Project 2 – Breast Cancer Decision Tree

Course	INFO-3135, Data Structures and Algorithms
Professor	Janice Manning
Assigned	Monday, November 11, 2024
Due	Sunday, December 1, 2024 by 11:59 pm
Weight	15%

Introduction:

This project is based on an article which was published in International Journal of Computer Applications, July 2014. See the link below:

https://www.researchgate.net/publication/272863357_Diagnosis_of_Breast_Cancer_using_Decision_Tree_Data_Mining_Technique

The article presents a data mining technique based on a decision tree for early detection of breast cancer. Breast cancer diagnosis differentiates benign from malignant breast tumours. Benign tumours have not yet spread to other parts of the body whereas malignant tumours have already spread to other parts of the body. The study used a Binary Decision tree to determine if a patient has benign or malignant breast cancer tumours to within a 94.5% accuracy.

Project Specifications:

Build a binary decision tree using the logic shown in Figure 13, which will run on a dataset using the attributes shown in Table 3. The data to run is in the csv file in the format shown in Figure 12, with the diagnosis shown in Column K as "0" to start.

Read in patients' data from a file called **unformatted_data_v1.0.0.csv**. Note that you must validate the patient data for the correct quantity of data because there will be patients lacking enough data to work through the decision tree. If a patient's data set is invalid, discard the patient and maintain a counter. Optionally, report an error message to the console identifying the patient.

Then run the data through the binary decision tree and output the results to a file as shown with the diagnosis Benign or Malignant (Figure 12), where benign is represented by 2 and malignant is represented by 4.

Store the results in a new file named **results.csv** following the same format as the input file, replacing Column K "0" with "2" or "4" based on the result for that row.

After processing all the rows in the console, print a summary showing the following:

- Total Patients Processed
- Total Benign
- Total Malignant
- Total Invalid Patient

```
Total Patients Processed:   683
Total Benign:               443
Total Malignant:            240
Total Invalid Patient:      16
```

Input uses the Wisconsin Breast Cancer dataset attributes (see Table 3 below), which have been determined to satisfy the prerequisites of the data mining technique (see Figure 13 next page).

Table 3. Wisconsin Breast Cancer Dataset Attribute

S.No	Attribute	Domain
1	Sample Code Number	Id number
2	Clump Thickness	1 – 10
3	Uniformity of Cell Size	1 – 10
4	Uniformity of Cell Shape	1 – 10
5	Marginal Adhesion	1 – 10
6	Single Epithelial Cell Size	1 – 10
7	Bare Nuclei	1 – 10
8	Bland Chromatin	1 – 10
9	Normal Nucleoli	1 – 10
10	Mitoses	1 – 10
11	Class	2(Benign) or 4(Malignant)

Input is a dataset stored in the csv file. There are 11 columns which correspond to 11 attributes shown in Table 3. Each row in the csv file can be represented as a Patient object holding the attributes. The Class column has been set to 0 for now. It is this attribute that will be calculated using your decision tree.

The data represents 699 instances of patients' test results taken from their biopsy samples. The attributes' values are within a range from 1-10. These values will be compared to test cases in the decision tree. The objective is to find a classification of benign or malignant using the tree for each patient.

Example of unformatted data taken from Row 1 in the csv file:

	A	B	C	D	E	F	G	H	I	Formula Bar	K
1	1000025	5	1	1	1	2	1	3	1	1	0

Images Taken from the article:

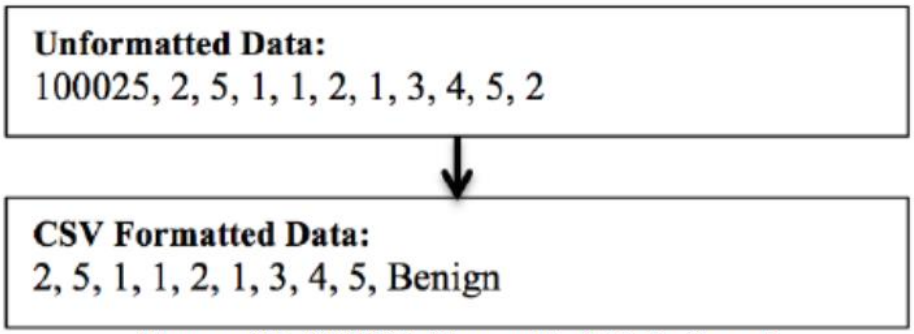

Figure 12. WEKA Formatted Data Input

Use **Figure 13 J48 Pruned Tree** shown below as the logic for your decision tree which outlines the rules.

```
Uniformity of Cell Size <= 2
|   Bare Nuclei    <= 3: Benign (405.39/2.0)
|   Bare Nuclei    > 3
|   |   Clump Thickness <= 3: Benign (11.55)
|   |   Clump Thickness > 3
|   |   |   Bland Chromatin <= 2
|   |   |   |   Marginal Adhesion <= 3: Malignant (2.0)
|   |   |   |   Marginal Adhesion > 3: Benign (2.0)
|   |   |   Bland Chromatin > 2: Malignant (8.06/0.06)
Uniformity of Cell Size > 2
|   Uniformity of Cell Shape <= 2
|   |   Clump Thickness <= 5: Benign (19.0/1.0)
|   |   Clump Thickness > 5: Malignant (4.0)
|   Uniformity of Cell Shape > 2
|   |   Uniformity of Cell Size <= 4
|   |   |   Bare Nuclei    <= 2
|   |   |   |   Marginal Adhesion <= 3: Benign (11.41/1.21)
|   |   |   |   Marginal Adhesion > 3: Malignant (3.0)
|   |   |   Bare Nuclei    > 2
|   |   |   |   Clump Thickness <= 6
|   |   |   |   |   Uniformity of Cell Size <= 3: Malignant (13.0/2.0)
|   |   |   |   |   Uniformity of Cell Size > 3
|   |   |   |   |   |   Marginal Adhesion <= 5: Benign (5.79/1.0)
|   |   |   |   |   |   Marginal Adhesion > 5: Malignant (5.0)
|   |   |   |   Clump Thickness > 6: Malignant (31.79/1.0)
|   |   Uniformity of Cell Size > 4: Malignant (177.0/5.0)
```

Steps to Writing the Project Code:

1. Write a Patient class (see Table 3 and csv data file).
2. Write 12 decision bool functions, based on left-hand tests (see Figure 13, J48 Pruned Tree).
3. Write a BinaryDecisionTree class which includes a Node struct which has a function pointer to a decision function, and Node* left, Node* right.
4. Write a buildDecisionTree function which builds the tree from the bottom up.
5. Write a classifyPatient function, to process each patient's data traversing the decision tree to get to a result of 2 or 4.
6. Write a main class which reads in the csv file data. Make calls to build the tree and then to classify the patient data. Output a summary report and save the updated patient data.

Submission Guidelines:

Submit your file to the "Project 2 – Breast Cancer Decision Tree" submission folder on FOL. Make sure to add your name to all documents.

Submit your project on time!
Submissions must be made on time! Late projects will be subject to divisional policy on missed test and late projects. There is a penalty of 10% per day for a maximum of 5 days after which the submission will receive a zero grade.

Submit your own work and *keep it to yourself*!
It is considered cheating to submit work done by another student or from another source. Helping another student cheat by sharing your work with them is also not tolerated. Students are encouraged to share ideas and to work together on practice exercises, but any code or documentation prepared for a project must be done by the individual student. Penalties for cheating or helping another student cheat may include being assigned zero on the project with even more severe penalties if you are caught cheating more than once. Just submit your own work and benefit from having made the effort on your own.
All work will be subject to "TurnItIn" scrutiny.

Grading Scheme:

Functional Requirements	Marks Available	Marks Awarded
Correctly diagnose the patient data using the decision tree	10	
Input from unformatted_data.csv file parsed correctly	10	
Output results.csv file with the diagnosis in the correct location of the file. Note: Output must match the format of the input file with the diagnosis (0) replaced with a (2 or 4). Do not overwrite the input file.	10	
Print a summary showing the following: • Total Patients Processed • Total Benign • Total Malignant • Total Invalid	10	
Program runs, creates the appropriate files, outputs and displays to console the total number of patients, number of patients diagnosed with benign tumours and malignant tumours	10	
Non-functional Requirements		
Code documentation: • File headers • Function headers Code Style: • use of naming conventions • Use of functions • Descriptive comments within code	5 5	
Penalties		
• Plagiarism	-100%	
• Does not compile	-100%	
• Late (10% per day up to 5 days)		
Total	**60**	