Health Informatics and Biomedical Engineering Applications, Vol. 78, 2023, 49–56
https://doi.org/10.54941/ahfe1003454

PBC-ML: Predicting Breast Cancer in Humans Using Machine Learning Approach

Oni Damilola Igbagbo, Satyam Mishra, Vu Minh Phuc, Le Trung Thanh, and Nguyen Ngoc Linh

International School, Vietnam National University, Hanoi (VNU-IS), Vietnam

ABSTRACT

Cancer is a disease in which cells grow uncontrollably, potentially causing harm to surrounding healthy tissue and organs. Breast cancer is a specific type of cancer that affects the breast and is the second most common cancer among women worldwide. Symptoms of breast cancer include a lump or tumour, swelling, nipple discharge, and swollen lymph nodes. Breast cancer is staged, with stage 0 being the earliest stage with minimal symptoms and stage 4 indicating the cancer has spread to other parts of the body. The future burden of breast cancer is predicted to increase, with over 3 million new cases and 1 million deaths in 2040. Early detection is crucial for successful treatment and recovery, and machine learning can be used to predict the likelihood of breast cancer based on symptoms. So we propose in our research to use machine learning algorithms such as CART, SVM, NB, and KNN to analyse and build models for breast cancer detection. These findings offer a summary of relevant machine learning methods for breast cancer detection as it will help to curb it and we got an accuracy of 98.2% compared to the state of art methods which has accuracy of 99%. It proves to be a valuable tool in the early detection of breast cancer and can improve the accuracy of existing diagnostic methods.

Keywords: Cancer, Breast cancer, Cart, Svm, Knn, Machine learning

INTRODUCTION

Cancer is a disorder in which the body's cells grow unrestrained, it's a disease in which some of the body's cells grow uncontrollably and spread to other parts of the body. (*Cancer - Symptoms and Causes*, n.d.) Cancer can start almost anywhere in the human body, which is made up of trillions of cells. Normally, human cells grow and multiply (through a process called cell division) to form new cells as the body needs them. When cells grow old or become damaged, they die, and new cells take their place. Although this orderly process breaks down, and abnormal cells grow and multiply when they shouldn't. These cells may form tumours, which are lumps of tissue. Tumours can be cancerous or benign. Cancerous tumours invade nearby tissues and usually move to other part of the body to form new tumours in a process called metastasis. Cancerous tumours can also be called malignant tumours. There are some works being done for image processing as well using Canny Edge Detection Algorithm (Mishra & Thanh, 2022).

Breast cancer is a disease in which cells in the breast grow out of control. It might begin in either one or both breasts. After skin cancer, breast cancer is the most common cancer diagnosed in women in the world and it can occur in both men and women, but it's far more common in women. (CDCBreastCancer, 2022) There are different kinds of breast cancer, though the most common types are Invasive ductal carcinoma and Invasive lobular carcinoma. Breast cancer can begin in different parts of the breast. The breast is made up of lobules, ducts, and connective tissue which are the major parts. The lobules produces milk, while ducts are tubes that carry milk to the nipple and the connective tissue consists of fibrous and fatty tissue that surrounds and holds everything together. Most breast cancers begin in the ducts or lobules and it can spread outside the breast through blood vessels and lymph vessels to other parts of the body.

Breast cancer is the most commonly diagnosed cancer worldwide, and its burden has been rising over the past years, breast cancer today accounts for 1 in 8 cancer diagnoses and a total of 2.3 million new cases in both sexes combined. Representing a quarter of all cancer cases in females, it was by far the most commonly diagnosed cancer in women in 2020, and its burden has been growing in many parts of the world, particularly in transitioning countries. An estimated 685,000 women died from breast cancer in 2020, corresponding to 16% or 1 in every 6 cancer deaths in women. While the future burden of breast cancer is predicted to increase to over 3 million new cases and 1 million deaths in 2040. In order to curb prevent all these predicted cases early detection must be prioritized therefore machine learning can be employed to predict the cases of breast cancer correctly and early diagnosis is an extremely important step in rehabilitation and treatment (Cruz-Ramírez et al., 2007). Also, some neural network approach training for object detection etc. for driverless vehicles to take pandemic affected patients (Mishra et al., 2022). In our research, we propose to build and analyse machine learning model to predict if a given set of symptoms lead to breast cancer. This is a binary classification problem, and a few algorithms are appropriate for use. Since we do not know which one will perform the best at the point, we will do a quick test on the few appropriate algorithms with default setting to get an early indication of how each of them perform. The following nonlinear algorithms will be used, namely: Classification and Regression Trees (CART), Linear Support Vector Machines (SVM), Gaussian Naive Bayes (NB) and k-Nearest Neighbors (KNN).

METHODOLOGY

In this research we employ the following process in the methodology and the process was visualize using a pictorial representative in figure *1*.

Data pre-processing is the manipulation of data before it is used in order to enhance performance, it has four sub steps which are data cleaning, data integration, data reduction, and data discretization, after data pre-processing we moved to model selection which is the process where we select the best model from a set of candidate models for the dataset although it can be applied both across different types of models and across models of the same type

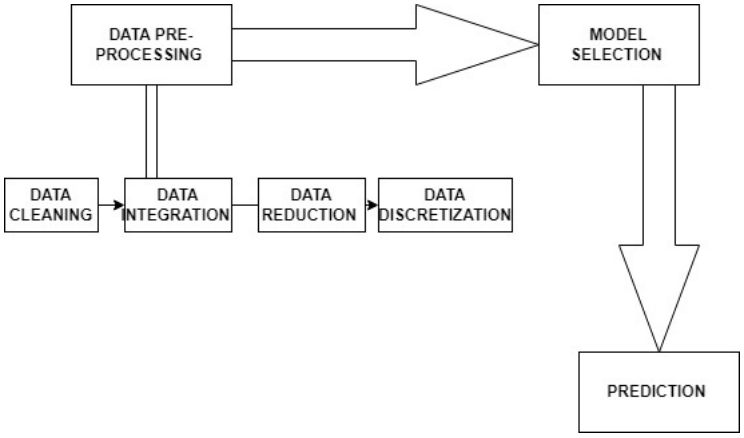

Figure 1: Pictorial representative of the methodology.

configured with different model. For this research we used three different models non-linear algorithms namely: Classification and Regression Trees (CART), Linear Support Vector Machines (SVM), Gaussian Naive Bayes (NB) and k-Nearest Neighbors (KNN). After applying the algorithms we will have the prediction which is the output of the chosen algorithm after it has been trained on the particular dataset and applied to new data when forecasting the likelihood of a particular outcome.

ALGORITHMS USED: CLASSIFICATION AND REGRESSION TREES (CART)

The Classification and Regression Tree methodology, also known as the CART were introduced in 1984 by Leo Breiman, Jerome Friedman, Richard Olshen, and Charles Stone. Classification and regression trees (CART) is a supervised machine learning technique used for both classification and regression tasks. It is a decision tree-based algorithm that uses a tree-like structure to represent the relationships between the input features and the target variables (Venkatesan & Velmurugan, 2015).

HOW CART ALGORITHM WORKS

- The best split point of each input is obtained.
- Based on the best split points of each input in Step 1, the new "best" split point is identified.
- Split the chosen input according to the "best" split point.
- Continue splitting until a stopping rule is satisfied or no further desirable splitting is available.

LINEAR SUPPORT VECTOR MACHINES

Linear Support Vector Machines (SVMs) is a supervised machine learning algorithm used for both classification and regression tasks. It was invented by Vladimir Vapnik and Alexey Chervonenkis in the 1960s. They developed the

algorithm as a way to solve the problem of pattern recognition in machine learning. The algorithm was later refined and improved by other researchers, and it is now widely used in many applications. (*Performance Analysis of Support Vector Machines Classifiers in Breast Cancer Mammography Recognition | SpringerLink*, n.d.)

HOW LINEAR SVM ALGORITHM WORKS

Linear Support Vector Machines (SVMs) is a supervised machine learning algorithm used for both classification and regression tasks. It is a linear model that uses a hyperplane to separate the data into two classes. The model is trained using a cost function that penalizes misclassification, and the optimal hyperplane is determined by maximizing the margin between the two classes.

GAUSSIAN NAIVE BAYES (NB)

Gaussian Naive Bayes is a supervised machine learning algorithm used for classification tasks. It is a probabilistic classifier that uses Bayes' theorem to make predictions based on the probability of a given class given the input features. It assumes that the input features are independent of each other, which simplifies the calculations. This can handle both continuous and discrete data (Kamel et al., 2019).

HOW GAUSSIAN NAIVE BAYES (NB) WORKS

Naïve Bayes' Theorem usually predict the probability of an outcome, given some input features. It is based on a strong assumption of independence between the input features

- Naive Bayes is a generative model.
- (Gaussian) Naive Bayes assumes that each class follow a Gaussian distribution.
- Naive Bayes assumes independence of the features, which means the covariance matrices are diagonal matrices.

K-NEAREST NEIGHBORS (KNN)

K-Nearest Neighbors (KNN) was invented in the late 1950s by the American computer scientist and statistician Arthur Samuel. It is a supervised learning algorithm used for classification and regression. It works by identifying the closest K-Neighbors of a given data point, and uses them to determine the class or value of that data point. KNN can work with both discrete and continuous data (Odajima & Pawlovsky, 2014).

HOW K-NEAREST NEIGHBORS (KNN) WORKS

K-Nearest Neighbors (KNN) works by calculating the distance between a given data point and its closest k-Neighbors. It then uses this information to determine the class or value of the data point by taking a majority vote amongst its closest Neighbors. KNN is an example of a supervised learning algorithm, meaning that it uses prior knowledge of labelled data points to make predictions about unclassified data points.

APPLICATION & RESULTS

Our updated dataset was gotten from Kaggle, and it was implemented on 'colab.research.google.com' we examine the dataset and create a model to determine whether a specific group of symptoms is indicative of breast cancer. There aren't many methods that are suitable for use in this binary classification problem. We quickly test a few suitable algorithms with default settings because we are unsure which one will work best at this time in order to gain a preliminary idea of how each of them perform. For each test, we do cross validation that is 10 fold.

From the first run in *figure 2* Gaussian NB, KNN and CART performed (all above 92% mean accuracy) while Support Vector Machine has a bad performance. However when we standardize the input dataset the performance improve.

The performance of the few machine learning algorithm was improved after a standardized dataset has being used as seen in *figure 3*. The improvement was observed in all the models. That way we get a fair estimation of how each model with standardized data might perform on unseen data.

After the above stage we did Algorithm Tuning for linear support vector machines with aim of tuning the model to ensure that it performs at its best.

```
[ ] num_folds = 10
    results = []
    names = []
    for name, model in models_list:
        kfold = KFold(n_splits=num_folds)
        start = time.time()
        cv_results = cross_val_score(model, X_train, Y_train, cv=kfold, scoring='accuracy')
        end = time.time()
        results.append(cv_results)
        names.append(name)
        print( "%s: %f (%f) (run time: %f)" % (name, cv_results.mean(), cv_results.std(), end-start))

    CART: 0.927488 (0.021920) (run time: 0.097241)
    SVM: 0.906039 (0.082881) (run time: 0.074807)
    NB: 0.938744 (0.038473) (run time: 0.029184)
    KNN: 0.940918 (0.042447) (run time: 0.061981)
```

Figure 2: Cross validation result.

```
[ ] # Standardize the dataset
    pipelines = []

    pipelines.append(('ScaledCART', Pipeline([('Scaler', StandardScaler()),('CART',
                                                    DecisionTreeClassifier())])))
    pipelines.append(('ScaledSVM', Pipeline([('Scaler', StandardScaler()),('SVM', SVC( ))])))
    pipelines.append(('ScaledNB', Pipeline([('Scaler', StandardScaler()),('NB',
                                                   GaussianNB())])))
    pipelines.append(('ScaledKNN', Pipeline([('Scaler', StandardScaler()),('KNN',KNeighborsClassifier())])))
    results = []
    names = []
    with warnings.catch_warnings():
        warnings.simplefilter("ignore")
        kfold = KFold(n_splits=num_folds)
        for name, model in pipelines:
            start = time.time()
            cv_results = cross_val_score(model, X_train, Y_train, cv=kfold, scoring='accuracy')
            end = time.time()
            results.append(cv_results)
            names.append(name)
            print( "%s: %f (%f) (run time: %f)" % (name, cv_results.mean(), cv_results.std(), end-start))

    ScaledCART: 0.931981 (0.024560) (run time: 0.110841)
    ScaledSVM: 0.971449 (0.022053) (run time: 0.072004)
    ScaledNB: 0.934300 (0.037659) (run time: 0.034247)
    ScaledKNN: 0.964976 (0.026049) (run time: 0.084428)
```

Figure 3: Result of standardized dataset.

This process involves adjusting various elements of the model to achieve optimal results. By fine-tuning the model, we maximize its performance and get the highest rate of performance possible. We tune two key parameter of the SVM algorithm - the value of C and the type of kernel. The default C for SVM is 1.0 and the kernel is Radial Basis Function (RBF). We used the grid search method using 10-fold cross-validation with a standardized copy of the sample training dataset and over a combination of C values and the following kernel types 'linear', 'poly', 'rbf' and 'sigmoid.

We later apply the SVC on another dataset. We fit the SVM into the dataset and observe how it performs on the given dataset. It brings out the accuracy of 98.2% as seen in *figure 5*.

```
[ ] scaler = StandardScaler().fit(X_train)
    rescaledX = scaler.transform(X_train)
    c_values = [0.1, 0.3, 0.5, 0.7, 0.9, 1.0, 1.3, 1.5, 1.7, 2.0]
    kernel_values = ['linear', 'poly', 'rbf', 'sigmoid']
    param_grid = dict(C=c_values, kernel=kernel_values)
    model = SVC()
    kfold = KFold(n_splits=num_folds)
    grid = GridSearchCV(estimator=model, param_grid=param_grid, scoring='accuracy', cv=kfold)
    grid_result = grid.fit(rescaledX, Y_train)
    print("Best: %f using %s" % (grid_result.best_score_, grid_result.best_params_))
    means = grid_result.cv_results_['mean_test_score']
    stds = grid_result.cv_results_['std_test_score']
    params = grid_result.cv_results_['params']
    for mean, stdev, param in zip(means, stds, params):
        print("%f (%f) with: %r" % (mean, stdev, param))

Best: 0.971498 using {'C': 0.1, 'kernel': 'linear'}
0.971498 (0.019810) with: {'C': 0.1, 'kernel': 'linear'}
0.820676 (0.101618) with: {'C': 0.1, 'kernel': 'poly'}
0.954106 (0.035613) with: {'C': 0.1, 'kernel': 'rbf'}
0.956377 (0.037570) with: {'C': 0.1, 'kernel': 'sigmoid'}
0.969275 (0.020118) with: {'C': 0.3, 'kernel': 'linear'}
0.857778 (0.076128) with: {'C': 0.3, 'kernel': 'poly'}
0.964928 (0.024403) with: {'C': 0.3, 'kernel': 'rbf'}
0.967246 (0.024241) with: {'C': 0.3, 'kernel': 'sigmoid'}
0.967053 (0.017801) with: {'C': 0.5, 'kernel': 'linear'}
0.875314 (0.071021) with: {'C': 0.5, 'kernel': 'poly'}
```

Figure 4: Results of algorithm tuning. The best: 0.971498 using {'C': 0.1, 'kernel': 'linear'}.

```
# prepare the model
with warnings.catch_warnings():
    warnings.simplefilter("ignore")
    scaler = StandardScaler().fit(X_train)
X_train_scaled = scaler.transform(X_train)
model = SVC(C=2.0, kernel='rbf')
start = time.time()
model.fit(X_train_scaled, Y_train)
end = time.time()
print( "Run Time: %f" % (end-start))

Run Time: 0.007863
```

```
[ ] # estimate accuracy on test dataset
    with warnings.catch_warnings():
        warnings.simplefilter("ignore")
        X_test_scaled = scaler.transform(X_test)
    predictions = model.predict(X_test_scaled)
```

```
[ ] print("Accuracy score %f" % accuracy_score(Y_test, predictions))
    print(classification_report(Y_test, predictions))

    Accuracy score 0.982456
```

Figure 5: Accuracy score.

```
[ ] Accuracy score 0.982456
                  precision    recall  f1-score   support

               0       1.00      0.98      0.99        88
               1       0.93      1.00      0.96        26

        accuracy                           0.98       114
       macro avg       0.96      0.99      0.98       114
    weighted avg       0.98      0.98      0.98       114


[ ] print(confusion_matrix(Y_test, predictions))

    [[86  2]
     [ 0 26]]
```

Figure 6: Accuracy results.

We also evaluated the performance of the model through confusion matrix, the calculation was based on performance metrics like accuracy, precision, recall, and F1-score. Confusion matrices are widely used because they give a better idea of a model's performance than classification accuracy does. With an accuracy of 98.2% on the held-out test dataset and from the confusion matrix we got 2 cases of misclassification. The performance of this algorithm is expected to be high given the symptoms for breast cancer should exhibits certain clear patterns.

CONCLUSION

Our Findings	State of the Art (SOTA)
Accuracy of 98.2%	Accuracy of 99%
We used the following algorithm: K-nearest, linear support vector machine, Gaussian Naïve Bayes, Classification & Regression Tress	Algorithms in SOTA; Decision tree, AD-Tree, Multi-layer Neural network, Bayesian Network
The best algorithm is Linear support vector machine.	The best algorithm is Bayesian network.

The results of this research will be beneficial in the early diagnosis of breast cancer as diagnosing breast cancer earlier has many benefits, including increasing the chances of successful treatment and reducing the risks associated with late-stage diagnosis. Early diagnosis also reduces the financial burden of treating advanced-stage breast cancers, as it is often more expensive to treat them. Additionally, early diagnosis can improve quality of life for those affected, as they can start the treatment sooner and avoid the physical and emotional distress caused by a late diagnosis and we believe it will help doctors in making better decisions and prompt medical interventions.

ACKNOWLEDGMENT

Foremost, we would like to express our sincere gratitude to our advisor Dr. Nguyen Van Tanh for the continuous support for our group's study and research, for his patience, motivation, enthusiasm, and immense knowledge. His guidance helped us in all the time of research and writing of this research. We could not have imagined having a better advisor and mentor for our work.

REFERENCES

Cancer—Symptoms and causes. (n.d.). Mayo Clinic. Retrieved February 2, 2023, from https://www.mayoclinic.org/diseases-conditions/cancer/symptoms-causes/syc-20370588

CDCBreastCancer. (2022, March 9). *What Is Breast Cancer?* Centers for Disease Control and Prevention. https://www.cdc.gov/cancer/breast/basic_info/what-is-breast-cancer.htm

Cruz-Ramírez, N., Acosta-Mesa, H. G., Carrillo-Calvet, H., Alonso Nava-Fernández, L., & Barrientos-Martínez, R. E. (2007). Diagnosis of breast cancer using Bayesian networks: A case study. *Computers in Biology and Medicine*, *37*(11), 1553–1564. https://doi.org/10.1016/j.compbiomed.2007.02.003

Kamel, H., Abdulah, D., & Al-Tuwaijari, J. M. (2019). Cancer Classification Using Gaussian Naive Bayes Algorithm. *2019 International Engineering Conference (IEC)*, 165–170. https://doi.org/10.1109/IEC47844.2019.8950650

Mishra, S., Minh, C. S., Thi Chuc, H., Long, T. V., & Nguyen, T. T. (2022). Automated Robot (Car) using Artificial Intelligence. *2021 International Seminar on Machine Learning, Optimization, and Data Science (ISMODE)*, 319–324. https://doi.org/10.1109/ISMODE53584.2022.9743130

Mishra, S., & Thanh, L. T. (2022). SATMeas - Object Detection and Measurement: Canny Edge Detection Algorithm. In X. Pan, T. Jin, & L.-J. Zhang (Eds.), *Artificial Intelligence and Mobile Services – AIMS 2022* (pp. 91–101). Springer International Publishing. https://doi.org/10.1007/978-3-031-23504-7_7

Odajima, K., & Pawlovsky, A. P. (2014). A detailed description of the use of the kNN method for breast cancer diagnosis. *2014 7th International Conference on Biomedical Engineering and Informatics*, 688–692. https://doi.org/10.1109/BMEI.2014.7002861

Performance analysis of support vector machines classifiers in breast cancer mammography recognition | SpringerLink. (n.d.). Retrieved February 2, 2023, from https://link.springer.com/article/10.1007/s00521-012-1324-4

Venkatesan, E. v, & Velmurugan, T. (2015). Performance analysis of decision tree algorithms for breast cancer classification. *Indian Journal of Science and Technology*, *8*(29), 1–8.