King Saud University
College of Computer and Information Sciences
Department of Information Technology

IT 326: (Data Mining Project)

1st Semester 1446 H

BRAIN STROKE DATASET

Report

Section #	NAME	ID
56546 Group number #3	***Rose Mady***	***444200107***
	Munira Alsaleem	***444200814***
	Layan Aldbays	***444200653***

Supervised By: ***Hessah Alsaaran***

Tavel of Content

Problem

Recently, the incidence of brain strokes has been on the rise, becoming increasingly common among individuals. This condition can lead to severe health complications, including death. In our project, we will study and analyze patients' data to identify possible factors and risks that contribute to brain strokes. By understanding these risk factors, we aim to help individuals take preventive measures by predicting the likelihood of experiencing a brain stroke.

Data Mining Task

The primary aim of collecting the Brain Stroke Data dataset is to analyze and predict stroke occurrences based on various factors. Our data mining project will focus classification and clustering to predict the occurrence of stroke based on age, gender, hypertension,heart_disease and other attributes in the dataset. helping doctors and researchers to understand the factors that contribute to the risk of stroke as this is crucial for developing these strategies.

Clustering can help identify risk groups by uncovering subgroups with similar risk profiles for stroke without considering the class labels. This approach is valuable for gaining deeper insights into the factors influencing stroke.

Data

Data Source: https://www.kaggle.com/datasets/niranjanank/brain-stroke-data

- Number of objects in original dataset: 4981.
- Number of attributes: 11.
- Class labels: stroke.
- Missing values: there is no missing values.

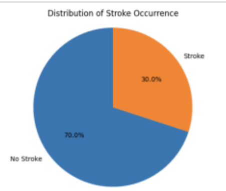
Attributes description:

Attributes name	Data type	Possible Values
gender	Binary (symmetric)	Male (0), Female (1)
Age	Numeric (Ratio)	0.08-82
hypertension	Binary (asymmetric)	1(Yes),0(No)
heart_disease	Binary (asymmetric)	1(Yes),0(No)
ever_married	Binary (asymmetric)	1(Yes),0(No)
work_type	Nominal	0(Private),1(Self-employed),2(Government job),3(Never worked)
Residence	Binary (asymmetric)	0(Urban),1(Rural)
avg_glucose_level (Average glucose level)	Numeric (Ratio)	55.12-271.74
bmi (Body mass index)	Numeric (Ratio)	14-48.9
smoking_status	Nominal	0(formerly smoked),1(never smoked),2(smokes),3(Unkown)
stroke	Binary (Asymmetric)	1(Yes),0(No)

Data distribution:

```
Original class distribution:
 stroke
0    4733
1     248
Name: count, dtype: int64
Balanced sample class distribution:
 stroke
0    560
1    240
```

Sample distribution:



No stroke count: 560
Stroke count: 240

Statistical measures:

- **Gender**:

Gender is a binary variable (0 or 1), The mode is 1, indicating that females (1) are more represented compared to males (0).

- **Age**:

The ages span from 0.08 to 82 years, with a median of 53 years and a mean of 50.4 years. The mode is 78 years. This indicates a wide distribution, reflecting the inclusion of individuals from infancy to old age.

- **Hypertension**:

Hypertension is a binary variable (0 or 1), The mode is 0, indicating that most individuals do not have hypertension.

- **Heart Disease**:

Heart disease is a binary variable, The mode is 0, meaning most individuals do not have heart disease.

- **Ever Married**:

This binary variable captures marital status, The mode is 0, indicating that most individuals have never been married.

- **Work Type**:

Work type is a categorical variable ranging from 0 to 3, The mode is 0, meaning most individuals are classified in the first category of work type.

- **Residence Type**:

Residence type is binary (0 for Urban, 1 for Rural), The mode is 1, indicating that most individuals live in urban areas.

- **Average Glucose Level**:

Glucose levels vary significantly, ranging from 55.34 to 271.74, with a median of 114.13 and a mean of 110. The mode is 66.03, indicating some individuals have significantly higher glucose levels, which might suggest outliers or extreme cases.

- **BMI**:

BMI values range from 14 to 48.9, with a median of 28.50 and a mean of 29.35 The mode is 30.9, suggesting that a significant portion of individuals fall into the overweight category.

- **Smoking Status**:

Smoking status is represented by categories from 0 to 3, The mode is 1, indicating that most individuals fall into the second category of smoking status.

- **Stroke**:

Stroke occurrence is binary, The mode is 0, meaning most individuals have not experienced a stroke.

(Min., 1st Qu., Median, Mean ,3rd Qu.,Max.):
using summary_stats()

```
          gender         age  hypertension  heart_disease  ever_married  \
count  800.000000  800.000000    800.000000     800.000000    800.000000
mean     0.562500   49.071350      0.165000       0.083750      0.293750
std      0.496389   23.450985      0.371413       0.277186      0.455764
min      0.000000    0.080000      0.000000       0.000000      0.000000
25%      0.000000   32.000000      0.000000       0.000000      0.000000
50%      1.000000   53.000000      0.000000       0.000000      0.000000
75%      1.000000   70.000000      0.000000       0.000000      1.000000
max      1.000000   82.000000      1.000000       1.000000      1.000000

        work_type  Residence_type  avg_glucose_level         bmi  \
count  800.000000      800.000000         800.000000  800.000000
mean     0.733750        0.502500         113.240588   28.829250
std      1.046054        0.500307          52.414878    6.734212
min      0.000000        0.000000          55.420000   14.100000
25%      0.000000        0.000000          77.277500   24.100000
50%      0.000000        1.000000          93.025000   28.500000
75%      1.000000        1.000000         124.375000   33.000000
max      3.000000        1.000000         271.740000   48.900000

       smoking_status      stroke
count      800.000000  800.000000
mean         1.515000    0.300000
std          1.079923    0.458544
min          0.000000    0.000000
25%          1.000000    0.000000
50%          1.000000    0.000000
75%          3.000000    1.000000
max          3.000000    1.000000
```

Mode:

```
gender                1.00
age                  80.00
hypertension          0.00
heart_disease         0.00
ever_married          0.00
work_type             0.00
Residence_type        1.00
avg_glucose_level    66.03
bmi                  31.40
smoking_status        1.00
stroke                0.00
Name: 0, dtype: float64
```

Variance:

Variance helps to quantify the dispersion of values. A higher variance indicates a greater spread from the mean, reflecting more variability, while a lower variance suggests values are closer to the mean, indicating less variability.Therefore, our variance results indicate:

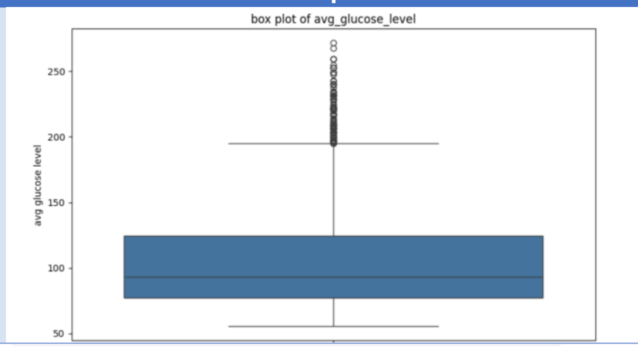
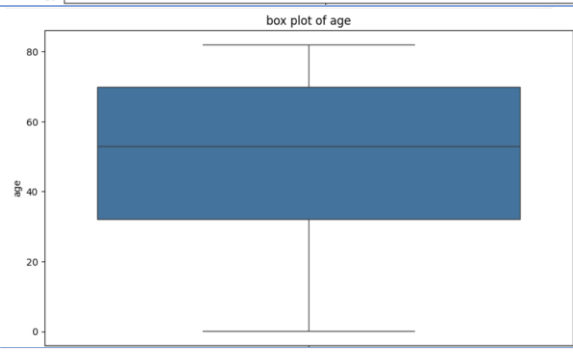
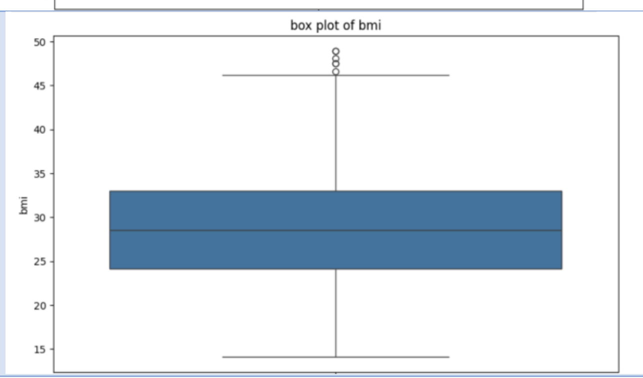
Age, Average Glucose Level, and and BMI: The variance is high with the values(540.97, 2777.9, 41.2) respectively, so the level of dispersion and spread of values is high.

Work Type, Smoking Status: The variance is moderate to high in these columns, having the values(1.06, 1.18) respectively, so the level of dispersion and spread of values is moderate to high.
Hypertension, Heart Disease, Ever Married, Gender ,Stroke: The variance is low in these columns, having the values (0.14, 0.08, 0.19, 0.24, 0.21)respectively, so the level of dispersion and spread of values is low.

```
gender                  0.246402
age                   549.948691
hypertension            0.137947
heart_disease           0.076832
ever_married            0.207721
work_type               1.094229
Residence_type          0.250307
avg_glucose_level    2747.319482
bmi                    45.349606
smoking_status          1.166233
stroke                  0.210263
dtype: float64
```

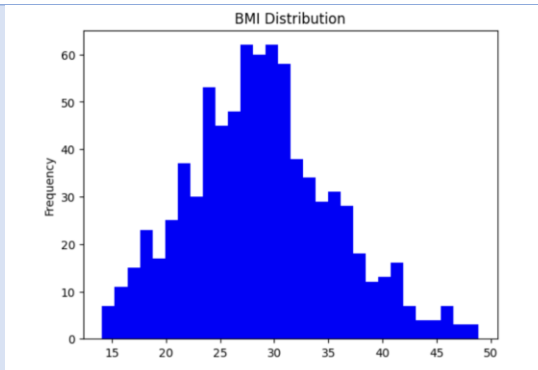
Graphical Reprisentations:

Box plots:
The box plot is a standardized way of displaying the distribution of data based on a five number summary

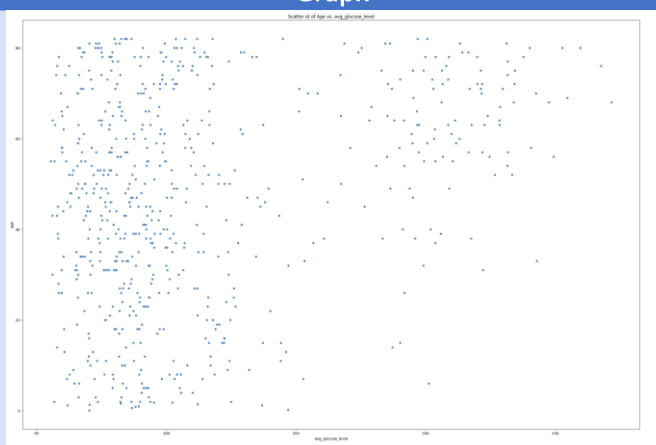
Graph	Description
	The attribute avg glucose level has a wide range of data with a median of 90 and outliers in the range 180-260
	The box plot of age is a standardized way of displaying the distribution of data based on a five number summary, as we see the attribute age has a wide range of data with a median of 52 and no outliers
	The box plot is a standardized way of displaying the distribution of data based on a five number summary, as we see the attribute BMI has a wide range of data with a median of 28 which indicates that more than 50% of the sample are obese, and there is some outliers between 45-50

Histogram:

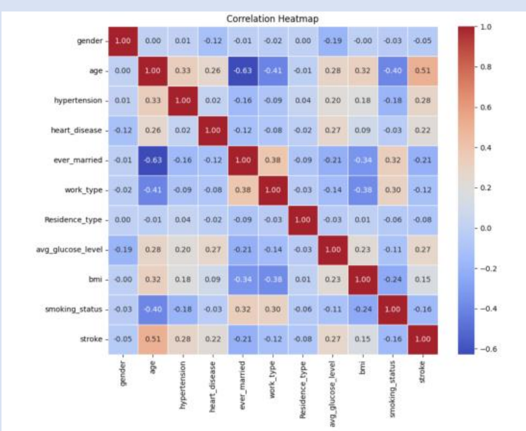
Graph	Description

	The histogram shows that most individuals have a BMI between 25 and 35, indicating that the majority fall within the normal to overweight range. The peak BMI is around 25-30 which has the highest frequency with 60. There are relatively few cases of underweight (BMI below 20) or extreme obesity (BMI above 40), suggesting that outliers on both ends are rare.

Scatter Plot:

Graph	Description
	The majority of individuals have an average glucose level within the range of 60 to 125. A significant portion of those with glucose levels exceeding 125 are older adults, typically aged 50 and above.

Heat Map:

Graph	Description
	1- "Age" has moderate positive correlation with stroke. As age increases, the risk of stroke tends to increase. 2- "Heart Disease", "Hypertension", "Average Glucose Level" have weak positive correlation, which means individuals having heart diseases or hypertension or increase in average glucose level show slight increase in stroke risk. 3- "Ever Married" has weak negative correlation with stroke, which means that being married is linked to a slightly lower risk of stroke 4- "BMI", "Smoking Status", "Residence Type", "Gender" have very weak correlation with stroke. they have minimal or no impact on stroke risk.

Data Processing

Data Cleaning:

Duplicates:

```
Number of duplicate rows: 129
```

Since our dataset doesn’t have a specific column for patient identification, such as a Patient ID or Social Insurance Number (SIN), we cannot assume that any rows are duplicated, as similar symptoms could appear for multiple cases. Therefore, we have not removed any duplicates.

Outliers:

Detecting outliers:

We identified outliers in our data by separately calculating them for each attribute type (binary, nominal, numeric), we calculated the outliers of the binaries and nominal integers to ensure that there is no mis entering values out of range, and to avoid mistakenly considering values as outliers based on comparison to the entire dataset. Once we identified these outliers, we took appropriate measures to handle them.

```
Outlier Counts:
hypertension: 132 rows with outliers
avg_glucose_level: 114 rows with outliers
bmi: 6 rows with outliers
Total Rows with Outliers: 252
```

```
stroke: 0 mis-entries found
gender: 0 mis-entries found
heart_disease: 0 mis-entries found
ever_married: 0 mis-entries found
Residence_type: 0 mis-entries found
```

```
work_type: 0 mis-entries found
smoking_status: 0 mis-entries found
```

After analyzing outliers for numeric values we found that our numeric attributes contains exactly 252 outlier value overall, 132 raw of them from hypertension, 114 from the average glucose level, and 6 of them from the BMI column. Since the rows are a lot we preferred not to drop the raw, and other than that we handled the outliers by smoothing them using the capping out process(Winsorization), we preferred it most because it reduce the value to the nearest minimum/ maximum value that is not outliered.

After handling outliers:

```
Outlier Counts:
hypertension: 0 rows with outliers
avg_glucose_level: 0 rows with outliers
bmi: 0 rows with outliers
```

Transformation:

Encoding:

The columns that needed to be transformed were the nominal ones. Since attributes like "work_type" and "smoking status" were already transformed from nominal to integer values, we didn't need to perform this step as part of our data preparation process.

Normalization:

We preformed the normalization process to the attributes having continuous values that has big difference in the range to be in shorter limited range (between 0 and 1) rather than large numbers.

```
Output after Max/Min Scaling Normalization:

     gender   age  hypertension  heart_disease  ever_married  work_type  \
0         0  80.0             0              0             0          0
1         1  55.0             0              1             0          0
2         0  79.0             0              1             0          0
3         0  75.0             0              0             0          0
4         0  82.0             0              0             0          2
..      ...   ...           ...            ...           ...        ...
795       0  32.0             0              0             0          0
796       1  66.0             0              0             0          2
797       0  47.0             0              0             0          0
798       1   5.0             0              0             1          3
799       0  20.0             0              0             1          0

     Residence_type  avg_glucose_level       bmi  smoking_status  stroke
0                 1           1.000000  0.545736               2       1
1                 0           1.000000  0.803101               2       1
2                 1           0.534093  0.263566               0       1
3                 0           1.000000  0.362791               2       1
4                 0           1.000000  0.462016               0       1
..              ...                ...       ...             ...     ...
795               1           0.115185  0.548837               1       0
796               0           0.262319  0.279070               1       0
797               1           0.542760  0.440310               2       0
798               0           0.269697  0.155039               3       0
799               1           0.495984  0.409302               1       0

[800 rows x 11 columns]
```

Discretization:

We applied discretization to the "age" attribute, as it is commonly divided into three categories based on typical medical classifications:

Child: [0, 17], Adult: [18, 64], Older Adult: [65, 100]

***the last partition upper limit is 100 because the maximum age in our dataset is 82.*

```
Output after Age Descritization:

     gender          age  hypertension  heart_disease  ever_married  \
0         0  Older Adult             0              0             0
1         1        Adult             0              1             0
2         0  Older Adult             0              1             0
3         0  Older Adult             0              0             0
4         0  Older Adult             0              0             0
..      ...          ...           ...            ...           ...
795       0        Adult             0              0             0
796       1  Older Adult             0              0             0
797       0        Adult             0              0             0
798       1        Child             0              0             1
799       0        Adult             0              0             1

     work_type  Residence_type  avg_glucose_level       bmi  smoking_status  \
0            0               1           1.000000  0.545736               2
1            0               0           1.000000  0.803101               2
2            0               1           0.534093  0.263566               0
3            0               0           1.000000  0.362791               2
4            2               0           1.000000  0.462016               0
..         ...             ...                ...       ...             ...
795          0               1           0.115185  0.548837               1
796          2               0           0.262319  0.279070               1
797          0               1           0.542760  0.440310               2
798          3               0           0.269697  0.155039               3
799          0               1           0.495984  0.409302               1

     stroke
0         1
1         1
2         1
3         1
4         1
..      ...
795       0
796       0
797       0
798       0
799       0

[800 rows x 11 columns]
```

Feature Selection:

We opted for the filter selection method due to its computational efficiency and speed, making it ideal for our dataset. This approach allows us to quickly identify the top 5 most important features that are strongly correlated with the occurrence of strokes. By focusing on these high-correlation features, we can streamline the model building process while retaining the variables most likely to impact stroke prediction.

To prevent biased correlations, we removed the 'stroke' column before analyzing with other features, because it will always give the highest correlation.

and as shown, based on the selection process, the top 5 attributes selected with highest correlation are(age, hypertension, heart_disease, ever_married, and avg_glucose_level)

```
The highest correlation is 0.6771 between ('age', 'ever_married')
```

```
Selected Features: Index(['age', 'heart_disease', 'ever_married', 'avg_glucose_level',
       'smoking_status'],
```

Data Mining Techniques

We utilized both supervised and unsupervised learning methods on our data through the use of classification and clustering techniques.

Classification:

We employed a supervised learning approach to classify individuals as having a brain stroke or not. To achieve this, we divided our dataset into training and testing subsets. The model was then trained on the training subset and evaluated on the testing subset using metrics such as accuracy, sensitivity, specificity, and precision.

To visualize and interpret the decision-making process, we utilized a decision tree implemented using “scikit-learn” library in Python. We used the tree because it is easy to interpret and gives simplify presents the resultant decision, with each leaf node indicating whether an individual is likely to have a brain stroke based on their attributes, including gender, age, marital status, glucose level, heart disease history, hypertension, smoking status, BMI, residence, and work type.

To optimize the model's performance, we experimented with two attribute selection measures (Entropy and Gini Index) and three different data partitions (70/30, 80/20, and 60/40). By comparing the performance results across these combinations, we identified the optimal attribute selection measure and data partition for our specific dataset.

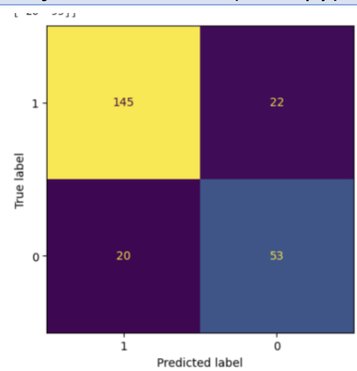
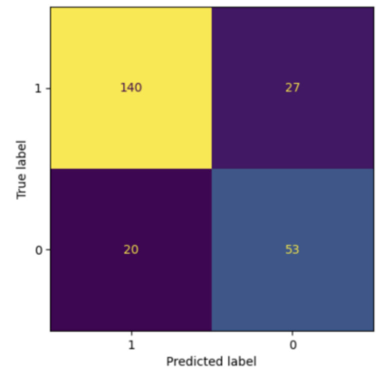
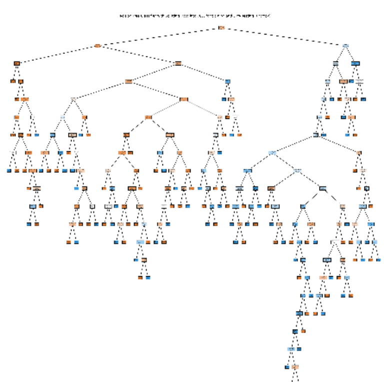
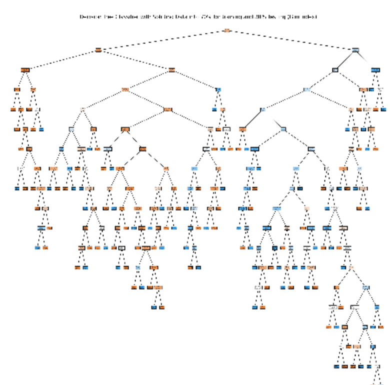
Clustering:

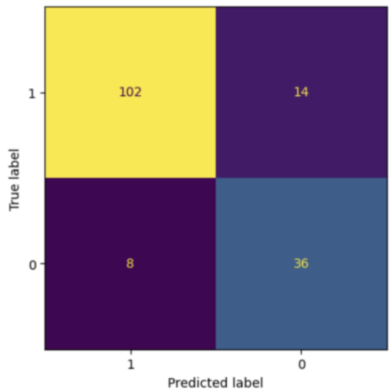
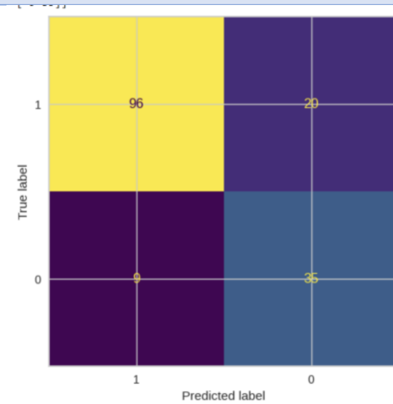
We utilized an unsupervised learning approach, specifically the “K-means” clustering algorithm, to group similar data points. This algorithm iteratively assigns each data point to the nearest cluster centroid, refining the centroids with each iteration. We choose “K-means” because of its popularity and its effectiveness even with large datasets.

To implement “K-means”, we employed the “KMeans” class from the Python library “scikit-learn”. Given the unsupervised nature of clustering, we excluded the class label "stroke" and considered all other attributes in our dataset.

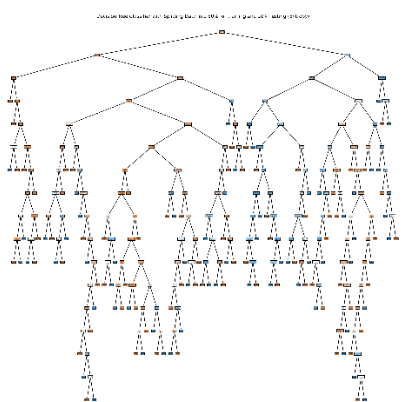
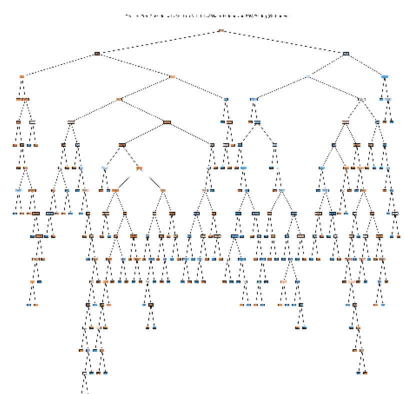
To evaluate the quality of the clusters, we computed the average silhouette score for each cluster. Additionally, we employed the Within-Cluster Sum of Squares (WSS) method to compare the sizes of the three clusters (2, 3, 7) and determine the optimal number of clusters, balancing cluster separation and compactness.

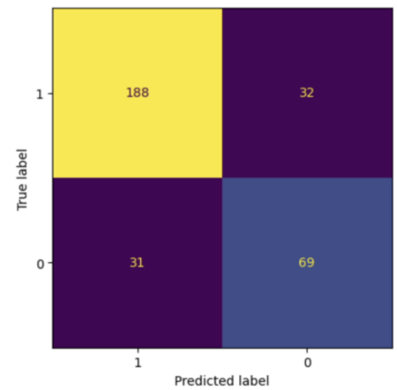
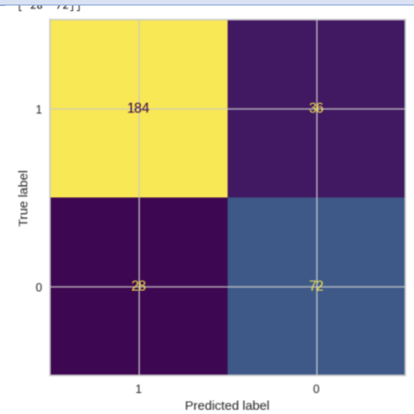
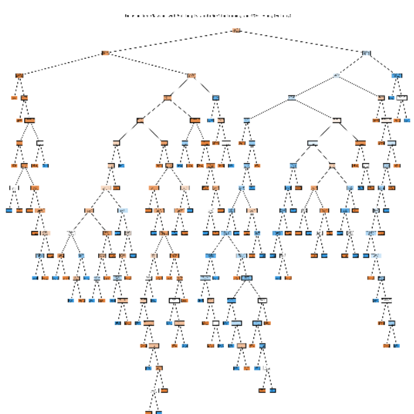
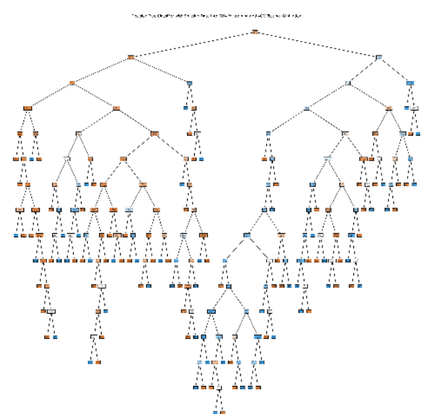
Evaluation and comparison

<table>
<tr><th colspan="3">Splitting data into 70% for Training and 30% for Testing:</th></tr>
<tr><th></th><th>Information Gain (Entropy):</th><th>Gini Index:</th></tr>
<tr><td rowspan="2">Model Evaluation:</td><td>
</td><td>
</td></tr>
<tr><td>Accuracy: 0.825
Sensitivity: 0.726027397260274
Specificity: 0.8682634730538922
Precision: 0.7066666666666667</td><td>Accuracy: 0.8041666666666667
Sensitivity: 0.726027397260274
Specificity: 0.8383233532934131
Precision: 0.6625</td></tr>
<tr><td>Decision Tree:</td><td></td><td></td></tr>
</table>

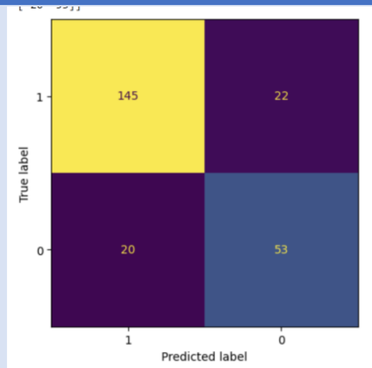
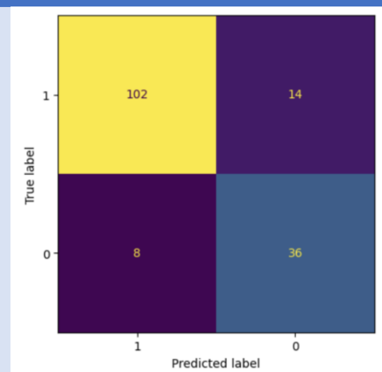
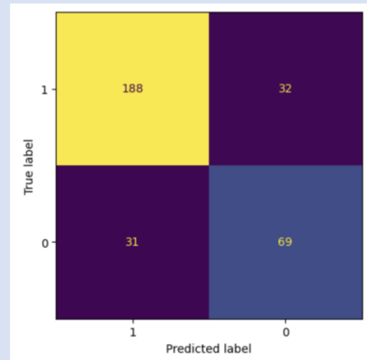
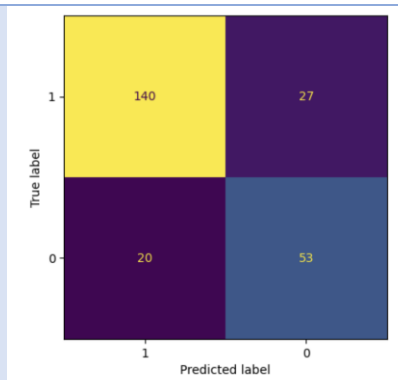
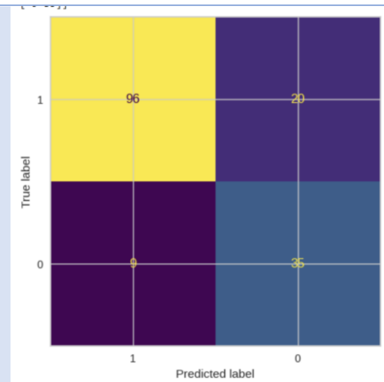
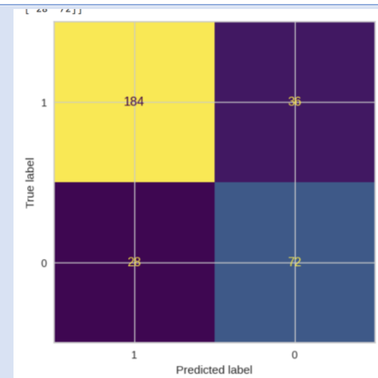
Splitting data into 80% for Training and 20% for Testing:		
	Information Gain (Entropy):	***Gini Index:***
Model Evaluation:		

	`Accuracy: 0.8625`
	`Sensitivity: 0.8181818181818182`
	`Specificity: 0.8793103448275862`
	`Precision: 0.72`
		`Accuracy: 0.81875`
		`Sensitivity: 0.7954545454545454`
		`Specificity: 0.8275862068965517`
		`Precision: 0.6363636363636364`

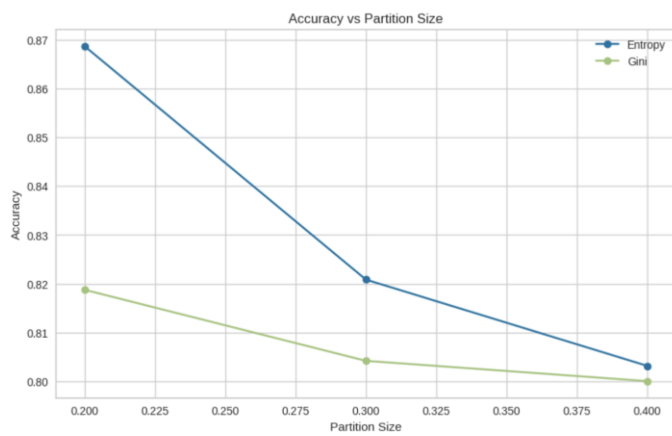
Decision Tree:		

Splitting data into 60% for Training and 40% for Testing:		
	Information Gain (Entropy):	***Gini Index:***
Model Evaluation:		
	Accuracy: 0.803125 Sensitivity: 0.69 Specificity: 0.8545454545454545 Precision: 0.6831683168316832	Accuracy: 0.8 Sensitivity: 0.72 Specificity: 0.8363636363636363 Precision: 0.6666666666666666
Decision Tree:		

Classification Evaluation:

		70% training set 30% test set	80% training set 20% test set	60% training set 40% test set
Entropy				
	Accuracy	83%	80%	80%
Gini Index				
	Accuracy	80%	81%	80%

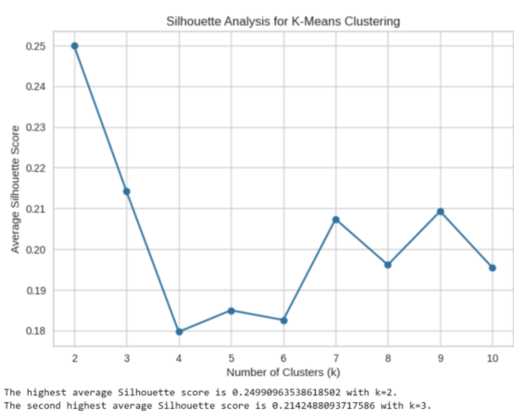
Findings:

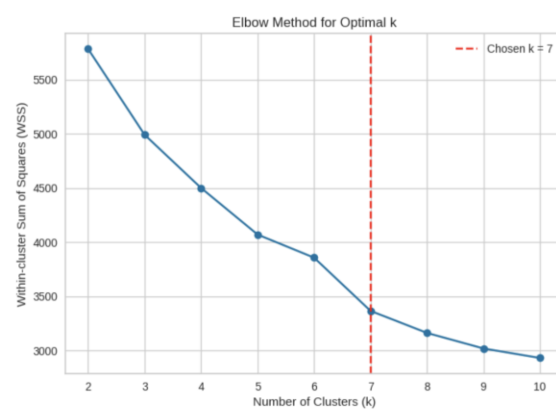


Information Gain (Entropy) is superior in overall accuracy, specificity, and precision, making it generally more effective across splits. Gini Index provides competitive sensitivity and precision in certain configurations but slightly trails in accuracy and specificity. If accuracy and specificity are the primary goals, Information Gain is preferable. However, Gini Index might be useful when focusing on sensitivity with a slightly larger training set.

Clustering Evaluation:

Number of cluster chosen:





Cluster Figures:

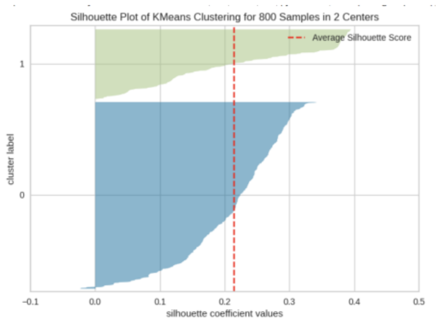


Figure 1 K=2

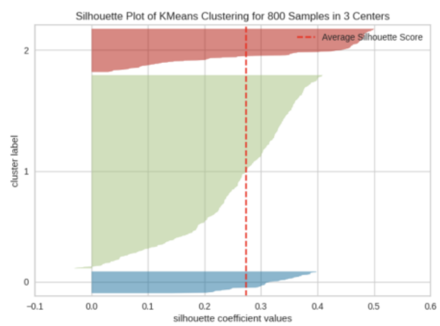


Figure 2 K=3

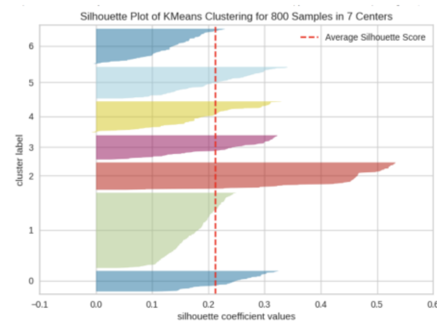


Figure 3 K=7

Number of clusters	K=2	K=3	K=7
Average Silhouette width	0.250	0.214	0.207
Total within-cluster sum of square	5781.88	4990.2	3365.38

Findings:

We’ve decided that K=2 is the best choice for our clustering model based on the metrics we’ve analyzed (WSS, Average Silhouette Score, Visualization of K-mean). This choice is because K=2 gives the highest silhouette width, and also K=2 has the highest value of WSS compared to the WSS values for K=3 and K=7.
Additionally, having a silhouette plot of K-Means clustering of 800 samples with 2 centers was one of the most important criteria for choosing K=2 as the best K, indicating that it creates distinct and cohesive clusters.

Findings

We begun with selecting the dataset under interest that meets our goal which was studying the cases of brain strokes.

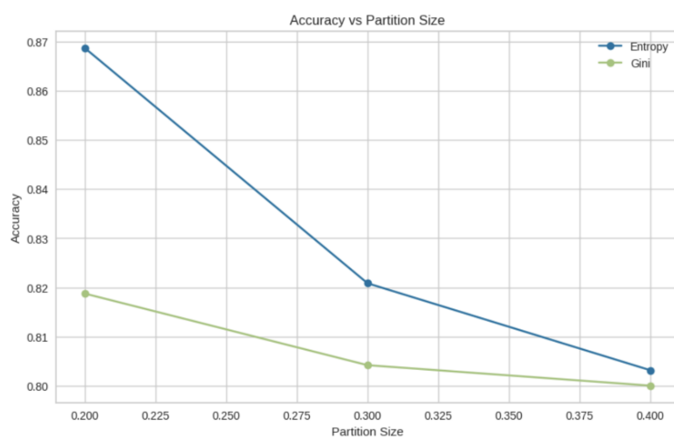
To address the significant imbalance in our dataset, where stroke cases were significantly fewer than non-stroke cases, we employed a sampling technique to create a more balanced subset. This balanced subset aimed to include approximately 30% stroke cases and 70% non-stroke cases.

To gain insights into the data, we utilized various visualization techniques. Box plots were employed to identify outliers and understand data distribution. Scatter plots to visualize relationships between variables, such as the association between age and glucose levels. Histograms were used to visualize the distribution of numerical variables. Finally, heatmaps were employed to explore correlations between different features within the dataset.

Consequently, we began the data processing phase by smoothing outliers. However, as our dataset was already encoded, this step was not necessary. Additionally, we retained duplicate records as we lacked unique identifiers like patient ID or SIN to distinguish individual data duplications. Furthermore, we implemented data transformation, including normalization and discretization. Then we used the filter selection measure to quickly identify the top most important features that are strongly correlated with the occurrence of strokes.

We conducted data mining tasks, including classification and partitioning. For classification, we employed decision trees using the Gini index and information gain metrics. By experimenting with different training and testing set sizes, we optimized model construction and evaluation after comparing results.

Metric	Best Result/ split (Information Gain)	Best Result/ split (Gini Index)
Accuracy	83% using (70% Train, 30% Test) split	81% using (80% Train, 20% Test) split
Sensitivity	80% using (80% Train, 20% Test) split	79% using (80% Train, 20% Test) split
Specificity	87% using (70% Train, 30% Test)	84% using (60% Train, 40% Test) split
Precision	72% using (80% Train, 20% Test)	67% using (60% Train, 40% Test) split



- **Best Split for Information Gain:** The 70% Training, 30% Testing split achieves the highest accuracy and stable specificity, offering a balanced trade-off.
- **Best Split for Gini Index:** The 80% Training, 20% Testing split performs better and having the best sensitivity.

So, Information Gain yielded the highest percentage of all metrics, the highest accuracy when using the 70% Training 30% Testing split. This split also demonstrated strong performance in terms of specificity, while maintaining good sensitivity and precision. Figure (4) shows the decision tree for the best split.

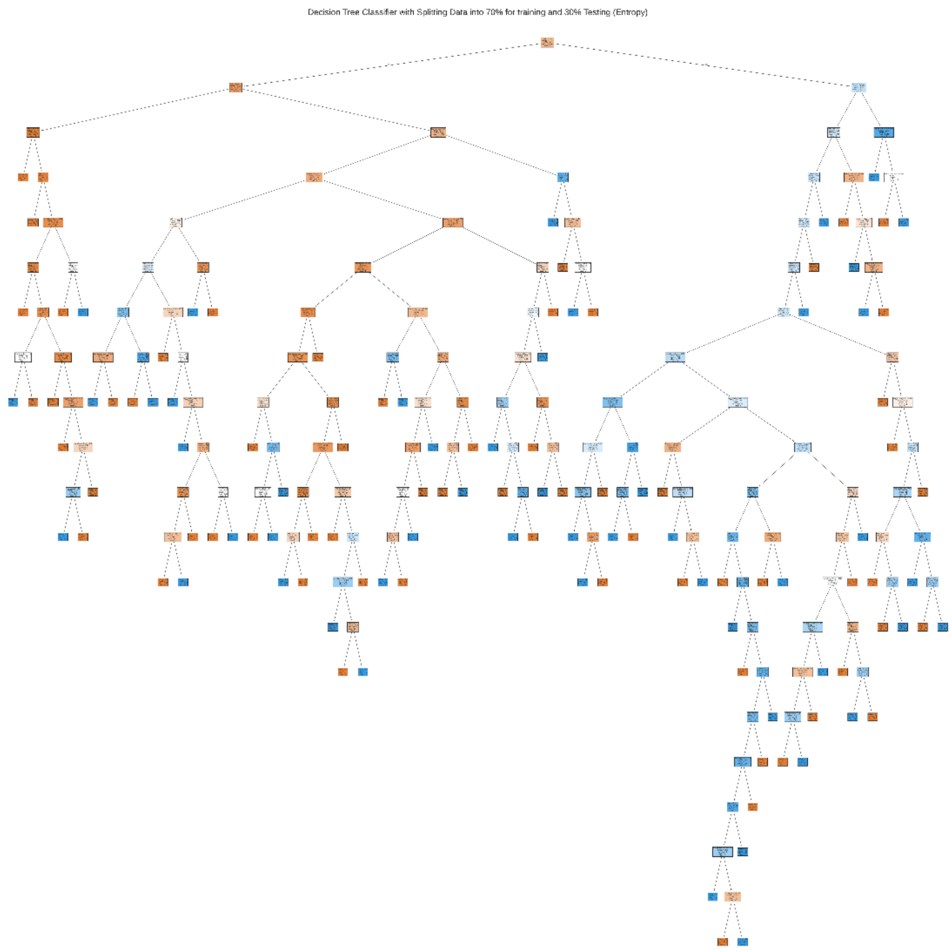


Figure 4 30/70 split decision tree structure

Figure(4) illustrates a decision tree model for classification, using Entropy as the splitting criterion. The dataset is split into 70% for training and 30% for testing. The tree begins with “age” as the root node, which plays a critical role in the initial classification.
In this tree, the condition on the age node is evaluated as follows: if age <= 15, the condition is false, and the model proceeds to check the next node, “ever married”. This node splits the data further based on marital status, directing individuals along different paths depending on whether they are married or not.
However, if age > 15, the condition on the age node is true. In this case, the model moves to the next decision node, “BMI”, where the classification is further refined based on the individual’s body mass index.
After these initial decisions, the model continues branching down, using additional features in a specific order to make further splits at each level. These nodes might include attributes such as average glucose level, hypertension status, heart disease, work type, residence type, and smoking status. Each node along the path serves as a decision checkpoint, progressively narrowing down the classification.
Each path from the root to a leaf node represents a unique sequence of decisions, leading to a specific classification outcome at the end. With multiple levels and branches, the tree captures data complexity by focusing on essential features at each split to create increasingly homogeneous groups in each branch. This structured decision-making process results in terminal nodes (leaves) that consistently represent the predicted class for each subset, offering insight into how various features influence the classification.
The Entropy-based tree is generally simpler than the Gini-based tree, with fewer branches, making it more streamlined and interpretable.

For clustering we used K-means algorithm with 3 different K to find the optinal number of clusters, then we calculated the average silhouette width for each K and total within-cluster sum of square:

	K=2	K=3	K=7
WSS	5781.87	4990.23	3365.38
Average Silhouette Score	0.2499	0.2142	0.2073

According to results, we've decided that K=2 is the best choice for our clustering model based on the metrics we've analyzed (WSS, Average Silhouette Score, Visualization of K-mean). This choice is because K=2 gives the highest silhouette width, and also K=2 has the highest value of WSS compared to the WSS values for K=3 and K=7.
Additionally, having a silhouette plot of K-Means clustering of 800 samples with 2 centers was one of the most important criteria for choosing K=2 as the best K, indicating that it creates distinct and cohesive clusters.

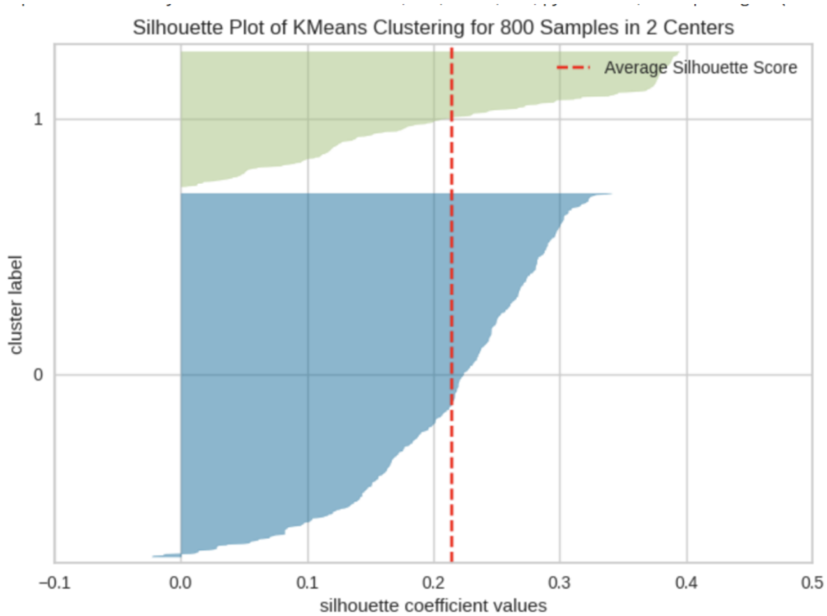


Figure 5 Silhouette plot for K=2

Figure (5) represents silhouette plot for clustering 800 samples into 2 centers using KMeans, most samples have positive silhouette scores, indicating that the samples are generally well-matched to their respective clusters and are reasonably distant from neighboring clusters.
However, Cluster 0 (blue region) has a broader range of silhouette coefficients, with many values close to 0 or even slightly negative. Negative values indicate samples that may be assigned to the wrong cluster, suggesting that some samples are closer to the neighboring cluster.
Cluster 1 (green region) has mostly positive silhouette coefficients, and samples in this cluster seem to be better clustered with relatively higher silhouette values.

Finally, both models are helpful for predicting whether a person can have a brain stroke, and helped us to reach our goal which is helping to have an impact on promoting public health. but since our data contains a class label "stroke" This makes Supervised Learning models(classification) more accurate and suitable to apply than unsupervised learning model(clustering), as the expected output is known beforehand this way we makes use of the class label attribute.

References

1. Niranjanan K, Brain Stroke Data dataset, Kaggle, Brain Stroke Data
2. Labs and Lecture Slides, College of Computer Science, Department of Information Technology, King Saud University.
3. Jiawei Han, Jian Pei and Hanghang Tong, "Data Mining: Concepts and Techniques", Morgan Kaufmann, *4th Edition, 2022*