Mammogram Mass Prediction

Estimate the best algorithm for breast cancer detection

Mingming Zhang

Software Engineering

McMaster University

Hamilton, Canada

1280 Main St W

Hua Yao

Software Engineering

McMaster University

Hamilton, Canada

1280 Main St W

***Abstract***—**There are many machine learning algorithms available to apply in all kinds of study fields nowadays. Moreover, new algorithms keep popping up based on existing models. It gives us unprecedented power to forecast the future. However, we are not only looking for the tendency or an approximate result on the unknown data sometimes. We insist on maximizing prediction correctness in some critical fields. This report compares the different machine learning algorithms to find the best prediction model for breast cancer detection. It explains the methodology of the experimental approach and discusses the different methods in each step based on the mammography mass dataset.**

***Keywords****—cross-validation; ROC; decision tree; random forest; SVM; KNN; HNSW; Naïve Bayes; logistic regression; artificial neural networks*

1. Introduction

This report focuses on predicting breast cancer by utilizing different AI algorithms to analyze the mammography to identify whether benign or malignant mammogram mass. Breast cancer is the most common cancer in women globally, accounting for 25 percent.[[1]](#footnote-1) Many areas utilize the AI algorithm to determine the biology's status and health to get excellent results.[[2]](#footnote-2) For instance, a sensor uses AI algorithms to monitor the dairy and discern the critical dairy metrics such as mastitis, somatic cell count, fat and other milk-quality elements. It supports better decisions for the farmer to keep healthier herds and supply higher-quality milk for people.[[3]](#footnote-3) Mammography is an effective way to detect breast cancer, but there is some error in distinguishing benign from malignant. However, the high efficiency and accuracy in judging whether women have it is still the primary issue that disturbs the doctors and delays the patient's treatment. This project trains benign and malignant mammogram mass identification based on current breast X-ray results. Many AI algorithms can implement into prediction. In order to choose the compatible algorithm to deal with this kind of subject, the cross-validation and area under the receiver operating curve are better strategies to evaluate the rank of those algorithms.

1. STRATEGY

Generally speaking, the metrics of evaluating the models measure the model's performance. That means how well the model learns to discriminate among model results. For supervised learning, good models will have a strong generalization ability for unknown data processing. In this project, we will use different metrics to avoid bias in the evaluation.

* 1. *K-Fold Cross-validation*

We will obtain an average accuracy by using the K-Fold cross-validation method. The test set is divided into K subsets (K depends on the amount of data or personal experience). One of the k subsets is used for validation, and the k-1 subset will be used for training. Training repeat for k times, and the results are averaged.[[4]](#footnote-4) This method can maximize the amount of training data, which will minimize the risk of losing important patterns in the data set.

* 1. *Area under* *Receiver Operating Characteristic Curve*

An intuitive way to see the model performance is the ROC- AUC curve. It plots the fraction of true positives out of the positives (TPR) vs. the fraction of false positives out of the negatives (FPR), giving continuous threshold values. Because the accuracy is known as the accuracy paradox, especially in the cases of cancer prediction or fraud detection, its metric basically has rarely reference value. The ROC is suitable for evaluating the overall performance of the classifier. One of the reasons is that the distribution of the positive and negative samples in the test set changes. The ROC curve remains the same, which means the ROC and AUC can eliminate the effect of sample category imbalance on the indicator of results. Another reason is that ROC is an evaluation metric that does not depend on thresholds, and the ROC can produce an effective result compared to the required accuracy.

*Confusion Matrix*

A confusion matrix compares the actual target values with predicted target values from the model. It gives us a holistic view of the model classification result and precisely what type of errors the model is making.

Table 1 Confusion Matrix

|  |  |  |
| --- | --- | --- |
|  | Benign 0 | Malignant 1 |
| Prediction 0 | True Negative | False Negative (Type II) |
| Prediction 1 | False Positive (Type I) | True Positive |

1. Procedures

Different machine learning models will be implemented to yield the predictions. From the best solution perspective, we will choose the model with the highest score together with the highest AUC values as the solution to this problem. The cross-validation that we will apply to the models such as:

* Decision Tree
* Random Forest
* KNN
* HNSW
* Naive Bayes
* SVM
* Logistic Regression
* Neural Network
  1. *Data Requirements*

The dataset is retrieved from UCI Machine Learning Repository[[5]](#footnote-5) named "mammographic\_masses.data.txt" It provides 961 masses data detected in mammography with six numerical attributes: BI-RADS, AGE, SHAPE, MARGIN, DENSITY, and SEVERITY. The SEVERITY attribute is the target to predict, which means other attributes are the input attributes for the algorithm models. In the dataset, there are 516 benign cases and 445 malignant cases. The detailed structure of the data are as follow:



Table

Description automatically generated

Figure 1 Check the data

From the data integrity check, four columns are found with missing values: age, shape, margin and density.

Table

Description automatically generated

Figure 2 Check the invalid value

To deal with the missing value problem, first, we will fill a categorical mean into the missing values using fillna() method and train the model. This gives us more data for training. Secondly, We will also train the models using the data dropping the records with missing values. This data keeps the better integrity and accuracy of the original data. We will experiment with both methods and compare the results.

Initially, the features in the data have different scales. This could affect the correlation between features and the accuracy of the prediction. Therefore, to make each part equally important during the training process, we need to normalize them.

Text

Description automatically generated

Figure 3 Normalization of the valid value

Before implementing machine learning models, the data should be split to train set and test set. Then, we will obtain the model evaluation from the metrics of confusion metrics and ROC.



* 1. *Prerequisites*
* Python Interpreter
* Pandas
* numpy
* scikit-learn
* Tensorflow
* Keras
* matplotlib
* seaborn
  1. *Models*
* **Decision Trees**

**ID3** (**Iterative Dichotomiser 3**) algorithm generates the decision tree learning. It starts from a root node and keeps splitting the branch based on different node conditions. At the end of the branches, we obtain the leaf nodes as our output. First, the algorithm will test the best splitting point based on a cost function calculation, giving us the most homogeneous(similar values ) sub-branches. This process is called Greedy Splitting.

Here, prediction is the mean of the response of the input data of a specific group. Pk is the proportion of the same class inputs present in a particular group[[6]](#footnote-6).

More profoundly, the classification model calculates the Information Gain and the regression model calculates the Standard Deviation Reduction to construct the decision nodes for further splitting the branches.

Conditional entropy H(X|Y) of X given Y:

Information Gain( Mutual Information) of X and Y:

Standard Deviation Reduction:

*,* where T is the target and X is the specific feature group of the target[[7]](#footnote-8).

**Stopping Splitting**

The typical approach is to set a minimum number of input data points on every leaf. This method helps the model to be simple and not overfit the training data.

**Pruning branch**

Removing the branches constructed on low important features reduces the overfitting and increases the model predictive power. The fastest and most straightforward way is to evaluate the cost functions on the entire test set by testing the removal of each leaf node. Another complex way to do it is using a learning parameter(alpha) to decide whether a node can be removed based on the size of the sub-branches.

* **Random Forest**

A Random Forest Tree consists of many random trees with two types of randomness. First, each tree is built on a random sample of original data. Second, each tree node uses a randomly selected feature to generate the best split.

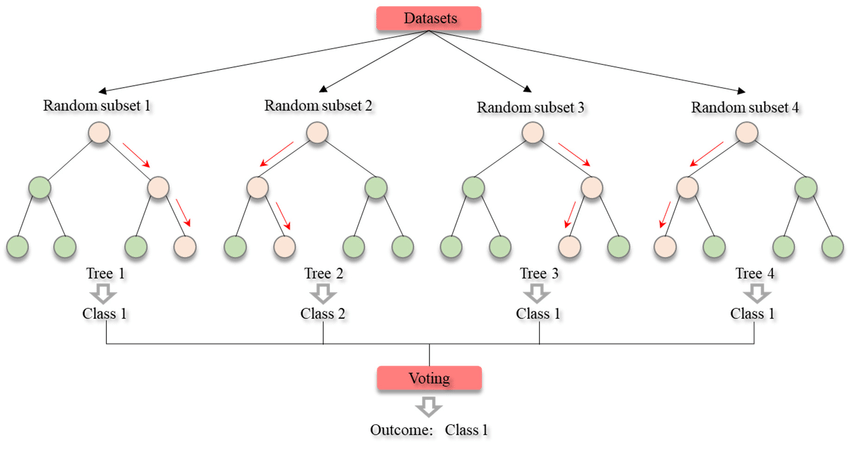


Figure 4 Random Forest Structure

In a classification problem, each tree predicts the classification of a test record. The final classification of the test record would be the classification wins the majority vote from the entire tree sets. In a case of a regression problem, each decision trees in the forest predict an output value. The final value is obtained by averaging the output from all the individual trees.[[8]](#footnote-9)

**Ensemble Learning**

Using individual models to induce the final prediction is called ensemble learning.

There are two types of ensemble methods. One is called bagging, which means training individual models in a parallel wayusing randomly selected data. Random Forest is this type. Another type is called boosting. This type trains the individual models in a sequential way. The next module learns the error made by the previous model.

* **HNSW**

HNSW stands for Hierarchical Navigable Small World. It is a reasonably new algorithm with similar results as KNN(k-nearest neighbour) with 380 times faster speed. This model is built on the KNN graphs but organizes the links between vertices into a hierarch.[[9]](#footnote-10)

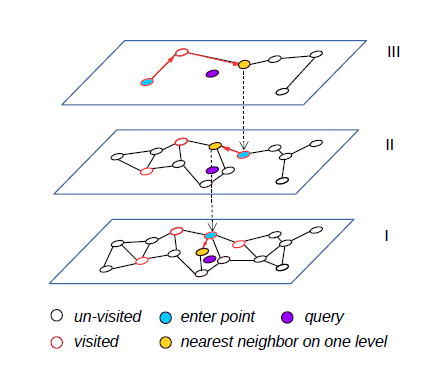


Figure 5 The illustration of HNSW structure

Each layer of HNSW is an NSW graph. The search starts from the top layer. The discovered nearest neighbour from the current layer will be used as the entry point for the NN search on the next layer until the bottom layer.

On the non-bottom layers, the NN search is a greedy search of finding the closest neighbour, which will be treated as the starting point of the next layer. On the bottom layer, a standard NN-Descent search is adopted for updating a top-k nearest neighbours list.

**Graph Diversification**

A heuristic pruning strategy is used to enhance the diversity of a vertex's neighbourhood list in HNSW model. As illustrated in Fig 5, sample se is removed from the NN list of sample sa.

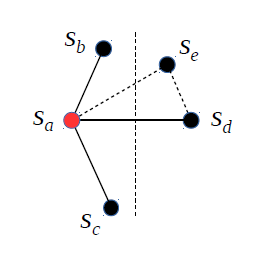


Figure 6 Sample Removal in Graph Diversification

The applied GD algorithm (Fig6) examines each sample in the K-NN list to sample a. A candidate is kept if its distance to sa is smaller than its distances to all the other currently kept samples. The reverse k-NN list of sample sa takes the same strategy and merge with the diversified k-NN list. All the graphs on each layer implement this diversification process which affiliates a more efficient NN search.[[10]](#footnote-11)

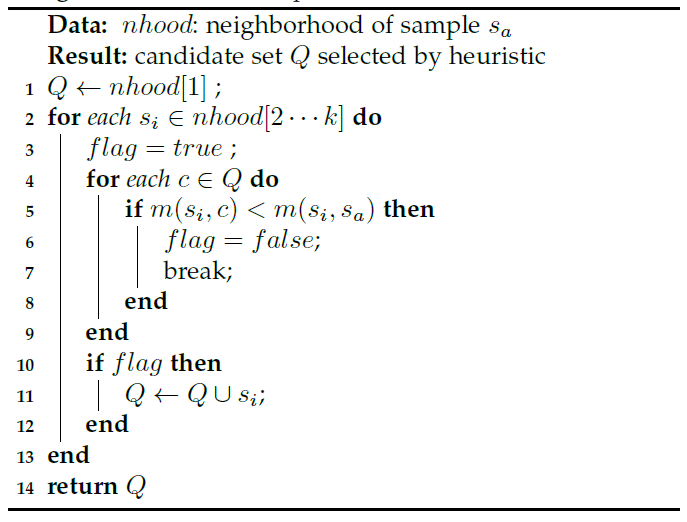


Figure 7 K-NN Graph Diversification Algorithm

* **SVM**

SVM stands for Support Vector Machine. This algorithm comes from the idea of using support vectors to find an optimal hyperplane(solid black line in Fig 8 ) which gives us the furthest distance from the closest data points(support vectors) in a classification or regression problem.[[11]](#footnote-12)

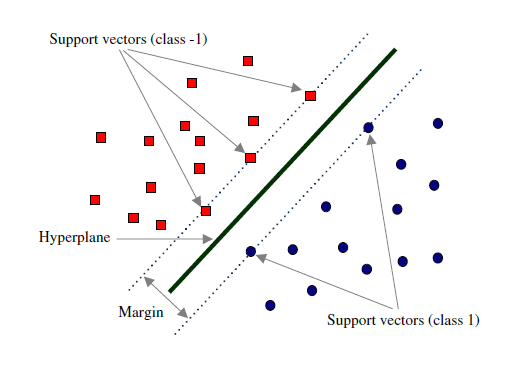


Figure 8 Support Vector Machine Concept Graph

The problem is mathematically interpreted as:

combine the above equations, written as:

The Margin area(M) maximization problem can be interpreted as

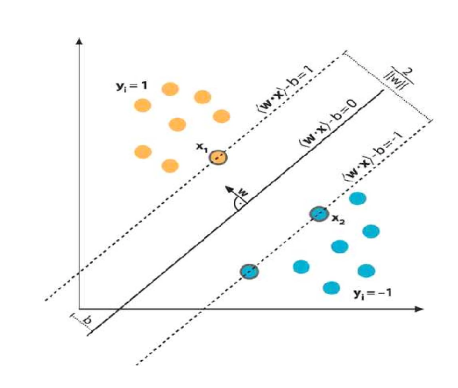


Figure 9 Margin Optimization in SVM

Above is a Primal Optimization problem. This constrained optimization can be converted into an unconstrained optimization problem as . For the solution of both b and w, the problem further moves to Dual Optimization and finally get simplified as follows:

K is the dot product of the input variable x. Finally, we obtain both w and b values as

and b = .

**A generalized version of SVM**

Data can have many shapes, such as linear, circular, ellipsoid, or anything. A more general version of SVM is needed to do the work in all the cases. The method of split or prediction on non-linear shape data is to increase one dimension in the sample space and do the linear split or prediction in the high dimensional space. The mathematical theorem behind it is called Mercer's Theorem. It says if a function ***K(a,b)*** satisfies all the constrains called Mercer's constrains, then there exists a function that maps ***a*** and ***b*** into a higher dimension. A kernel is a function used in this process. In the solution of a basic SVM, a case of linear Kernel

in generalized form

* **Naïve Bayes**

This algorithm works based on the Bayes theorem of probability to predict the class of unknown data set.

* P(c|x) is the posterior probability of class (c, target) given predictor (x, attributes).
* P(c) is the prior probability of class.
* P(x|c) is the likelihood which is the probability of predictor given class.
* P(x) is the prior probability of predictor.[[12]](#footnote-13)

**Types of Naïve Bayes**

* Multinomial Naïve Bayes. This is used mainly for document classification problems based on the features of specific words frequency in the document.
* Bernoulli Naïve Bayes. This is similar to Multinomial Naïve Bayes but the feature value is a boolean.
* Gaussian Naïve Bayes. When the features are continuous values, we assume they have a Gaussian distribution. The conditional probability changes to
* **Logistic Regression**

Logical regression, also known as a generalized linear regression model, is essentially the same as a linear regression model, with ax+b, where a and b are parameters to be solved. The difference lies in their dependent variables. Multiple linear regression directly takes Ax +b as the dependent variable, that is, Y = Ax +b, and logistic regression corresponds ax+b to an implicit state p through the function S,p = S(Ax +b), and then determines the value of the dependent variable according to the magnitude of p and 1-p. Here, the function S is the Sigmoid function:

Chart, histogram

Description automatically generated

Figure 10 Sigmoid function

Log-loss is th function that computes the average cost over the whole training instances. The equation of log-loss is as follows：

* **ANN**

Artificial Neural Network (ANN) is composed of interconnecting artificial neurons (programming constructs that mimic the properties of biological neurons). ANN model could apply different learning rules, including Hebbian learning rule, Perceptron learning rule, Delta learning rule, Correlation learning rule, Outstar learning rule, etc.

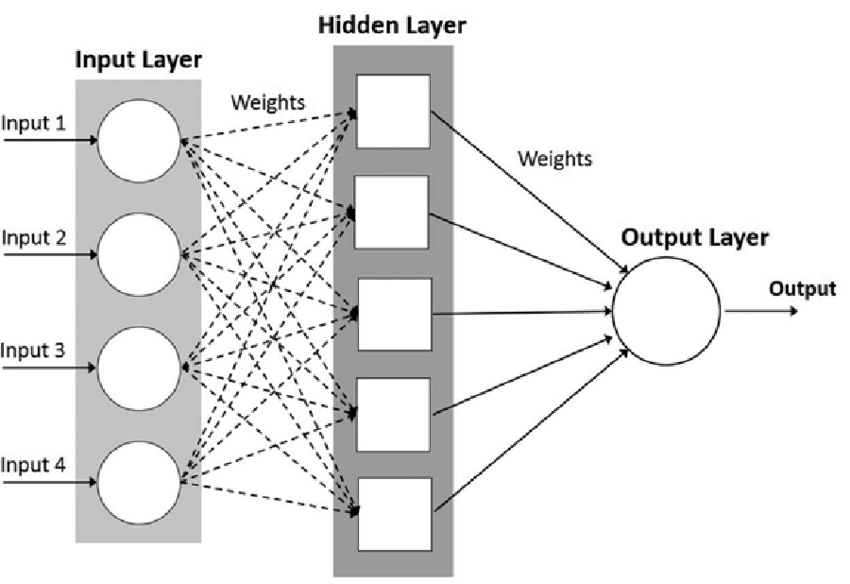


Figure 11 ANN Model

We used the most common learning rule in ANN – Perceptron Learning rule in this project. The network starts its learning by assigning a random value to each weight and bias. The weight and bias of each neuron are adjusted based on error backpropagation. For a unit (neuron) j in the hidden layer, the error is calculated by:

Where wjk  is the weight of the connection from unit j to unit k in the next hidden layer, and Errk is the error of unit k.

Weights are updated by:

Biases are updated by:

* 1. *Comparison*

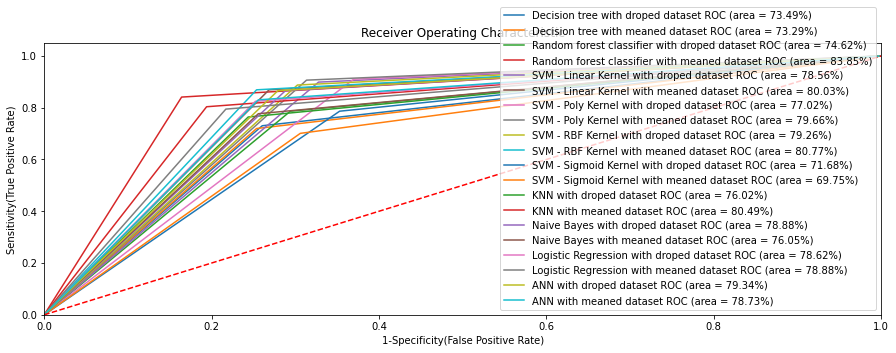
Our goal to study and compare the performance of the above eight different machine learning models is to find the best model for breast detection using Mammogram Mass data. During the experiment, we used different data set of data dropped missing values and data filling with categorical mean values. Cross-validation is used to confirm the valid model accuracy regardless of the data bias from the test data and train data split. ROC curve plots the true positive rate vs the false positive rate. This is different from the model accuracy, which is

Accuracy = . This allows us to focus more on the positive prediction. The confusion matrix gives us a breakdown view of prediction results on different classes. We can further look into the specific error type from the matrix to identify the model problem for future improvement.

From cross-validation metrics,

|  |  |  |
| --- | --- | --- |
| Model | Mean Result | Drop Result |
| Decision Tree | 0.73(+/- 0.07) | 0.73(+/- 0.05) |
| Random Forest | 0.76(+/- 0.07) | 0.76(+/- 0.08) |
| SVM Linear | 0.79(+/- 0.05) | 0.80(+/- 0.10) |
| SVM Poly | 0.79(+/-0.08) | 0.79(+/-0.08) |
| SVM RBF | 0.80(+/-0.07) | 0.80(+/-0.08) |
| SVM Sigmoid | 0.74(+/-0.06) | 0.75(+/-0.08) |
| KNN | 0.80(+/-0.06) | 0.77(+/-0.09) |
| HNSW | N/A | N/A |
| Naïve Bayes | 0.75(+/-0.08) | 0.79(+/-0.11) |
| Logistic Regression | 0.80(+/-0.06) | 0.81(+/0.08) |
| Neural Network | N/A | N/A |

Combining each ROC graph in one figure gives us a clear view of the current best model for mammography prediction. From the graph, the Random Forest using the data filling with mean values has a largest area of 83.85%.

Figure 12 ROC Comparison

1. Conclusion

This project explored the prediction results using eight different machine learning algorithms on two types of Mammogram Mass data set.

Based on the average value from the cross-validation metrics, we can see SVM, KNN and Logistic regression are better than Decision Trees, Random Forest and Naïve Bayes. Overall, the simple model logistic regression is the best one.

Based on ROC-AUC metrics, five models using mean value data set performances are slightly better than the models using dropped value data set, which gives us the best performance of 83.85% from the Random Forest model. In decision tree, SVM with sigmoid kernel, Naïve Bayes and ANN models, the model uses the dropped data set is slightly better. Overall, except the HNSW model gives a significantly lower accuracy than other models(exclude in the comparison ROC graph), the models' performances are close to or above 75%.

We found two interesting things here. First, the simple model performance could be better than the complex ones. Which model solves the problem best is depends on the problems' nature. More computation does not mean more accuracy. Another surprise is that the actual accuracy of the model could take the high-end value of the accuracy range. In this case, the Random Forest model achieves the 83.85% on the test data based on its accuracy range of 0.76(+/-0.7), which we can see from the ROC curve. We doubt this could be some coincidence based on the specific training data and test data split. If we have more data, we can further check if this is really the best model or has an average accuracy of around 0.76.

References

1. Wikipeida. (2021, December 2). Breast cancer. Retrieved from Wikipeida: https://en.wikipedia.org/wiki/Breast\_cancer
2. Beaumont, G. (2021, March 10). Cloud Computing and Artificial Intelligence in Health. Retrieved from Volpara Health: https://www.volparahealth.com/news/cloud-computing-and-artificial-intelligence-in-health/?gclid=Cj0KCQiA-eeMBhCpARIsAAZfxZBwA2WpmIFz-NY\_jE1mLpPu6oZ1wA6VYyfMH20XU7GAM9ovDKtuNS4aAoUhEALw\_wcB
3. SomaDetect. (2021, March 2). SomaDetect raises $6M in Series A funding, led by Ag Capital Canada, to accelerate development of AI technology for the dairy industry. Retrieved from SomaDetect: https://somadetect.com/pr-somadetect-series-a
4. Pedregosa.et.al. (2011). http://jmlr.csail.mit.edu/papers/v12/pedregosa11a.html. Journal of Machine Learning Research, 2825-2830. Retrieved from scikit-learn: https://scikit-learn.org/stable/modules/cross\_validation.html#cross-validation
5. M. Elter, R. S.-W. (2007). The prediction of breast cancer biopsy outcomes using two CAD approaches that both emphasize an intelligible decision process. Medical Physics 34(11), 4164-4172.
6. Brownlee, J. (2020, August 15). Classification And Regression Trees for Machine Learning. Retrieved from Machine Learning Mastery: https://machinelearningmastery.com/classification-and-regression-trees-for-machine-learning/
7. Boots, B. (2020, January 13). Decision Trees: Information Gain. Retrieved from Paul G. Allen School of Computer Science & Engineering: https://courses.cs.washington.edu/courses/cse446/20wi/Lecture3/03\_InformationGain.pdf
8. Sayad, D. S. (2021). Decision Tree - Regression. Retrieved from An Introduction to Data Science: http://www.saedsayad.com/decision\_tree\_reg.htm
9. Agrawal, S. (2021, Janurary 13). RANDOM FOREST. Retrieved from Medium: https://medium.com/swlh/random-forest-ac5227dabb08
10. Wan-Lei Zhao, Hui Wang, Peng-Cheng Lin, Chong-Wah Ngo. (2021, July 29). On the Merge of k-NN Graph. Retrieved from arXiv.org: https://arxiv.org/abs/1908.00814
11. Wan-Lei Zhao, Peng-Cheng Lin, Chong-Wah Ngo. (2020, April 15). On the Merge of k-NN Graph. Retrieved from arXiv.org: https://arxiv.org/abs/1908.00814v5
12. MLMath.io. (2019, February 9). Math behind SVM (Support Vector Machine). Retrieved from Medium: https://ankitnitjsr13.medium.com/math-behind-support-vector-machine-svm-5e7376d0ee4d
13. Ray, S. (2017, September 11). 6 Easy Steps to Learn Naive Bayes Algorithm with codes in Python and R. Retrieved from Analytics Vidhya: https://www.analyticsvidhya.com/blog/2017/09/naive-bayes-explained/
14. Gandhi, R. (2018, May 5). Naive Bayes Classifier. Retrieved from towards data science: https://towardsdatascience.com/naive-bayes-classifier-81d512f50a7c
15. Gao, Z. (2021). Week 5 Logistic Regression. Retrieved from Avenue to learn: https://avenue.cllmcmaster.ca/d2l/le/content/414840/viewContent/3352175/View
16. Fortuna, J. (2021, September). *Neural Networks.* Retrieved from Avenue to learn: https://avenue.cllmcmaster.ca/d2l/le/content/427252/viewContent/3357625/View

1. (Wikipeida, 2021) [↑](#footnote-ref-1)
2. (Beaumont, 2021) [↑](#footnote-ref-2)
3. (SomaDetect, 2021) [↑](#footnote-ref-3)
4. (Pedregosa.et.al, 2011) [↑](#footnote-ref-4)
5. (M. Elter, 2007) [↑](#footnote-ref-5)
6. (Brownlee, 2020) [↑](#footnote-ref-6)
7. (Sayad, 2021) [↑](#footnote-ref-8)
8. (Agrawal, 2021) [↑](#footnote-ref-9)
9. (Wan-Lei Zhao, Hui Wang, Peng-Cheng Lin, Chong-Wah Ngo, 2021) [↑](#footnote-ref-10)
10. (Wan-Lei Zhao, Peng-Cheng Lin, Chong-Wah Ngo, 2020) [↑](#footnote-ref-11)
11. (MLMath.io, 2019) [↑](#footnote-ref-12)
12. (Ray, 2017) [↑](#footnote-ref-13)