Fatality Classification Based on Abnormalities in the Brain

Anisah Binti Kamsin, Jeya Jassvine A/P Jeyabala Sundram, Mohamad Zairi Bin Abd Ghani

***Abstract*— Data can be in various levels of complexity and noise where a better algorithm needs to be developed by replacing missing values using mode to enhance the process of mining the data to get the hidden and useful information behind it. There are a lot of techniques that can be done to find solutions for the problems that we want to solve by having a set of data. Clustering is one of the methods that we're going to highlight in this project. We decided to implement a knowledge and data driven approach using fuzzy logic by implementing the Mamdani method as well as the Fuzzy C-Means technique. We used R Programming and Matlab to program our designed algorithm as this programming language provides many types of built-in functions that help us to analyze the data efficiently and to visualize them in a better way. We compare both Mamdani method and Fuzzy C-Means results in terms of the clusters formed, accuracy and few more metrics to validate the result.**

***Index Terms*—brain mri, fatality, clustering, brain abnormalities, Mamdani method, fuzzy c-means.**

I. Introduction

Magnetic resonance imaging (MRI) is a common imaging technique used extensively to study human brain activities.In today's world, the images used are in digital format. In recent times, the introduction of information technology and e-healthcare systems in the medical field helps clinical experts to provide better healthcare for patients. Recently, it has been used for scanning the fetal brain. Amongst 1000 people, 3 of them have brain abnormalities. Hence, the primary detection and classification is important. Abnormalities in the human brain include brain diseases and brain tumors. Although neurons are the longest living cells in the body, large numbers of them die during migration and differentiation. The lives of some neurons can take abnormal turns. Some diseases of the brain are the outcome of the unnatural deaths of neurons. In Parkinson's disease, neurons that produce the neurotransmitter dopamine die off in the basal ganglia, an area of the brain that controls body movements. This causes difficulty initiating movement. In Huntington's disease, a genetic mutation causes over-production of a neurotransmitter called glutamate, which kills neurons in the basal ganglia. As a result, people twist and writhe uncontrollably. In Alzheimer's disease, unusual proteins build up in and around neurons in the neocortex and hippocampus, parts of the brain that control memory. When these neurons die, people lose their capacity to remember and their ability to do everyday tasks. Physical damage to the brain and other parts of the central nervous system can also kill or disable neurons. Blows to the brain, or the damage caused by a stroke, can kill neurons outright or slowly starve them of the oxygen and nutrients they need to survive. Spinal cord injury can disrupt communication between the brain and muscles when neurons lose their connection to axons located below the site of injury. These neurons may still live, but they lose their ability to communicate.

Tumors are formed in the brain and can be very dangerous. These are called primary brain tumors. In other cases, cancer somewhere else in your body spreads to your brain. These are called secondary or metastatic brain tumors.

Brain tumors can be either malignant (cancerous) or benign (noncancerous). Doctors classify brain tumors as grades 1, 2, 3, or 4. Higher numbers indicate more aggressive tumors.

The cause of brain tumors is largely unknown. They can occur in people of any age. Symptoms of brain tumors depend on the size and location of the tumor.

Specialists detect chances of fatality depending on their experience. Sometimes the results are not accurate, so that this project can help specialists to detect chances of fatality.

II. Proposed Predictive Modelling

The predictive model proposed in this paper consists of three major components, i.e. (1) the data acquisition and pre-processing module, (2) mamdani, and (3) fuzzy c-means clustering.

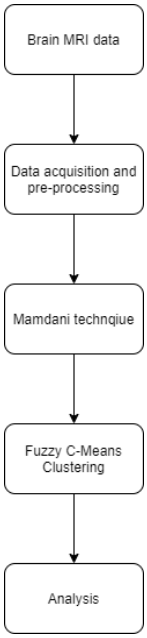

Figure 1: The Predictive Model

A. The Data Acquisition and Pre-processing Module

The raw Brain MRI Segmentation data obtained through Kaggle platform is used in the model training phase to realise the predictive model. The data is first processed to prepare the data for the model training and prediction activity where it is checked for missing values. The NA values have been replaced with mode value to preserve accuracy and then feature selection was carried out for the data driven technique.

B. Feature Selection

The feature selection technique is done to complete the data driven technique. The reason why we do feature selection is to reduce complexity of the clustering process as we only use the features that are considered important. It also improves the performance of the clustering technique. In this project, we used the Boruta technique. The Boruta technique is an improvised version of the random forest classifier which is the conventional technique.

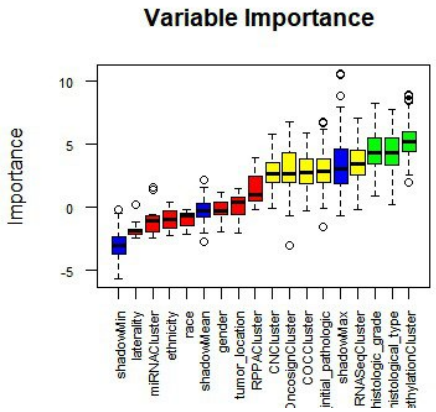

Figure 2 : Variable Importance

The variable importance shows the selected as well as rejected features. In this case, the 3 green attributes have been selected which is *MethylationCluster, neoplasm_histologic_grade, histological_type.* The yellow features are known as tentative attributes which means the algorithm cant decide whether to reject to accept as the importance value of these attributes are very close to the ShadowMax attribute.

To solve this we can run a rough fix function and get a new output which involves the selected tentative attributes if there are any. In our case one of the tentative attributes was selected which is the *age_at_intial_pathologic.* So in total we have 4 selected attributes from the feature selection technique to be used in the Fuzzy C-Mean Clustering.

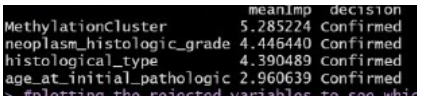

Figure 3 : Selected attributes from feature selection using Boruta

C. Mamdani

Mamdani fuzzy inference was first introduced as a method to create a control system by synthesizing a set of linguistic control rules obtained from experienced human operators. In a Mamdani system, the output of each rule is a fuzzy set. Since Mamdani systems have more intuitive and easier to understand rule bases, they are well-suited to expert system applications where the rules are created from human expert knowledge, such as medical diagnostics.

D. *Fuzzy C-Means Clustering*

Fuzzy C-Means Clustering gives the values of any point lying in some particular cluster to be either 0 or 1, Fuzzy C-Means (FCM) gives the fuzzy values of any particular data point to be lying in either of the clusters. Datasets that have a high fuzziness level are very suitable to be injected with a fuzzy method into its solution's algorithm.

For this project, we chose Matlab to model our Mamdani and R Programming Language to model our solution in Fuzzy C-Means due to its simplicity in handling data as well as visualizing them in an appropriate way by just calling the built-in function `fcm()` is the function used to handle clustering processes injected with a fuzzy method. The algorithm used inside the function is an iterative clustering algorithm with the objective function as below:

$$J_{FCM}(\mathbf{X};\mathbf{V},\mathbf{U}) = \sum_{i=1}^{n} u_{ij}^{m} d^2(\vec{x}_i, \vec{v}_j)$$

Figure 4: Iterative Clustering Algorithm

where m is the fuzzifier to specify the amount of 'fuzziness' of the clustering result: $1 \le m \le \infty$.

Since it is an iterative clustering algorithm, the objective function of FCM is minimized by using the following update equations:

$$u_{ij} = \left[\sum_{j=1}^{k} \left(\frac{d^2(\vec{x}_i, \vec{v}_j)}{d^2(\vec{x}_i, \vec{v}_l)} \right)^{1/(m-1)} \right]^{-1} \quad ; 1 \le i \le n,\ 1 \le l \le k$$

$$\vec{v}_j = \frac{\sum_{i=1}^{n} u_{ij}^{m} \vec{x}_i}{\sum_{i=1}^{n} u_{ij}^{m}} \quad ; 1 \le j \le k$$

Figure 5: Equation

Table 1: Description for Figure 3

VALUE	DESCRIPTION
x	a numeric matrix containing the processed data set
v	a numeric matrix containing the final cluster prototypes (centers of clusters)
u	a numeric matrix containing the fuzzy membership degrees of the data objects
d	a numeric matrix containing the distances of objects to the final cluster prototype
k	an integer for the number of clusters
m	a number for the fuzzifier

III. DESIGN

A. *Mamdani*

For Mamdani, we used Matlab to solve this problem. First, we collect the data from several websites and experts as below.

Attribute	Fuzzy set representation	Range
Age	young	20-38
	mild	33-45
	old	>40
Gender	male	<0.5
	female	>=0.5
TumorTissueSite	TRUE	>120
miRNACluster	dark	10%-30%
	medium	30%-70%
	bright	70%-100%
RNASeqCluster	grade1	<137
	grade2	127-153
	grade3	142-172
	grade4	>154
RPPACluster	grade1	<197
	grade2	178-240
	grade3	227-327
	grade4	>271

Figure 6: Attribute and Range

Next, we set the input, output and also membership function based on the range given using matlab tools.

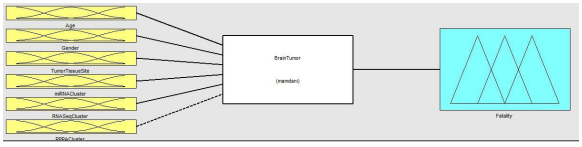

Figure 7: Set the Input and Output

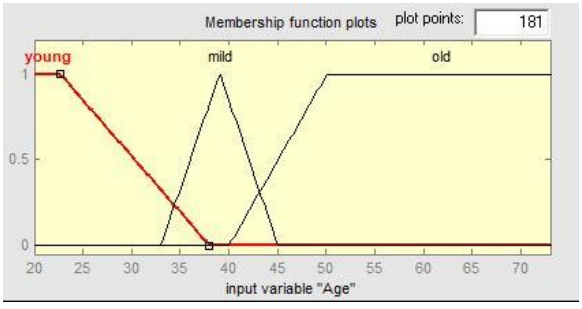

Figure 8: Membership of Age

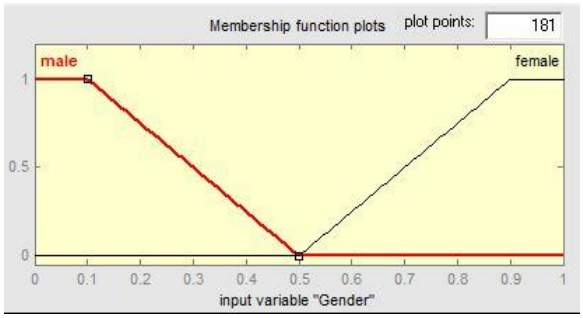

Figure 9: Membership of Gender

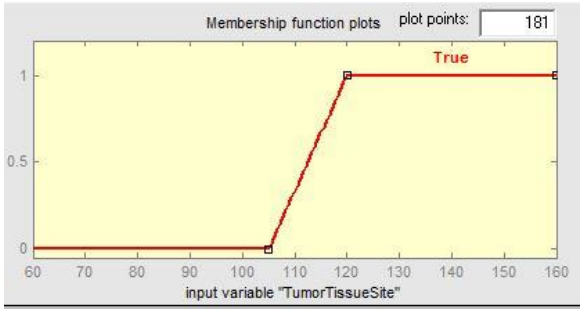

Figure 10: Membership of TumorTissueSite

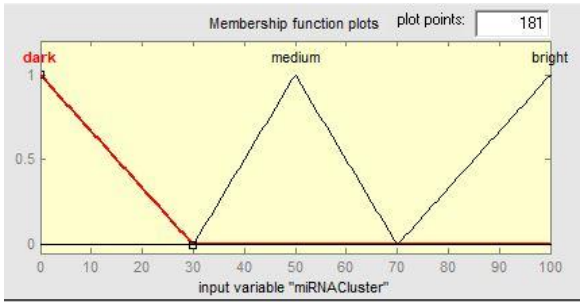

Figure 11: Membership of miRNACluster

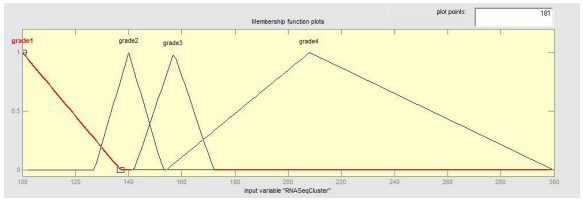

Figure 12: Membership of RNASeqCluster

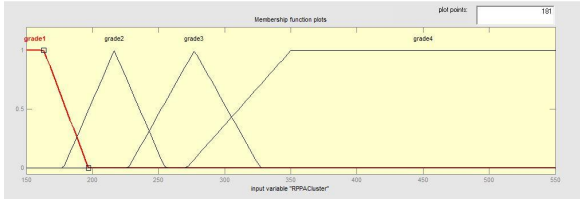

Figure 13: Membership of RPPACluster

B. Fuzzy C-Means Clustering

To come up with the outputs from the Fuzzy C-Means algorithms, there are few technical execution steps that we design to make the code executable and produce the desired output.

First, we import all the libraries below as our initial steps:

```
library("FactoMineR") # to
multivariate Exploratory Data
Analysis
library("factoextra") # Custom
visualizations for clusters
library("tidyverse")  # Data
handling
library("cluster")    # Clustering
algorithms
library(magrittr)
library(dplyr)        # for data
manipulation
library(fclust)       # for fuzzy
clustering
library(ppclust)      # for
clustering result
```

After that, we load our dataset using `read_excel` function since our data is in a .xlsx format:

```
mri_df <- read_excel("data.xlsx")
```

Next, we execute the clustering process using `fcm()` function:

```
res.fcm <- fcm(mri_df, centers=2)
```

We will see the results from the clustering function `fcm()` by calling the variable `res.fcm` into any visualization function on *Result* section.

IV. Results and Discussion

The Brain MRI Segmentation data of the Kaggle platform is extracted from 5 December 2019 till 20 December 2019.

A. Data Preprocessing

The raw Brain MRI Segmentation data extracted from the Kaggle platform initially has 18 attributes as shown below.

Table 2: Data Extracted from Kaggle

Attribute Name	Data Type	Missing Values Percentage (%)	Uniqueness Rate
Patient	Character	0	1
RNASeqCluster	Numeric	16.4	0.0455
MethylationCluster	Numeric	0.909	0.0545
miRNACluster	Numeric	0	0.0364

CNCluster	Numeric	1.82	0.0364
RPPACluster	Numeric	10.9	0.0455
OncosignCluster	Numeric	4.55	0.0364
COCCluster	Numeric	0	0.0273
histological_type	Numeric	0.909	0.0364
neoplasm_histologic_grade	Numeric	0.909	0.0273
tumor_tissue_site	Numeric	0.909	0.0182
laterality	Numeric	0.909	0.0364
tumor_location	Numeric	0.909	0.0545
gender	Numeric	0.909	0.0273
age_at_initial_pathologic	Numeric	0.909	0.418
race	Numeric	1.82	0.0273
ethnicity	Numeric	7.27	0.0273
death01	Numeric	0.909	0.0273

From the table above, the uniqueness rate of every attribute is calculated. Here, we can foresee that some of the data attributes contain missing values despite some of it being important for the modelling. This will cause the model to be working with data that will disrupt the accuracy. Hence, we carried out pre-processing by replacing NA values with mode. Mode value is easy to replace and more convenient as the dataset is randomised. If the attributes containing NA were removed, the dataset would become too small to work with which will affect the accuracy.

As for the data driven approach, Boruta feature selection technique was applied for smaller and more accurate data dimensions. Four attributes has been selected by the Boruta technique and is used for the Fuzzy C-Means Clustering.

B. Mamdani

Based on the Mamdani *Design*, we can create some rules using matlab tools.

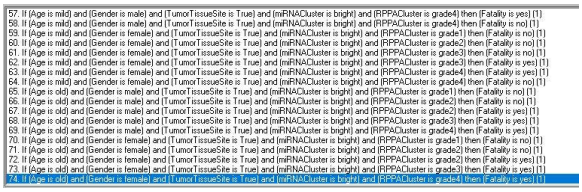

Figure 14: Rules of Mamdani

There are 74 rules with 6 inputs and 1 output that we managed to create from the attributes.

Next, we do the defuzzification to compare between RNASeqCluster and RPPACluster since they have the same fuzzy set representation. This is because we want to see whether this model is good for the dataset or not.

	RNASeqCluster	RPPACluster	Mamdani
grade1	119.1	150	1.98
grade2	142.8	217.9	2.21
grade3	154.9	312.4	2.21
grade4	201.4	401.1	2.21

Figure 15: Comparison between RNASeqCluster and RPPACluster

Based on the comparison, we can see that grade2, grade3 and grade4 have the same value where we know that this knowledge-driven does not give accurate data.

C. Fuzzy C-Means Clustering

To get the outcomes from the fuzzy clustering process using `fcm()` function, we use `summary()` function to output the summarized result of the clustering process:

```
Summary for 'res.fcm'

Number of data objects:  110

Number of clusters:  2

Crisp clustering vector:
  [1] 2 1 1 2 1 1 2 2 2 2 1 2 1 1 2 1 2 1 2 1 2 2 1 1 2 2 2 1 1 2 1 1 1 2 2 2 2 1 1 2 2 2 1 2 1 2 1 2 2
 [52] 1 2 2 1 2 1 2 1 2 2 2 2 2 2 1 2 1 2 2 1 1 2 1 1 2 1 2 2 1 2 1 1 1 2 2 1 1 2 1 1 1 1 2 1 1 2 2 2
[103] 1 2 1 1 2 1 1 2

Initial cluster prototypes:
          MethylationCluster histological_type neoplasm_histologic_grade age_at_initial_pathologic
Cluster 1                  3                 1                         2                        38
Cluster 2                  2                 3                         1                        66

Final cluster prototypes:
          MethylationCluster histological_type neoplasm_histologic_grade age_at_initial_pathologic
Cluster 1           4.061236          2.003713                 1.483982                 33.25288
Cluster 2           3.374887          2.250002                 1.610147                 57.74647

Distance between the final cluster prototypes
          Cluster 1
Cluster 2  600.4836

Difference between the initial and final cluster prototypes
          MethylationCluster histological_type neoplasm_histologic_grade age_at_initial_pathologic
Cluster 1           1.061236          1.0037127                -0.5160181                -4.747123
Cluster 2           1.374887         -0.7499979                 0.6101466                -8.253534

Root Mean Squared Deviations (RMSD): 6.923908
Mean Absolute Deviation (MAD): 36.63331
```

Figure 16: Summary 1

```
Membership degrees matrix (top and bottom 5 rows):
    Cluster 1  Cluster 2
1  0.07145097 0.92854903
2  0.62165742 0.37834258
3  0.96408627 0.03591374
4  0.18602712 0.81397288
5  0.91000737 0.08999263
...
      Cluster 1  Cluster 2
106 0.98381591 0.01618409
107 0.10124083 0.89875917
108 0.98015077 0.01984923
109 0.88933140 0.11066859
110 0.03864992 0.96135008

Descriptive statistics for the membership degrees by clusters
           size      Min       Q1      Mean    Median        Q3       Max
Cluster 1    53 0.6184363 0.8893314 0.9107516 0.9413499 0.9821537 0.9966125
Cluster 2    57 0.5389695 0.8719712 0.9093958 0.9432462 0.9856779 0.9989084

Dunn's Fuzziness Coefficients:
dunn_coeff normalized
 0.8575814  0.7151629

Within cluster sum of squares by cluster:
     1      2
2527.698 2850.246
(between_SS / total_SS =  75.41%)

Available components:
 [1] "u"          "v"          "v0"         "d"          "x"          "cluster"    "csize"      "sumsqrs"
 [9] "k"          "m"          "iter"       "best.start" "func.val"   "comp.time"  "inpargs"    "algorithm"
[17] "call"
```

Figure 17: Summary 2

To visualize the shape of the clusters formed, we then used `fviz_cluster()` function to outputs the cluster figure as shown below:

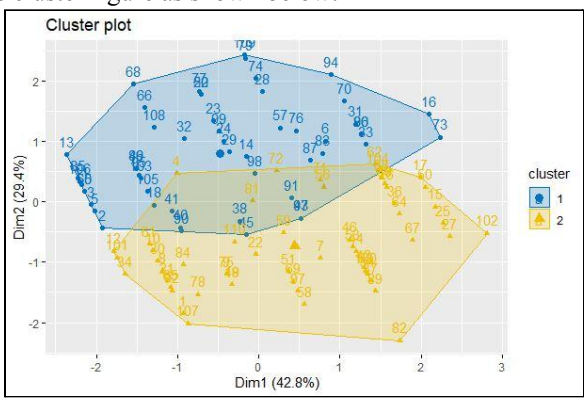

Figure 18: Cluster Plot for FCM

We then used silhouette method to measure of how close each point in one cluster is to points in the neighboring clusters and thus provides a way to assess parameters like number of clusters visually:

```
Average silhouette width per cluster:
[1] 0.6196890 0.6228549
```

Figure 19: Average silhouette width per cluster

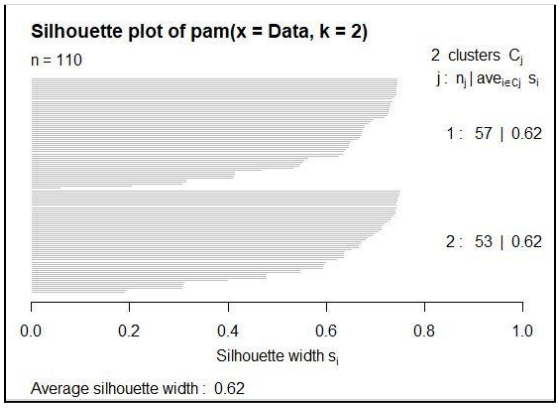

Figure 20: Silhouette Plot for FCM

V. Conclusion

Since our Mamdani does not give the accurate data, hence, we assume that FCM is more suitable for this dataset than Mamdani by giving 75.41% accuracy. Hence the data driven approach is better than knowledge based approach.

Acknowledgement

We are really grateful because finally we managed to complete our mini project within the time given. We would like to express the deepest appreciation to our beloved lecturer of Fuzzy Logic subject, Assoc. Prof. Dr. Choo Yun Huoy for the guidance, support, motivation, and encouragement in finishing this mini project and also for teaching us in this course. This assignment also cannot be completed without the effort and co-operation from our group members. Lastly, we also thank our fellow friends for always supporting us in any situation.

References

[1] Mohamed Nasor, Walid Obaid, "Detection and Localization of Early-Stage Multiple Brain Tumors Using a Hybrid Technique of Patch-Based Processing, k-means Clustering and Object Counting", *International Journal of Biomedical Imaging*, vol. 2020, Article ID 9035096, 9 pages, 2020. https://doi.org/10.1155/2020/9035096

[2] MatlabWorks, "Mamdani and Sugeno Fuzzy Inference Systems". https://www.mathworks.com/help/fuzzy/types-of-fuzzy-inference-systems.html

[3] somya13. (2019, September 17). ML: Fuzzy Clustering. Retrieved December 25, 2019, from https://www.geeksforgeeks.org/ml-fuzzy-clustering/.

[4] Dr.E.N.Sathishkumar. (2015, February 19). Fuzzy c means manual work. Retrieved December 25, 2019, from https://www.slideshare.net/ENSathishkumar/fuzzy-c-means-manual-work.

[5] ppclust. (n.d.). Retrieved December 25, 2019, from https://www.rdocumentation.org/packages/ppclust/versions/0.1.3/topics/fcm.