The Meaningful Features of judging the malignancy of breast cancer cells Base on Data Mining

17 计算机科学与技术（全英文教学）　邱翼钦　1725171026

Abstract:

This paper analyzes the mutual information from datasets and evaluate them by algorithms. In probability theory and information theory, the mutual information (MI) of two random variables is a measure of the mutual dependence between the two variables. With the development of this project, it was possible to verify that among the available characteristics of the cancer, it is highly feasible to predict whether it is malign or benign by just considering Clump Thickness, Uniformity of Cell Size and Bare Nuclei characteristics.

Keywords: Data mining; MI (Mutual information); Decision tree; ID3 algorithm

Table of Contents

1. Introduction

The aim of this project is to extract any useful information from the downloaded dataset, and find whether it was possible to verify that among the available characteristics of the cancer.

Two possible datasets were given as a choice:

a. Breast Cancer Wisconsin Original

(available at "https://archive.ics.uci.edu/ml/datasets/Breast Cancer Wisconsin Original")

b. Mammographic Mass

(available at "https://archive.ics.uci.edu/ml/datasets/Mammographic Mass").

Since there was the choosing option, the first dataset was chosen for processing.

For the purpose of meaningful information extraction, it is considered the selection of meaningful features and meaningful combination of features. As an upgrade to check if the information was correctly extracted, the developed MATLAB program includes the construction of decision trees in order to evaluate the correctness of the obtained results.

2. Algorithm

A. Mutual Information (MI)

The Mutual Information between two random variables can be defined as the measure of mutual dependence between them and quantifies the amount of information that can be obtained about a variable through the other. It is highly linked with the concept of entropy of a random variable, which defines the amount of information held in a random variable.

The Mutual Information can be obtained using the following formula:

$$I(X;Y) = \sum_{i=1}^{N_X} \sum_{j=1}^{N_Y} P(x_i; y_i) log_2 \frac{P(x_i; y_i)}{P(x_i)P(y_i)}$$

This measure of information can also be displayed in a Venn diagram as shown in figure 1.

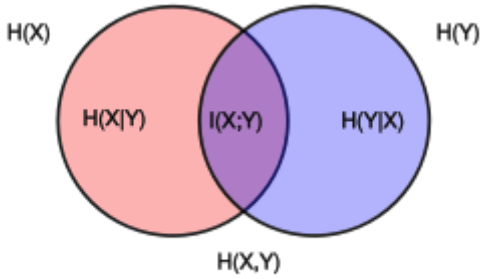

Figure 1 - Venn diagram of Mutual Information

B. Decision Tree

Decision tree builds classification or regression models in the form of a tree structure. It breaks down a dataset into smaller and smaller subsets while at the same time an associated decision tree is incrementally developed. The result is a tree with decision nodes and leaf nodes. A decision node (e.g., Outlook) has two or more branches (e.g., Sunny, Overcast and Rainy). Leaf node (e.g., Play) represents a classification or decision. The topmost decision node in a tree which corresponds to the best predictor called root node. Decision trees can handle both categorical and numerical data.

Figure 2 – Decision Tree

C. ID3 Algorithm

The ID3 algorithm begins with the original set S as the root node. On each iteration of the algorithm, it iterates through every unused attribute of the set S and calculates the entropy $H(S)$ or the information gain $IG(S)$ of that attribute. It then selects the attribute which has the smallest entropy (or largest information gain) value. The set S is then split or partitioned by the selected attribute to produce subsets of the data. (For example, a node can be split into child nodes based upon the subsets of the population whose ages are less than 50, between 50 and 100, and greater than 100.) The algorithm

continues to recurse on each subset, considering only attributes never selected before.

Recursion on a subset may stop in one of these cases:

- every element in the subset belongs to the same class; in which case the node is turned into a leaf node and labelled with the class of the examples.
- there are no more attributes to be selected, but the examples still do not belong to the same class. In this case, the node is made a leaf node and labelled with the most common class of the examples in the subset.
- there are no examples in the subset, which happens when no example in the parent set was found to match a specific value of the selected attribute. An example could be the absence of a person among the population with age over 100 years. Then a leaf node is created and labelled with the most common class of the examples in the parent node's set.

Throughout the algorithm, the decision tree is constructed with each non-terminal node (internal node) representing the selected attribute on which the data was split, and terminal nodes (leaf nodes) representing the class label of the final subset of this branch.

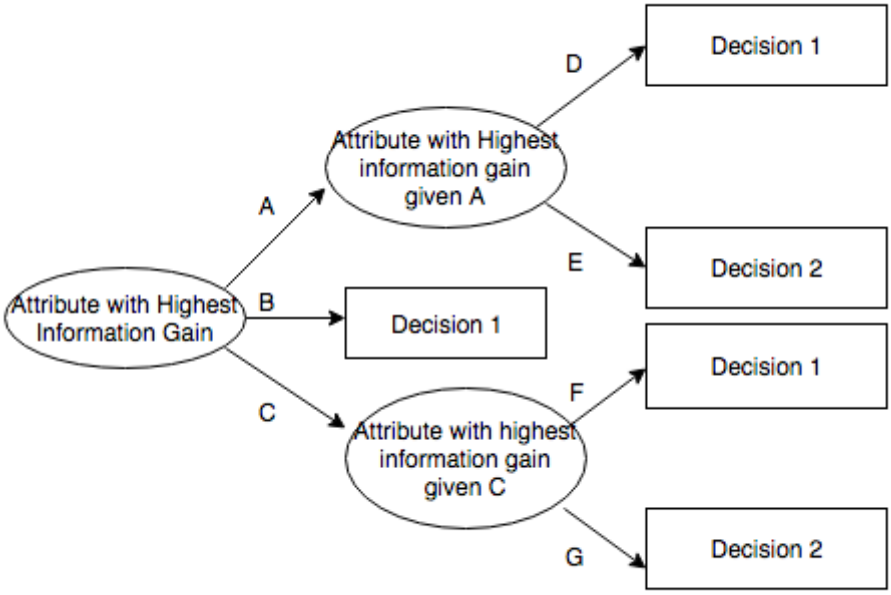

Figure 3 – Potential ID3-generated decision tree

3. Experiments

A. Description of the data

The “Breast Cancer Wisconsin Original” dataset contains eleven features. Each

feature is disposed in a column of the given data file in the following order:

1. *Sample Code Number* → contains information of the id number, which makes every row of the matrix unique
2. *Clump Thickness* → attribute range is 1-10;
3. *Uniformity of Cell Size* → attribute range is 1-10;
4. *Uniformity of Cell Shape* → attribute range is 1-10;
5. *Marginal Adhesion* → attribute range is 1-10;
6. *Single Epithelial Cell Size* → attribute range is 1-10;
7. *Bare Nuclei* → attribute range is 1-10;
8. *Bland Chromatin*→ attribute range is 1-10;
9. *Normal Nucleoli*→ attribute range is 1-10;
10. *Mitoses*→ attribute range is 1-10;
11. *Class* → has value "2" if is benign or value "4" if is malign.

It is also known the following information:

- There are sixteen instances from instances 1-6 that contain unknown values/measurements errors that are specified with a "?".
- There are 458 instances representing benign data and 241 representing malign data. The dataset description is provided on "https://archive.ics.uci.edu/ml/machine-learningdatabases/breast-cancer-wisconsin/breast-cancer-wisconsin.names".

B. Preprocessing of the data

After loading the data, missing values are converted to a non-existing integer "0". Two options are available for the processing of these instances: omission (delete the missing instances) or filling these with values computed based on the other records. Since the data has not yet been processed and according to the instructions given on the assignment, these instances were deleted.

Due to the nature of the data and the same ranges for each feature, it is not needed to perform Normalization, Discretization or Dimensionality Reduction.

À priori, and considering the description provided by the owners of the dataset, it is possible to remove the first column of the data matrix which only purpose is to identify each entry as unique.

C. Meaningful Features and Feature Selection

In order to maximize the performance of the data mining method while reducing the number of features, it is important to perform feature selection and identify the meaningful features. Therefore, Mutual Information proves to be a good indicator of relations between the input feature and the target variable.

1. Meaningful Single Feature evaluation

Meaningful features can be ranked according to their entropy and their MI with the class feature. By running the MATLAB code, it is possible to obtain the following information about the single features:

Table 1 - Entropy of each feature

Feature	**Entropy**
Clump Thickness	3.049588
Uniformity of Cell Size	2.343874
Uniformity of Cell Shape	2.489039
Marginal Adhesion	2.212998
Single Epithelial Cell Size	2.290806
Bare Nuclei	1.992490
Bland Chromatin	2.769368
Normal Nucleoli	2.051699
Mitoses	1.129896

Table 2 - Mutual Information between each feature and the class feature

Rank	Feature	MI
1	Uniformity of Cell Size	0.702333
2	Uniformity of Cell Shape	0.676771
3	Bare Nuclei	0.603095
4	Bland Chromatin	0.555260
5	Single Epithelial Cell Size	0.534426
6	Normal Nucleoli	0.487187
7	Marginal Adhesion	0.464424
8	Clump Thickness	0.463995
9	Mitoses	0.211958

According to the obtained information, if only one feature is available, *Uniformity of Cell Size* is the feature that gives the user a most accurate prediction of the class feature. In this case, is the random variable that can by its own given a most accurate prediction about the Breast Cancer being benign or malign.

Although this information is a lot useful, we should also check whether a combination of the available random variables could result in better prediction.

2. Meaningful Feature combinations

With the purpose of checking the information provided by a group of random variables, there are two possible approaches:

1. Consider that all variables are independent from one another and therefore use the previously obtained MI to group the n random variables that are better correlated with the class.
2. Consider that the random variables can have some dependency among each other and calculate the MI for each pair of random variables with the class feature. Since we do not know anything à priori and no information is provided in these terms on the description file given by the dataset creators, the program is built according to the second possibility. Also, this situation is justified since other diseases also cause a group of symptoms that are correlated between them, so breast cancer probably has similar effects on people.

Some aggregations were made and the best MI achieved for each case is displayed on

table 3.

Table 3 - Best results achieved in MI for a certain feature group

Number of combined features	Combined Features	MI
2	Uniformity of Cell Size & Bare Nuclei	0.846539
3	Clump Thickness & Uniformity of Cell Size & Bare Nuclei	0.920467
4	Uniformity of Cell Shape & Marginal Adhesion & Bare Nuclei & Normal Nucleoli	0.934003
5	Clump Thickness & Uniformity of Cell Size & Bare Nuclei & Bland Chromatin & Mitoses	0.934003
6	Clump Thickness & Uniformity of Cell Size & Marginal Adhesion & Bare Nuclei & Bland Chromatin & Mitoses	0.934003

In these terms, there should be a compromise between the achieved MI and the number of features. It is possible that a higher number of features, with a higher MI with the class feature gives a worst prediction since the results are too much linked with the data extracted and not the universe of possible results. To do an evaluation of the obtained results, the following graph was plotted:

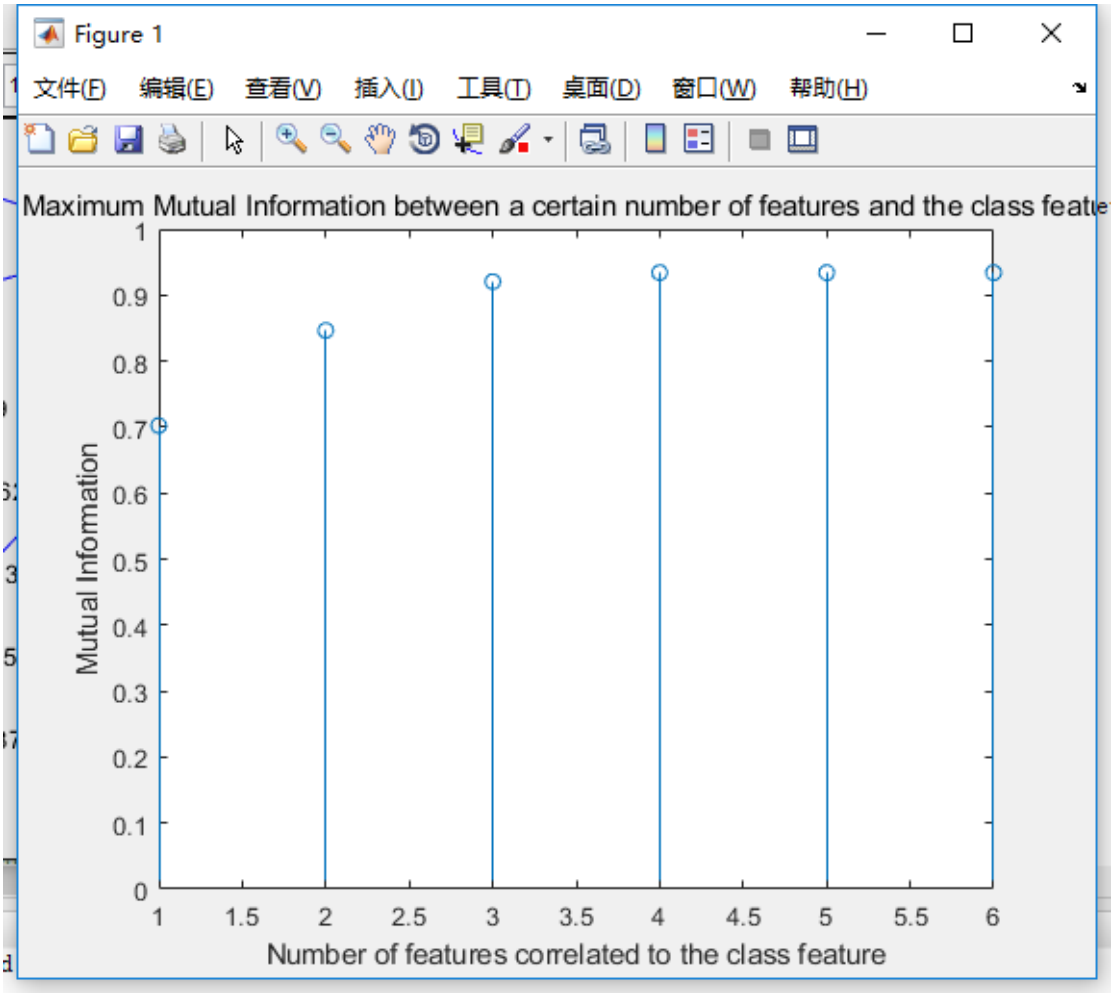

Figure 4 – Relationship between the MI with the class features and the number of features aggregated

There is no bound to the best correlation between the variables and the desired classification.

Therefore, the method used to determine the number of features to be used is the "knee method". It is expected that by looking at the figure 4 graph, the existent "knee" at 3 variables indicates that this aggregation gives the needed information to predict if the cancer is malign or benign.

Choosing several random variables after the "knee" to describe the class feature might induce error in future classifications, although it provides a higher MI. This way, classification of the training data will be improved but when classifying external data from the general universe, results will have more errors. This phenomenon is called overfitting.

The "knee" method is largely used for PCA and can be transposed to the current problem analysis.

It is possible to conclude through the graph observation that the most meaningful combination of features is: **Clump Thickness & Uniformity of Cell Size & Bare Nuclei.**

D. Parallel confirmation

Apart from the confirmation that will be performed by the construction of the decision tree, there is a built-in function in MATLAB to perform feature selection. It was decided by me to include a section of code dedicated to this function that is used only for comparison purposes with the MI method.

The functions used are *classify* and *sequentialfs* that rely on machine learning algorithms. Therefore, data was divided into training set and test set. The information about the functions can be obtained in the respective webpages:

- "https://www.mathworks.com/help/stats/classify.html"
- "https://www.mathworks.com/help/stats/sequentialfs.html".

It is important to refer that the existing loop in the code is due to the fact that the *sequentialfs* function relies on the *Monte Carlo method* and therefore is highly likely that two different runs of the function provide different main features. The result used for comparison rely only on the features that were chosen by the function in 75% of the runs. As expected, the features match the ones obtained in the previous method.

```
%Confirmation of the results through Matlab System Functions
X=data_BreastCancer(:, 2:10);
Y=data_BreastCancer(:, 11);

x_train = X(1:478, :);
x_test = X(479:end, :);
y_train = Y(1:478);
y_test = Y(479:end);
ypred = classify(x_test, x_train, y_train);
sum(y_test ~= ypred);
f=@(x_train, y_train, x_test, y_test) sum(y_test~=classify(x_test, x_train, y_train));

count=zeros(1,9);
for i=1:1:20
    inmodel = sequentialfs(f,X,Y);
    count=count+inmodel;
end
Final_inmodel=count>15;
[string, columns]=SpecifyFeatures(Final_inmodel);
disp(string);
```

Figure 5 - Code for Parallel Confirmation Algorithm

Then we can get the result: **Through the MATLAB built in function, the selected features are: Clump Thickness, Uniformity of Cell Size, Bare Nuclei**.

E. Classification tree

At this point, the relevant features and their best combination was already achieved, which means that the dataset useful information was already extracted. However, to check the feasibility of the information, a decision tree was built.

1. MATLAB Built-in functions test

Decision trees can be easily built using MATLAB functions. Since the aim of this section is the proof of the feasibility of the previous results, a classification tree was built instead of a regression tree. The built-in function name is called *fitctree*. After building the regression tree, the dataset was submitted to the classification through the predict function. The results are shown in table 4.

Table 4 - Number of classification errors according to the number of combined features

Number of combined features	**Number of classification errors**
2	40
3	25
4	22
5	10

The above result might seem that the previously obtained information was not correct since the number of errors keeps to be reduced as a new feature is introduced in the built of the decision tree. However, the reader should keep in mind that the above results have its origin in a tree that was built and tested with the same data points. Therefore, the reduction of the number of errors are a consequence of overfitting and not a result of a better correspondence between the variables and the class feature.

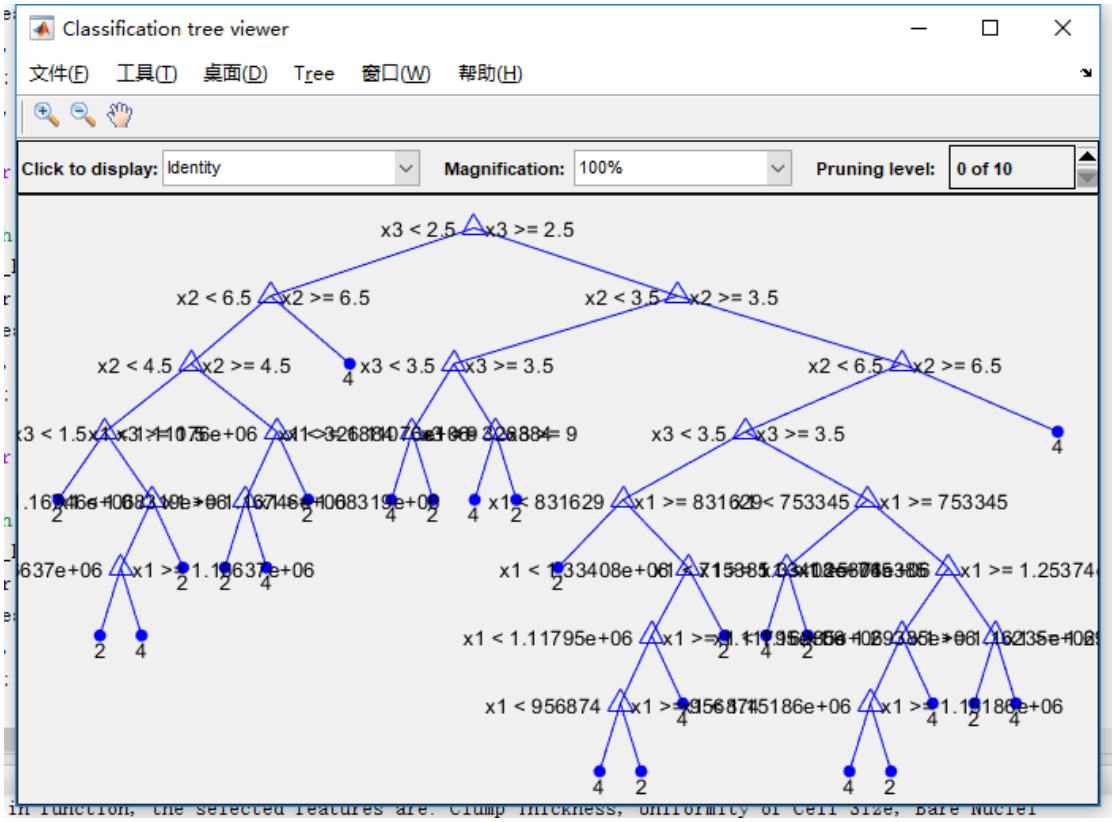

Figure 6 – Decision Tree

2. ID3 Algorithm

A decision tree can also be built according to entropy measures. This algorithm is called ID3 and is "a greedy search top-down divide and conquer algorithm to build a decision tree, picking the best attribute at each node and never looking back to reconsider early choices in the past".

Unfortunately, it was not possible for me to concretize this algorithm in MATLAB due to deadlines to deliver the project. MATLAB is a language directed to programming using vectors and matrices and the best way to code ID3 is using recursive construction of a tree made with pointers and node structures. Although the language provides the possibility to create data structures ("https://www.mathworks.com/help/matlab/ref/struct.html"), it was not possible for me in the given time to overcome the pointers inexistence problem.

The pseudo-code for the given algorithm is the following:

Function ID3

- Create a root node for the tree
- If all members are classified as benign (class=2), return single-node tree root with label 2
- If all members are classified as malign (class=4), return single-node tree root with label 4
- If there are no more features, but members do not belong to the same class, return single node tree root with label=most common value of class feature
- Otherwise:

  A←Attribute that provides better classification
  Decision tree attribute for root=A For each possible
  value of A (k_i) do:

Add a new tree branch bellow root corresponding to the test A=k_i

Call ID3 function for each branch with the correspondent new subset of

samples (subset of samples that have k_i as a value for A)

Figure 7 - Pseudo-code for ID3 algorithm

The way of checking which attribute/feature that provides better classification is done by checking its MI with the class feature.

Although the algorithm was not implemented, it is expected that it would provide similar results as the ones provided by MATLAB built-in functions.

F. Result

The result of code for MATLAB is shown:

The entropy of Clump Thickness is 3.049588

The entropy of Uniformity of Cell Size is 2.343874

The entropy of Uniformity of Cell Shape is 2.489039

The entropy of Marginal Adhesion is 2.212998

The entropy of Single Epithelial Cell Size is 2.290806

The entropy of Bare Nuclei is 1.992490

The entropy of Bland Chromatin is 2.769368

The entropy of Normal Nucleoli is 2.051699

The entropy of Mitoses is 1.129896

The most relevant features are the following (by order):

Uniformity of Cell Size with 0.702333 mutual information with the class feature

Uniformity of Cell Shape with 0.676771 mutual information with the class feature

Bare Nuclei with 0.603095 mutual information with the class feature

Bland Chromatin with 0.555260 mutual information with the class feature

Single Epithelial Cell Size with 0.534426 mutual information with the class feature

Normal Nucleoli with 0.487187 mutual information with the class feature

Marginal Adhesion with 0.464424 mutual information with the class feature

Clump Thickness with 0.463995 mutual information with the class feature

Mitoses with 0.211958 mutual information with the class feature

---

The most relevant association of 2 features is Uniformity of Cell Size with Bare Nuclei with a mutual information of 0.846539

The most relevant association of 3 features is Clump Thickness with Uniformity of Cell Size and Bare Nuclei with a mutual information of 0.920467

The most relevant association of 4 features is Uniformity of Cell Shape with Marginal Adhesion and Bare Nuclei and Normal Nucleoli with a mutual information of 0.934003

The most relevant association of 5 features is Clump Thickness with Uniformity of Cell Size and Bare Nuclei and Bland Chromatin and Mitoses with a mutual information of 0.934003

The most relevant association of 6 features is Clump Thickness with Uniformity of Cell Size and Marginal Adhesion and Bare Nuclei and Bland Chromatin and Mitoses with a mutual information of 0.934003

Through the previously obtained graph, we can select only the following 3 features: Clump Thickness, Uniformity of Cell Size and Bare Nuclei to be correlated with the class feature

Through the MATLAB built in function, the selected features are: Clump Thickness, Uniformity of Cell Size, Bare Nuclei

---

The number of classification errors using 2 variables is 40

The number of classification errors using 3 variables is 25

The number of classification errors using 4 variables is 22

The number of classification errors using 5 variables is 10

For these algorithms, we can find that the feature combinations are more meaningful than single feature in MI (Mutual information).

MI, Parallel Confirmation and Classification Tree (contains ID3 Algorithm) all give birth to the same result, the algorithms adopted in this paper and their verification methods have the same validity. MI is the most complex, but it yields the most intuitive results. Classification Tree (contains ID3 Algorithm) needs validation from other algorithms to get accurate results and it takes a little too long to execute. Compared to other algorithms, parallel confirmation can confirm and get results in the shortest time and it has low code implementation complexity.

4. Conclusion

With the result of this project, it was possible to verify that among the available characteristics of the cancer, it is highly feasible to predict whether it is malign or benign by just considering *Clump Thickness*, *Uniformity of Cell Size* and *Bare Nuclei characteristics*.

References:

[1] "https://www.mathworks.com/help/stats/classify.html"

[2] "https://www.mathworks.com/help/stats/sequentialfs.html".

[3] EG Learned-Miller. Entropy, Joint entropy, and mutual information. 2013

[4] "http://athena.ecs.csus.edu/~mei/177/ID3_Algorithm.pdf"