction with high accuracy. Results analysis reveals that the integration of multidimensional data with various regression or classification, dimensionality reduction and feature selection have various characteristics in prediction. Further research in this field should continue to obtain better performance in both PCR and RFS prediction.

8. REFERENCE

[1] R. Bhardwaj and N. Hooda, "Prediction of Pathological Complete Response after Neoadjuvant Chemotherapy for breast cancer using ensemble machine learning," *Informatics in Medicine Unlocked,* vol. 16, p. 100219, 2019/01/01/ 2019, doi: https://doi.org/10.1016/j.imu.2019.100219.

[2] S. Karim *et al.*, "Gene expression study of breast cancer using Welch Satterthwaite t-test, Kaplan-Meier estimator plot and Huber loss robust regression model," *Journal of King Saud University - Science,* vol. 35, no. 1, p. 102447, 2023/01/01/ 2023, doi: https://doi.org/10.1016/j.jksus.2022.102447.

[3] T. Mahboob, A. Ijaz, A. Shahzad, and M. Kalsoom, "Handling Missing Values in Chronic Kidney Disease Datasets Using KNN, K-Means and K-Medoids Algorithms," in *2018 12th International Conference on Open Source Systems and Technologies (ICOSST)*, 19-21 Dec. 2018 2018, pp. 76-81, doi: 10.1109/ICOSST.2018.8632179.

[4] V. N. G. Raju, K. P. Lakshmi, V. M. Jain, A. Kalidindi, and V. Padma, "Study the Influence of Normalization/Transformation process on the Accuracy of Supervised Classification," in *2020 Third International Conference on Smart Systems and Inventive Technology (ICSSIT)*, 20-22 Aug. 2020 2020, pp. 729-735, doi: 10.1109/ICSSIT48917.2020.9214160.

[5] D. A. Otchere, T. O. A. Ganat, J. O. Ojero, B. N. Tackie-Otoo, and M. Y. Taki, "Application of gradient boosting regression model for the evaluation of feature selection techniques in improving reservoir characterisation predictions," *Journal of Petroleum Science and Engineering,* vol. 208, p. 109244, 2022/01/01/ 2022, doi: https://doi.org/10.1016/j.petrol.2021.109244.

[6] M. Li *et al.*, "In situ simultaneous quantitative analysis multi-elements of archaeological ceramics via laser-induced breakdown spectroscopy combined with machine learning strategy," *Microchemical Journal,* vol. 182, p. 107928, 2022/11/01/ 2022, doi: https://doi.org/10.1016/j.microc.2022.107928.

[7] I. T. Jolliffe and J. Cadima, "Principal component analysis: a review and recent developments," *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences,* vol. 374, no. 2065, p. 20150202, 2016, doi: doi:10.1098/rsta.2015.0202.

[8] J. Neumann, C. Schnörr, and G. Steidl, "Combined SVM-Based Feature Selection and Classification," *Machine Learning,* vol. 61, no. 1, pp. 129-150, 2005/11/01 2005, doi: 10.1007/s10994-005-1505-9.

[9] V. Rodriguez-Galiano, M. Sanchez-Castillo, M. Chica-Olmo, and M. Chica-Rivas, "Machine learning predictive models for mineral prospectivity: An evaluation of neural networks, random forest, regression trees and support vector machines," *Ore Geology Reviews,* vol. 71, pp. 804-818, 2015/12/01/ 2015, doi: https://doi.org/10.1016/j.oregeorev.2015.01.001.

Task and Weighting	Data pre-processing (10%)	Feature Selection (20%)	ML method development (30%)	Method Evaluation (10%)	Report Writing (30%)
Chang Neng Chek (20200912)	20	20	20	20	13
Peng Yang How (20112506)	20	20	20	20	20
Ser Vin Chan (20112392)	20	20	20	20	17
Alden Sia Zheng Heng (20196637)	20	20	20	20	25
Wei Sheng Ooi (20204891)	20	20	20	20	25