



74557King Saud University
College of Computer and Information Sciences
Information Technology department

IT 326: Data Mining
Course Project

Score Cast

Project final Report

Group#
LAB Day-Time:
Wednesday-1
Group members:

Group#:	9	
Section#:	71162	
Group Members	Name	ID
	Sarah Aloqiel	444203016
	Aljwharah Alhowidy	443201015
	Rana Alobathani	444200855
	Nouf AlMansour	444200525
	Shahad Alumtairi	444200935

Problem Statement

The FIFA World Cup stands as one of the most celebrated and globally followed football tournaments, showcasing intense competition and exceptional performances from teams and players. The aim of this project was to analyze a comprehensive dataset of World Cup matches to evaluate team performance using key metrics such as match outcomes.

Our objectives included:

1. **Performance Evaluation:** Conducting a detailed analysis of teams based on historical data.
2. **Predictive Modeling:** Developing models capable of forecasting match outcomes.
3. **Trend Discovery:** Identifying patterns in match results, to gain insights into game dynamics and competitive behaviors.

To achieve these goals, we employed machine learning techniques such as clustering and classification, utilizing methods learned throughout the course. By mining the historical data provided by this dataset, the analysis aimed to offer valuable insights to sports analysts, researchers, and football enthusiasts.

This project focuses on leveraging data science methodologies to uncover trends and predictive indicators within World Cup dynamics, contributing to a better understanding of factors influencing match outcomes and team strategies.

Data Mining Tasks

To achieve the project objectives of analyzing World Cup matches and predicting outcomes, the following data mining tasks were undertaken:

1. Data Preprocessing

- **Cleaning:** Addressed missing values and inconsistencies in the dataset to ensure data integrity.
- **Normalization:** Scaled numerical features, such as scores, to improve the performance of machine learning algorithms.
- **Categorical Encoding:** Converted categorical variables, such as team names and match outcomes, into numerical formats using techniques like one-hot encoding and label encoding.

2. Understanding the Data through Graph Representations:

- Analyzed the distribution of key variables.
- Visualized trends in team performance across tournaments using histograms, box plots, and time-series plots.

3. Clustering

- Used clustering algorithms such as **K-Means** to group teams based on performance metrics, such as goals outcomes.
- Analyzed clusters to identify teams with similar playing styles or performance levels.

4. Classification

- Implemented classification algorithms to predict match outcomes (win, lose, or draw).
- Evaluated the importance of features such as outcomes and scores.

5. Predictive Modeling

- Developed models to predict key outcomes, including:
 - **Match Results:** Predicted the likelihood of a team winning, losing, or drawing a match.

6. Trend and Pattern Discovery

- Discovered trends over time, such as how some teams are favored to win more frequently.

7. Evaluation and Validation

- Evaluated models using metrics like accuracy, precision and recall to assess predictive performance.
- Validated findings using a holdout dataset and cross-validation techniques to ensure reliability and robustness of results.

By completing these tasks, the project uncovered actionable insights into World Cup match dynamics and provided reliable predictions for future matches and player performances.

Data Analysis

1. Attribute Distribution

Year:

- **Range:** 1930 to 2018
- **Mean:** 1986.92
- **Median:** 1990
- **Observation:** Matches are evenly distributed over decades, with most occurring between 1970 and 2006.

Home Score:

- **Range:** 0 to 10 goals
- **Mean:** 1.57 goals
- **Median:** 1 goal
- **Observation:** Most matches result in home teams scoring 2 or fewer goals.

Away Score:

- **Range:** 0 to 8 goals
- **Mean:** 1.26 goals
- **Median:** 1 goal
- **Observation:** Away teams generally score fewer goals than home teams, with most scoring 2 or fewer goals.

2. Statistical Measures for Numeric Columns

Metric	Year	Home Score	Away Score
Min	1930.00	0.00	0.00
1st Quartile	1970.00	0.00	0.00
Median	1990.00	1.00	1.00
Mean	1986.92	1.57	1.26
3rd Quartile	2006.00	2.00	2.00
			8.00
Max	2018.00	10.00	

3. Variance Analysis

Attribute	Variance	Interpretation
Year	535.94	High variance due to the wide range (1930–2018).
Home Score	2.22	Low variance indicating little fluctuation in home team scores.
Away Score	1.73	Low variance indicating little fluctuation in away team scores.

4. Handling Missing Values

The dataset includes missing values in the following columns:

- win_conditions:** 838 missing values
- winning_team:** 169 missing values
- losing_team:** 169 missing values

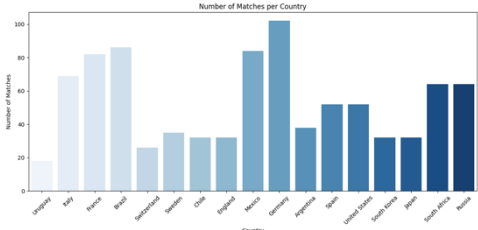
Strategy to Handle Missing Values:

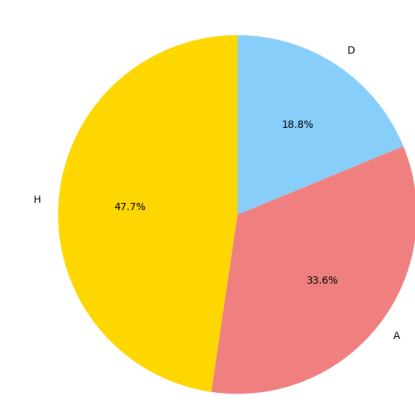

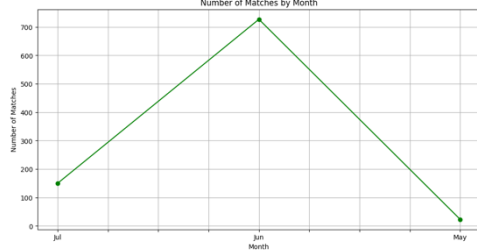
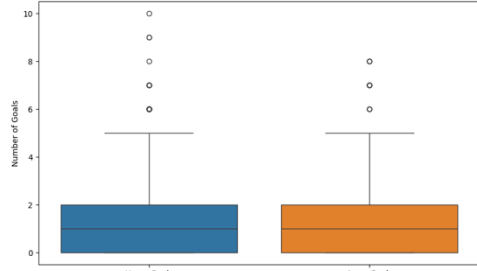
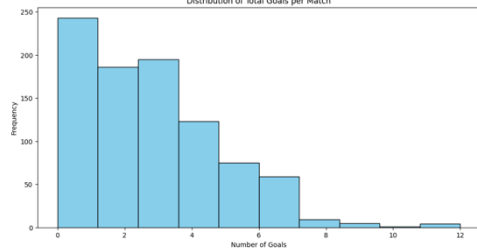
- win_conditions:** Filled with "no win conditions."
- winning_team** and **losing_team:** Filled with "Draw match," since missing values indicate a tie.

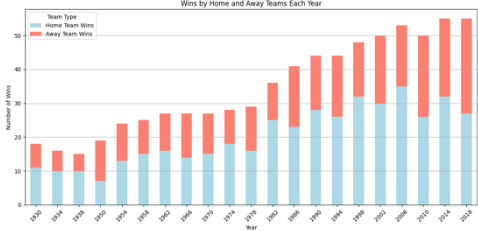
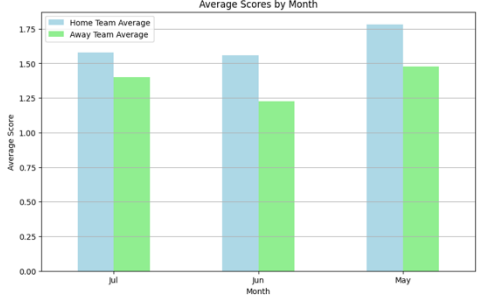
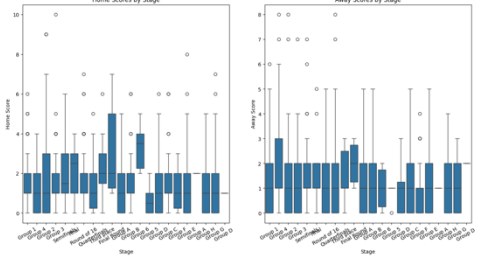
This approach ensures data completeness and avoids discarding useful information.

5. Graphical Analysis

The following table summarizes the visualizations created to analyze the FIFA World Cup dataset. Each chart provides unique insights into team performance, scoring trends, and match outcomes, helping uncover key patterns and relationships in the data.

Title	Description	Graph Illustration
Bar Chart: Match Counts by Country	Germany has the highest participation, followed by Brazil, Mexico, and France. Uruguay and Switzerland have the fewest matches.	

Pie Chart: Match Outcomes Distribution	Wins: 47.7% (Gold) - Losses: 33.6% (Light Coral) - Draws: 18.8% (Light Sky Blue) Nearly half of all matches result in a win, with draws being the least common outcome.	<p>Outcome Percentages for Each Team</p>  <table><tr><th>Outcome</th><th>Percentage</th></tr><tr><td>Wins (H)</td><td>47.7%</td></tr><tr><td>Losses (A)</td><td>33.6%</td></tr><tr><td>Draws (D)</td><td>18.8%</td></tr></table>	Outcome	Percentage	Wins (H)	47.7%	Losses (A)	33.6%	Draws (D)	18.8%																																											
Outcome	Percentage																																																				
Wins (H)	47.7%																																																				
Losses (A)	33.6%																																																				
Draws (D)	18.8%																																																				
Line Chart: Total Goals by Year	Demonstrates fluctuations, with peaks in the 1950s, 1980s, and 2000s.	<p>Development of Goals Scored Each Year</p>  <table><tr><th>Year</th><th>Number of Goals</th></tr><tr><td>1940</td><td>70</td></tr><tr><td>1945</td><td>85</td></tr><tr><td>1950</td><td>90</td></tr><tr><td>1955</td><td>140</td></tr><tr><td>1960</td><td>125</td></tr><tr><td>1965</td><td>90</td></tr><tr><td>1970</td><td>95</td></tr><tr><td>1975</td><td>100</td></tr><tr><td>1980</td><td>105</td></tr><tr><td>1985</td><td>145</td></tr><tr><td>1990</td><td>135</td></tr><tr><td>1995</td><td>115</td></tr><tr><td>2000</td><td>140</td></tr><tr><td>2005</td><td>170</td></tr><tr><td>2010</td><td>160</td></tr><tr><td>2015</td><td>145</td></tr><tr><td>2020</td><td>170</td></tr></table>	Year	Number of Goals	1940	70	1945	85	1950	90	1955	140	1960	125	1965	90	1970	95	1975	100	1980	105	1985	145	1990	135	1995	115	2000	140	2005	170	2010	160	2015	145	2020	170															
Year	Number of Goals																																																				
1940	70																																																				
1945	85																																																				
1950	90																																																				
1955	140																																																				
1960	125																																																				
1965	90																																																				
1970	95																																																				
1975	100																																																				
1980	105																																																				
1985	145																																																				
1990	135																																																				
1995	115																																																				
2000	140																																																				
2005	170																																																				
2010	160																																																				
2015	145																																																				
2020	170																																																				
Box Plot: Home vs. Away Scores by Country	Illustrates variations in scoring tendencies, with significant differences between home and away games for some countries.	<p>Distribution of Home and Away Scores by Country</p>  <table><tr><th>Country</th><th>Home Score (Median)</th><th>Away Score (Median)</th></tr><tr><td>England</td><td>1.5</td><td>1.0</td></tr><tr><td>Italy</td><td>1.0</td><td>1.0</td></tr><tr><td>France</td><td>1.0</td><td>1.0</td></tr><tr><td>Spain</td><td>1.0</td><td>1.0</td></tr><tr><td>South America</td><td>2.0</td><td>1.5</td></tr><tr><td>Sweden</td><td>1.5</td><td>1.0</td></tr><tr><td>China</td><td>1.0</td><td>1.0</td></tr><tr><td>Germany</td><td>1.0</td><td>1.0</td></tr><tr><td>Mexico</td><td>1.0</td><td>1.0</td></tr><tr><td>Indonesia</td><td>1.0</td><td>1.0</td></tr><tr><td>Japan</td><td>1.0</td><td>1.0</td></tr><tr><td>United States</td><td>1.0</td><td>1.0</td></tr><tr><td>South Korea</td><td>1.0</td><td>1.0</td></tr><tr><td>Japan</td><td>1.0</td><td>1.0</td></tr><tr><td>France-Korea</td><td>1.0</td><td>1.0</td></tr><tr><td>Sweden</td><td>1.0</td><td>1.0</td></tr></table>	Country	Home Score (Median)	Away Score (Median)	England	1.5	1.0	Italy	1.0	1.0	France	1.0	1.0	Spain	1.0	1.0	South America	2.0	1.5	Sweden	1.5	1.0	China	1.0	1.0	Germany	1.0	1.0	Mexico	1.0	1.0	Indonesia	1.0	1.0	Japan	1.0	1.0	United States	1.0	1.0	South Korea	1.0	1.0	Japan	1.0	1.0	France-Korea	1.0	1.0	Sweden	1.0	1.0
Country	Home Score (Median)	Away Score (Median)																																																			
England	1.5	1.0																																																			
Italy	1.0	1.0																																																			
France	1.0	1.0																																																			
Spain	1.0	1.0																																																			
South America	2.0	1.5																																																			
Sweden	1.5	1.0																																																			
China	1.0	1.0																																																			
Germany	1.0	1.0																																																			
Mexico	1.0	1.0																																																			
Indonesia	1.0	1.0																																																			
Japan	1.0	1.0																																																			
United States	1.0	1.0																																																			
South Korea	1.0	1.0																																																			
Japan	1.0	1.0																																																			
France-Korea	1.0	1.0																																																			
Sweden	1.0	1.0																																																			
Line Chart: Matches Played by Month	June: Over 700 matches, the peak month for matches. - May and July: 100-200 matches each. Steep increase from May to June, followed by a sharp drop in July, reflecting typical World Cup schedules.	<p>Number of Matches by Month</p>  <table><tr><th>Month</th><th>Number of Matches</th></tr><tr><td>Jul</td><td>150</td></tr><tr><td>Aug</td><td>350</td></tr><tr><td>Sep</td><td>450</td></tr><tr><td>Oct</td><td>550</td></tr><tr><td>Nov</td><td>650</td></tr><tr><td>Dec</td><td>700</td></tr><tr><td>Jan</td><td>650</td></tr><tr><td>Feb</td><td>550</td></tr><tr><td>Mar</td><td>450</td></tr><tr><td>Apr</td><td>350</td></tr><tr><td>May</td><td>150</td></tr></table>	Month	Number of Matches	Jul	150	Aug	350	Sep	450	Oct	550	Nov	650	Dec	700	Jan	650	Feb	550	Mar	450	Apr	350	May	150																											
Month	Number of Matches																																																				
Jul	150																																																				
Aug	350																																																				
Sep	450																																																				
Oct	550																																																				
Nov	650																																																				
Dec	700																																																				
Jan	650																																																				
Feb	550																																																				
Mar	450																																																				
Apr	350																																																				
May	150																																																				
Box Plot: Comparison of Home and Away Goals	Home Goals: Median 1-2, broader distribution, outliers up to 9-10 goals. Away Goals: Similar median but fewer extreme outliers. Confirms that home teams generally score slightly more than away teams.	<p>Comparison of Home and Away Scores</p>  <table><tr><th>Score Type</th><th>Median</th><th>Q1</th><th>Q3</th><th>Min</th><th>Max</th></tr><tr><td>Home Score</td><td>1.5</td><td>1.0</td><td>2.0</td><td>0.0</td><td>5.0</td></tr><tr><td>Away Score</td><td>1.0</td><td>0.5</td><td>2.0</td><td>0.0</td><td>5.0</td></tr></table>	Score Type	Median	Q1	Q3	Min	Max	Home Score	1.5	1.0	2.0	0.0	5.0	Away Score	1.0	0.5	2.0	0.0	5.0																																	
Score Type	Median	Q1	Q3	Min	Max																																																
Home Score	1.5	1.0	2.0	0.0	5.0																																																
Away Score	1.0	0.5	2.0	0.0	5.0																																																
Histogram: Total Goals per Match	Most matches feature 0-4 goals, indicating low-scoring games are the most frequent. Higher goal counts are rare, showcasing a conservative scoring trend in World Cup matches	<p>Distribution of Total Goals per Match</p>  <table><tr><th>Number of Goals</th><th>Frequency</th></tr><tr><td>0</td><td>240</td></tr><tr><td>1</td><td>190</td></tr><tr><td>2</td><td>195</td></tr><tr><td>3</td><td>125</td></tr><tr><td>4</td><td>75</td></tr><tr><td>5</td><td>60</td></tr><tr><td>6</td><td>10</td></tr><tr><td>7</td><td>5</td></tr><tr><td>8</td><td>2</td></tr><tr><td>9</td><td>1</td></tr><tr><td>10</td><td>1</td></tr><tr><td>11</td><td>1</td></tr><tr><td>12</td><td>1</td></tr></table>	Number of Goals	Frequency	0	240	1	190	2	195	3	125	4	75	5	60	6	10	7	5	8	2	9	1	10	1	11	1	12	1																							
Number of Goals	Frequency																																																				
0	240																																																				
1	190																																																				
2	195																																																				
3	125																																																				
4	75																																																				
5	60																																																				
6	10																																																				
7	5																																																				
8	2																																																				
9	1																																																				
10	1																																																				
11	1																																																				
12	1																																																				

Stacked Bar Chart: Wins by Year (1930–2018)	Home Wins (Blue): Dominate most years, showing the advantage of playing at home. - Away Wins (Red): Show an upward trend, particularly after the 1980s. Total number of matches and wins grows consistently over time.	
Bar Chart: Average Scores by Month	Home Teams: Highest average score in May (~1.75 goals). - Away Teams: Highest average score in May as well (~1.5 goals). Home teams consistently score more than away teams across May, June, and July.	
Box Plot: Scores Across Tournament Stages	Home Scores: Median and variability change depending on the stage, with more outliers in early stages. - Away Scores: Similar trends but with fewer outliers and a slightly lower range. Highlights differing scoring patterns across stages, with finals typically showing tighter score distributions.	

4- Data preprocessing:

- Checking for missing values:

```
Missing values in each column:
year                0
country             0
city               0
stage              0
home_team           0
away_team           0
home_score          0
away_score          0
outcome             0
win_conditions      838
winning_team        169
losing_team         169
date                0
month               0
dayofweek           0
dtype: int64
```

```
Missing values after filling:
year                0
country             0
city               0
stage              0
home_team           0
away_team           0
home_score          0
away_score          0
outcome             0
win_conditions      0
winning_team        0
losing_team         0
date                0
month               0
dayofweek           0
dtype: int64
```

Description:

null values can have a significant negative impact on the efficiency of a dataset and the quality of insights derived from it later . To address this issue, we carefully checked the dataset for any missing or null values and handled them appropriately. For the target columns `'winning_team'` and `'losing_team'`, we replaced missing values with the global constant `'Draw match'`, as the absence of data indicated a tie. Similarly, for the `'win_conditions'` column, we used `'no win conditions'` to represent cases where no specific conditions applied. These adjustments ensured a

more consistent and efficient dataset.

- **Detecting and removing the outliers**

```
Outlier Counts:  
year: 0 rows with outliers  
home_score: 24 rows with outliers  
away_score: 11 rows with outliers  
Total Rows with Outliers: 35
```

```
Outlier Counts:  
year: 0 rows with outliers  
home_score: 0 rows with outliers  
away_score: 0 rows with outliers  
Total Rows with Outliers: 0
```

Description:

We identified 35 outliers in our dataset, out of a total of 900 rows. To address this, we used the interquartile range (IQR) method. Rather than removing the outliers entirely, we chose to cap them by replacing them with the closest non-outlier values. This strategy helps maintain the integrity of the dataset while reducing the effect of extreme values on our analyses. By implementing this approach, we ensured that valuable data was preserved while still accounting for the presence of outliers.

- **Data Transformation:**

1. Encoding:

```

    year country      city      stage home_team away_team \
0   1930 Uruguay    Montevideo   Group 1      France      Mexico
1   1930 Uruguay    Montevideo   Group 4      Belgium  United States
2   1930 Uruguay    Montevideo   Group 2      Brazil   Yugoslavia
3   1930 Uruguay    Montevideo   Group 3      Peru     Romania
4   1930 Uruguay    Montevideo   Group 1      Argentina France
..   ...   ...           ...           ...           ...
895  2018 Russia      Sochi   Quarterfinals  Russia      Croatia
896  2018 Russia  Saint Petersburg Semifinals   France      Belgium
897  2018 Russia      Moscow Semifinals   Croatia     England
898  2018 Russia  Saint Petersburg Third place  Belgium     England
899  2018 Russia      Moscow      Final      France      Croatia

    home_score away_score outcome      win_conditions \
0           4           1      H      no win conditions
1           0           3      A      no win conditions
2           1           2      A      no win conditions
3           1           3      A      no win conditions
4           1           0      H      no win conditions
..   ...           ...   ...           ...
895          2           2      A  Croatia won in penalties (4 - 3)
896          1           0      H      no win conditions
897          2           1      H      Croatia won in AET
898          2           0      H      no win conditions
899          4           2      H      no win conditions
..   ...
898          Belgium  England  2018-07-14  Jul  Saturday
899          France   Croatia  2018-07-15  Jul  Sunday

```

```

Encoded DataFrame
    year country  city stage home_team away_team home_score away_score \
0   1930     16    83    3      27      42          4          1
1   1930     16    83    6       5      76          0          3
2   1930     16    83    4       8      80          1          2
3   1930     16    83    5      52      56          1          3
4   1930     16    83    3       2      27          1          0
..   ...   ...   ...   ...   ...   ...   ...
895  2018      9   134   18      57      16          2          2
896  2018      9   120   20      27       5          1          0
897  2018      9    85   20      16      25          2          1
898  2018      9   120   21       5      25          2          0
899  2018      9    85    1      27      16          4          2

    outcome win_conditions winning_team losing_team      date month \
0          2              48          22          46  1930-07-13    0
1          0              48          62           5  1930-07-13    0
2          0              48          66           8  1930-07-14    0
3          0              48          45          56  1930-07-14    0
4          2              48           1          30  1930-07-15    0
..   ...   ...   ...   ...   ...   ...
895          0              15          12          61  2018-07-07    0
896          2              48          22           5  2018-07-10    0
897          2              13          12          28  2018-07-11    0
898          2              48           4          28  2018-07-14    0
..   ...
898          2
899          3

```

Description:

Description:

We needed to encode several nominal attributes in our dataset to numerical values to simplify processing and facilitate analysis. For instance, the attribute "country," which initially contained text values such as "France," "Germany," and "Spain," was encoded to 0, 1, and 2 respectively. This transformation allows algorithms to process these values more efficiently while maintaining the original meaning.

Similarly, other attributes like "stage" were encoded into numerical values where "Group Stage" was replaced with 0, "Quarterfinals" with 1, and "Final" with 2. This makes it easier to perform operations and comparisons across stages.

This process was repeated for attributes like "city," "home_team," "away_team," "winning_team," and "losing_team," among others. By encoding these categorical attributes into numerical values, the dataset becomes more interpretable for machine learning algorithms while retaining its original information.

2. Normalization:

...	year	country	city	stage	home_team	away_team	\
0	1930	Uruguay	Montevideo	Group 1	France	Mexico	
1	1930	Uruguay	Montevideo	Group 4	Belgium	United States	
2	1930	Uruguay	Montevideo	Group 2	Brazil	Yugoslavia	
3	1930	Uruguay	Montevideo	Group 3	Peru	Romania	
4	1930	Uruguay	Montevideo	Group 1	Argentina	France	
..
895	2018	Russia	Sochi	Quarterfinals	Russia	Croatia	
896	2018	Russia	Saint Petersburg	Semifinals	France	Belgium	
897	2018	Russia	Moscow	Semifinals	Croatia	England	
898	2018	Russia	Saint Petersburg	Third place	Belgium	England	
899	2018	Russia	Moscow	Final	France	Croatia	
	home_score	away_score	outcome	win_conditions \			
0	4	1	H	no win conditions			
1	0	3	A	no win conditions			
2	1	2	A	no win conditions			
3	1	3	A	no win conditions			
4	1	0	H	no win conditions			
..			
895	2	2	A	Croatia won in penalties (4 - 3)			
896	1	0	H	no win conditions			
897	2	1	H	Croatia won in AET			
898	2	0	H	no win conditions			
899	4	2	H	no win conditions			
...							
898	Belgium	England	2018-07-14	Jul	Saturday		
899	France	Croatia	2018-07-15	Jul	Sunday		
[900 rows x 15 columns]							

DataFrame after Decimal Scaling Normalization:									
	year	country	city	stage	home_team	away_team	home_score	away_score	\
0	1930	16	83	3	27	42	0.000004	0.000001	
1	1930	16	83	6	5	76	0.000000	0.000003	
2	1930	16	83	4	8	80	0.000001	0.000002	
3	1930	16	83	5	52	56	0.000001	0.000003	
4	1930	16	83	3	2	27	0.000001	0.000000	
..
895	2018	9	134	18	57	16	0.000002	0.000002	
896	2018	9	120	20	27	5	0.000001	0.000000	
897	2018	9	85	20	16	25	0.000002	0.000001	
898	2018	9	120	21	5	25	0.000002	0.000000	
899	2018	9	85	1	27	16	0.000004	0.000002	
	outcome	win_conditions	winning_team	losing_team	date	month	\		
0	2	48	22	46	0	0			
1	0	48	62	5	0	0			
2	0	48	66	8	1	0			
3	0	48	45	56	1	0			
4	2	48	1	30	2	0			
..			
895	0	15	12	61	350	0			
896	2	48	22	5	351	0			
897	2	13	12	28	352	0			
898	2	48	4	28	353	0			
...									
898	2								
899	3								

In this normalization technique, we normalize the attributes to have a unified scale since the range of values varies significantly across attributes. This approach standardizes the dataset values, making the analysis process more efficient.

3. Aggregation:

Country Stats:				
	country	home_score_mean	away_score_mean	total_wins
0	0	0.142105	0.121053	9
1	1	0.144186	0.147674	12
2	2	0.140625	0.134375	5
3	3	0.143750	0.134375	5
4	4	0.175610	0.128049	19
5	5	0.138235	0.094118	21
6	6	0.147826	0.117391	10
7	7	0.112500	0.100000	8
8	8	0.147619	0.120238	16
9	9	0.135938	0.126562	9
10	10	0.117188	0.106250	14
11	11	0.162500	0.118750	6
12	12	0.173077	0.098077	16
13	13	0.191429	0.157143	10
14	14	0.250000	0.226923	5
15	15	0.155769	0.113462	8
16	16	0.211111	0.161111	4

In the aggregation method, we grouped the "Country" column and applied an aggregation function to the data. Specifically, we calculated the mean values for "Home Score" and "Away Score" for each country, as well as the total number of wins for the most frequent winning team in each country. This step helps us analyze differences in scoring patterns between countries and identify which teams tend to dominate in specific regions. By aggregating the data in this way, we can uncover trends and patterns that provide valuable insights for further analysis or decision-making.

4. Discretization:

	year	country	city	stage	home_team	away_team	home_score	away_score	\
0	1930	16	83	3	27	42	0.4	0.1	
1	1930	16	83	6	5	76	0.0	0.3	
2	1930	16	83	4	8	80	0.1	0.2	
3	1930	16	83	5	52	56	0.1	0.3	
4	1930	16	83	3	2	27	0.1	0.0	
	outcome	win_conditions	winning_team	losing_team	date	month	\		
0	2		48	22	46 1930-07-13	0			
1	0		48	62	5 1930-07-13	0			
2	0		48	66	8 1930-07-14	0			
3	0		48	45	56 1930-07-14	0			
4	2		48	1	30 1930-07-15	0			
	dayofweek	Year_Decade	Month						
0	3	1930s	July						
1	3	1930s	July						
2	1	1930s	July						
3	1	1930s	July						
4	5	1930s	July						

In the discretization method, the “year” column was categorized into meaningful decade ranges, assigning each year to its corresponding decade label (e.g., “1930s”). Furthermore, the “date” column was transformed into a datetime format, and the month name was extracted into a new column called “Month.” These classifications enhance the interpretability of the data by organizing temporal information into structured categories, enabling clearer analysis of events across decades and months..

• **Balance Data:**

Before starting the Data Mining Technique, we investigated whether the data was balanced or not:

```
Number of Home Wins: 429
Number of Away Wins: 302
Number of Draws: 169
-
Percentage of Home Wins: 47.67%
Percentage of Away Wins: 33.56%
Percentage of Draws: 18.78%
```

In the initial analysis, we reviewed the distribution of outcomes in the dataset (Home Wins, Away Wins, and Draws). It was observed that the percentages are imbalanced, as they do not fall within a balanced range of 40% to 60%, indicating a disparity between the categories.

- **Process of correcting data balancing**

```
Binary_outcome
1      598
0      598
Name: count, dtype: int64
```

- **Data after the balancing process:**

By applying the SMOTE (Synthetic Minority Oversampling Technique) method, we generated Synthetic samples for the minority class (Away Wins) to achieve balance between the two classes (Home Wins and Draws combined vs. Away Wins). This approach ensures that the model does not become biased towards the majority class and enhances its ability to generalize effectively to new data.

```
Percentage of Home wins and Draws combined: 50.00%
Percentage of Away wins: 50.00%
```

Finally, we calculated the percentage for each category to ensure that the data has become balanced. The three categories represent home wins, away wins, and draws, and balance has been achieved as the percentages for each category fall within the range of 40% to 60%.

5- Data Mining Techniques:

We employed both supervised and unsupervised learning methods on our dataset using **classification** and **clustering** techniques.

Classification Task

For the classification task, we used a **decision tree**, a recursive algorithm that builds a tree structure where each leaf node corresponds to a final decision. Our model was designed to predict the outcome of football matches, classifying results into two categories: **'0' (Away win)** or **'1' (Home win + Draw)**. The predictions were based on several attributes, such as **home_team, away_team, country, match conditions, and match-specific features**.

As a supervised learning approach, classification requires training data to build the model. To this end, we split our dataset into **training and testing subsets**, experimenting with three different splits: **70% training/30% testing**, **60% training/40% testing**, and **80% training/20% testing**. Additionally, we employed two attribute selection measures: **Information Gain (Entropy)** and **Gini Index**, to evaluate the best splitting criteria for the tree.

To assess the performance of our model, we evaluated its accuracy and examined confusion matrices. These matrices provided insights into key performance metrics such as **sensitivity, specificity, precision**, and **error rate**, helping us determine the most effective dataset partitioning and attribute selection method.

Clustering Task

In the clustering process, an unsupervised learning approach, we omitted the target variable ('Binary_outcome') as clustering does not utilize class labels. Instead, we used all other attributes, including **home_team**, **away_team**, **country**, **month**, **match-specific features**, and **others**, which were preprocessed using **one-hot encoding** for categorical variables.

We applied the **K-means algorithm** for clustering, which works by generating **K clusters** and assigning each data point to the nearest cluster based on their feature values. The algorithm iteratively recalculates the centroids of the clusters and reassigns objects until the centroids stabilize, signifying optimal cluster assignments.

To validate the clustering results, we computed the **Average Silhouette Score** for each cluster, which measures how well each data point fits within its cluster compared to others. Furthermore, we used the **Within-Cluster Sum of Squares (WSS)** method to evaluate different numbers of clusters and determine the optimal cluster count by analyzing the compactness and separation of the clusters.

This approach allowed us to comprehensively analyze the dataset through both supervised and unsupervised learning, providing valuable insights into classification accuracy and optimal cluster formation.

6- Evaluation and Comparison:

- **Classification [70% Training, 30% Testing] Information Gain:**

Figure (1) (decision tree):

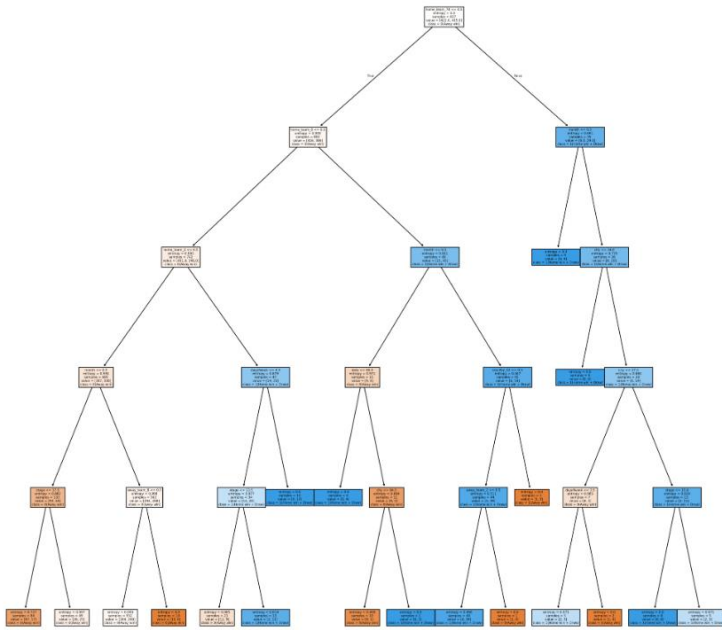
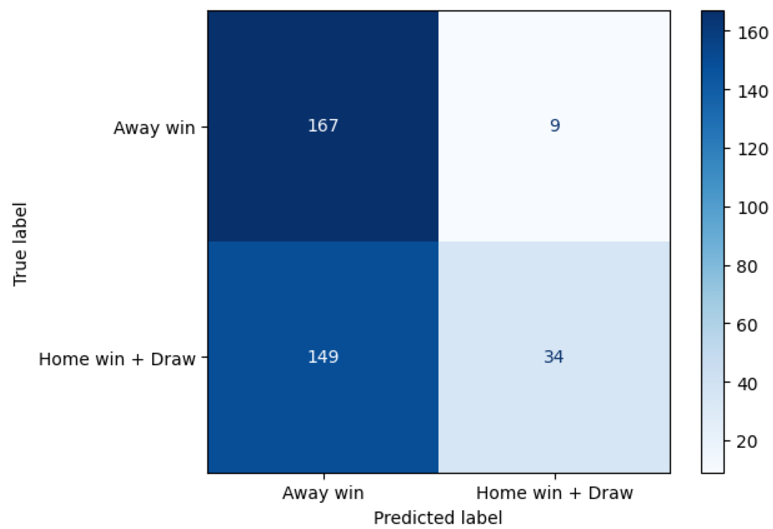


Figure (2) (confusion matrix):



- Classification [60% Training and 40% Testing] **Information Gain:**

Figure (1) (decision tree):

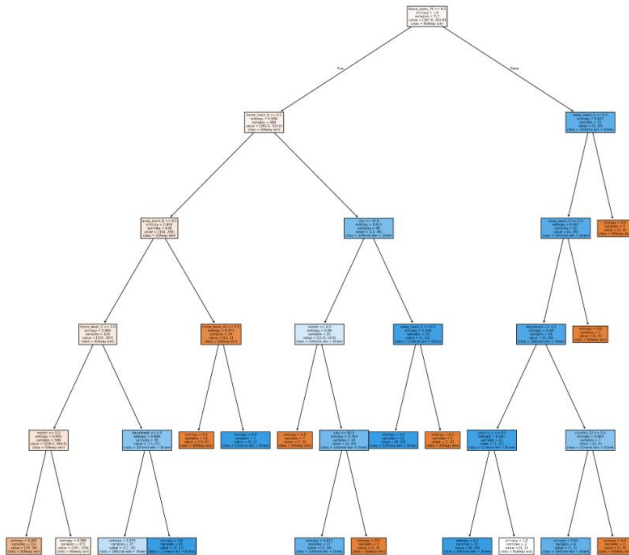
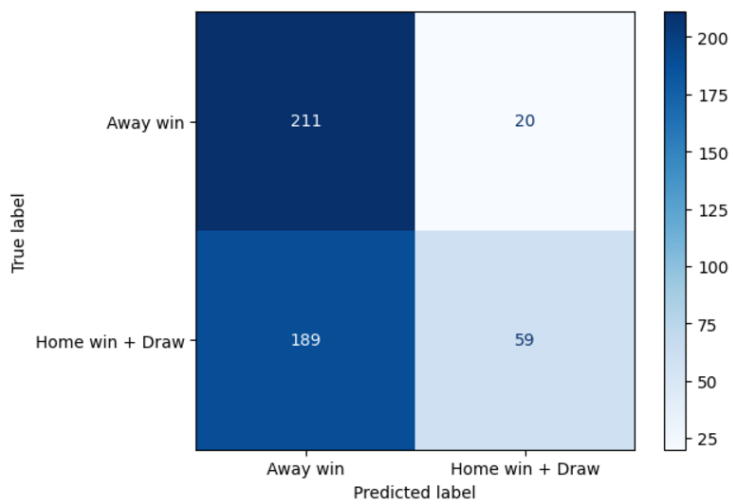


Figure (2) (confusion matrix):



- Classification [80% Training and 20% Testing] **Information Gain:**

Figure (1) (decision tree):

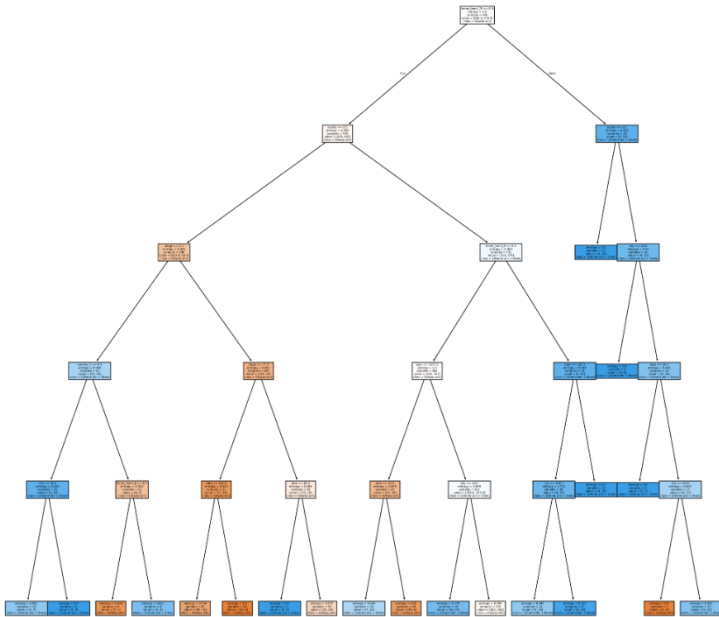
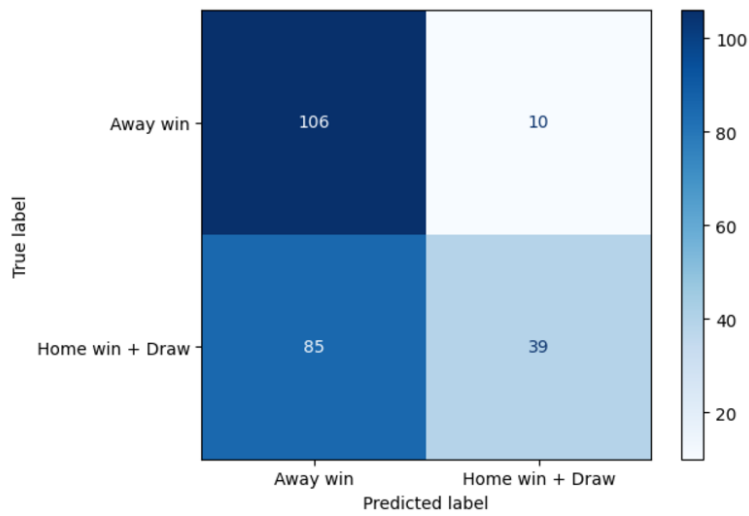


Figure (2) (confusion matrix):



Mining task	Comparison Criteria
Classification for Information Gain	We tried 3 different sizes for dataset splitting to create the decision tree:
	- 70% Training data, 30% Test data.
	Accuracy55.99%
	Precision79.07%
	Sensitivity18.58%
	Specificity94.89%
	Error rate44.01%
	- 60% Training data, 40% Test data.
	Accuracy56.37%
	Precision74.68%
	Sensitivity23.80%
	Specificity91.34%

	Error rate	43.63%	
	- 80% Training data, 20% Test data.		
	Accuracy	60.42%	
	Precision	79.59%	
	Sensitivity	31.45%	
	Specificity	91.38%	
	Error rate	39.58%	

- Classification [70% Training, 30% Testing] **Gini Index:**

Figure (1) (decision tree):

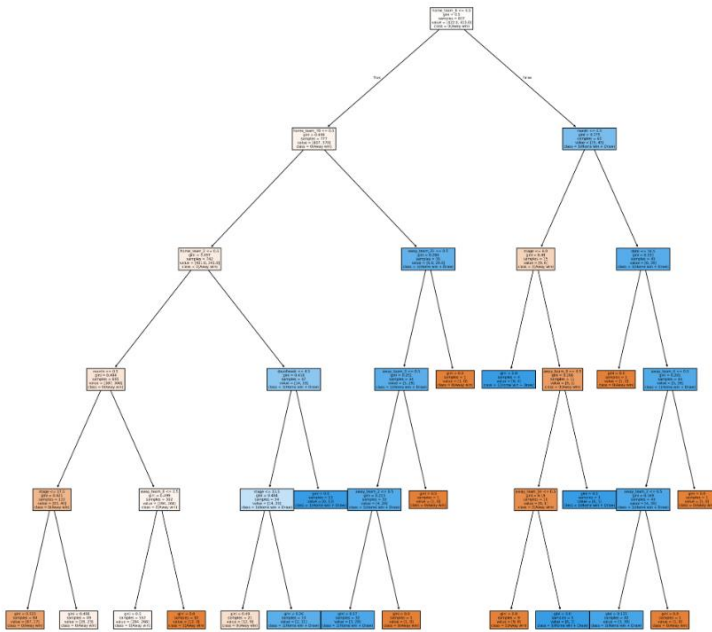
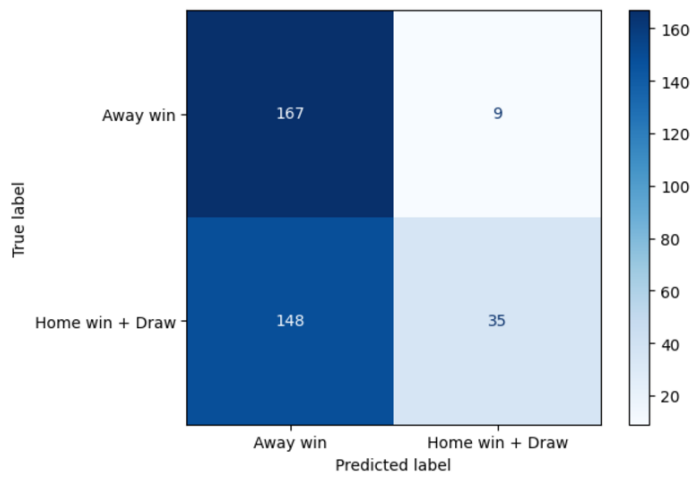


Figure (2) (confusion matrix):



- Classification [60% Training and 40% Test] **Gini Index:**

Figure (1) (decision tree):

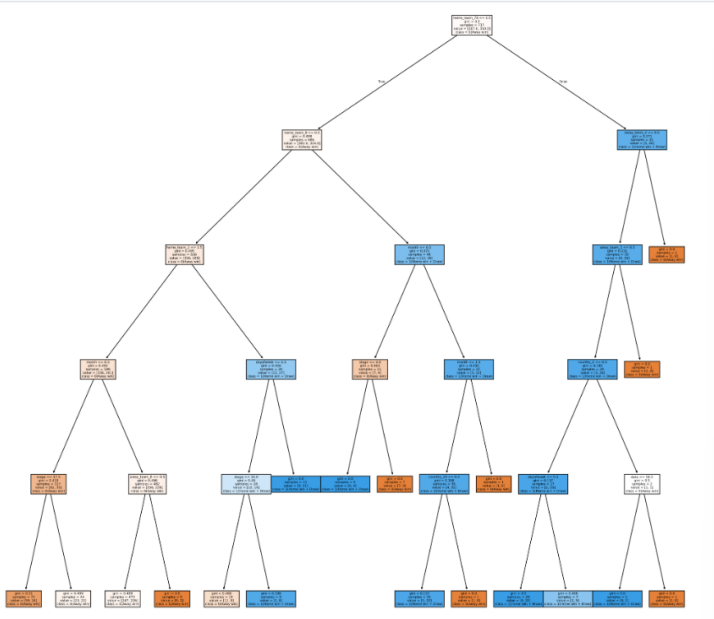
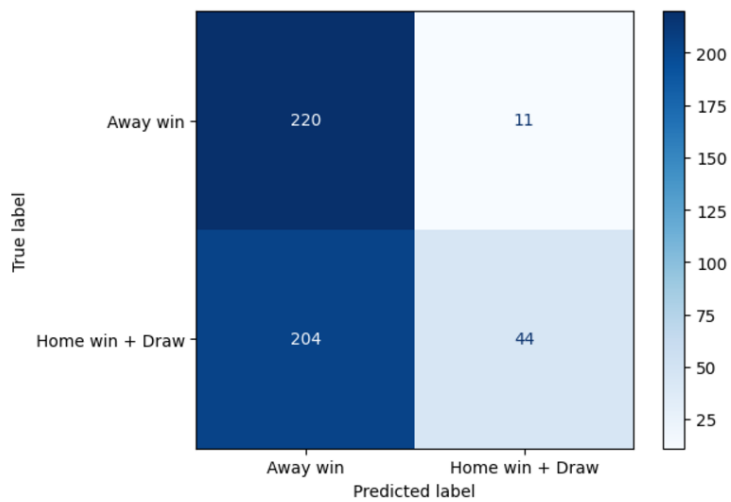


Figure (2) (confusion matrix):



- Classification [80% Training and 20% Testing] **Gini Index:**

Figure (1) (decision tree):

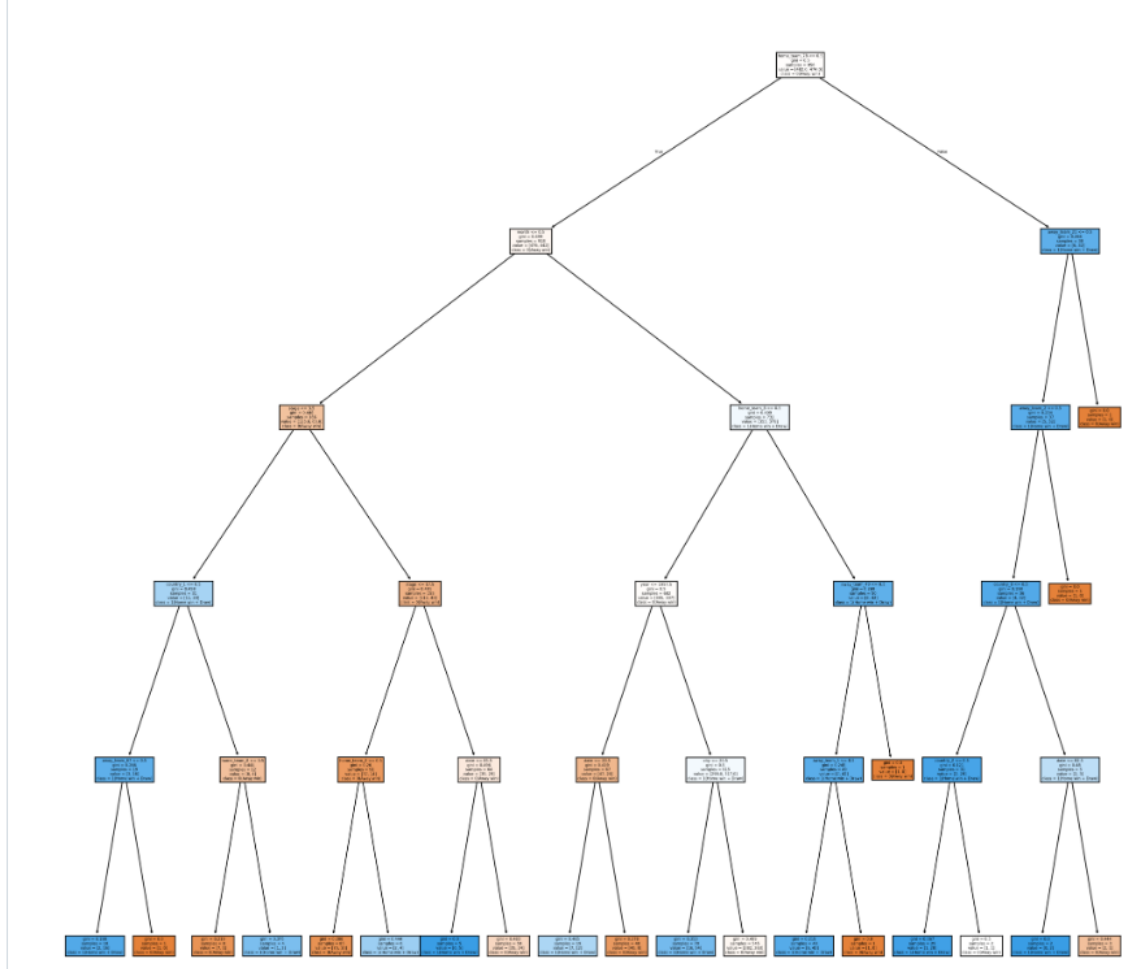
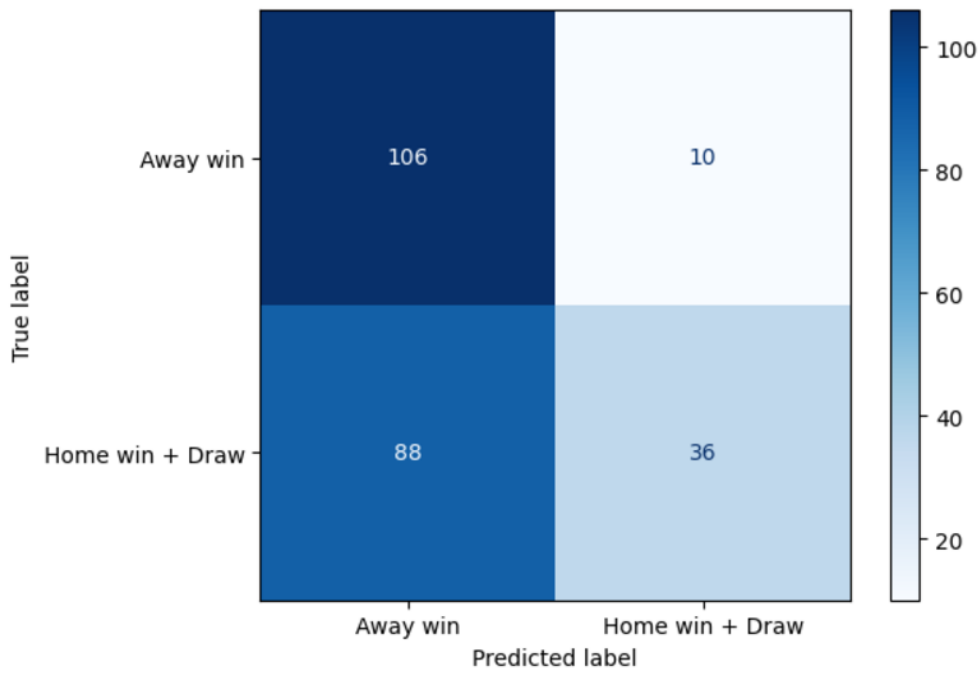


Figure (2) (confusion matrix):



Mining task	Comparison Criteria
Classification for Gini Index	We tried 3 different sizes for dataset splitting to create the decision tree:
	- 70% Training data, 30% Test data.
	Accuracy56.26%
	Precision79.54%
	Sensitivity19.12%
	Specificity94.88%
	Error rate43.73%
	- 60% Training data, 40% Test data.
	Accuracy55.12%
	Precision80%
	Sensitivity17.74%
	Specificity95.23%
	Error rate44.88%

-80% Training data, 20% Test data.

Accuracy	59.17%
precision	78.26%
sensitivity	29.03%
specificity	91.37%
Error rate	40.83%

- **the better partitioning:**

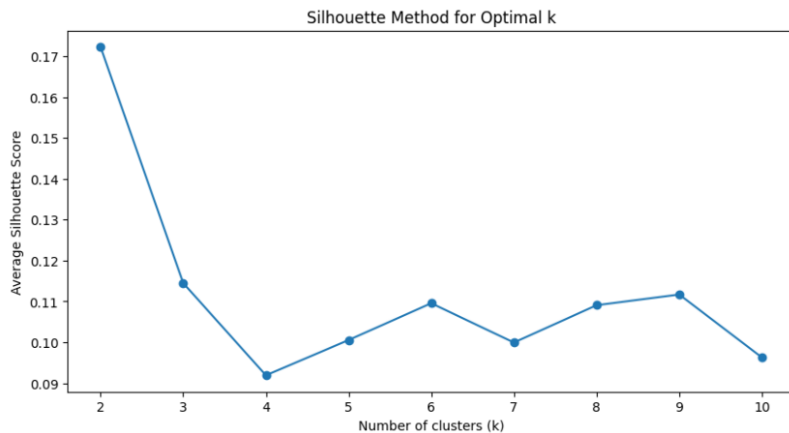
In summary, the 80/20 split model strikes a better balance overall, demonstrating the highest accuracy and a reasonable trade-off between sensitivity and specificity, making it a strong choice for prediction tasks based on the provided data.

- **Clustering**

We choose 3 different sizes [2,3,6] based on the result of the validation methods that we will apply then we will use these sizes to perform the k-means clustering.

Silhouette method:

The Silhouette method is a technique used to evaluate the quality of clustering results. It measures how well each data point fits within its assigned cluster compared to neighboring clusters.



Elbow method:

The Elbow method is a technique used to determine the optimal number of clusters in a dataset for K-means clustering

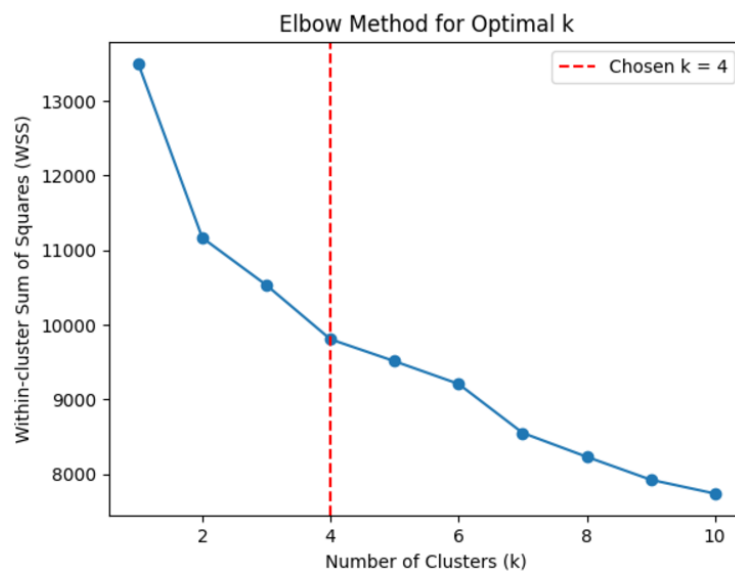


Figure (1): silhouette scores [K=2]



Figure (2): silhouette scores [K=3]

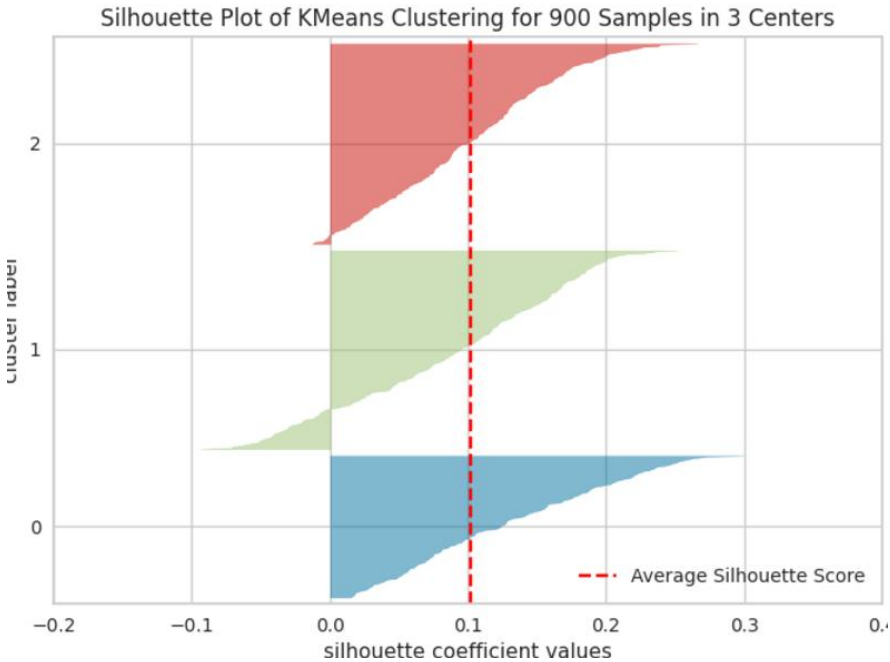


Figure (3): silhouette scores [K=6]



Mining task	Comparison Criteria			
Clustering	We tried 3 different sizes for dataset splitting			
	to create the decision tree:			
	K=2, K=3, K=6			
	No. of clusters	K=6 (BEST)	K=2	K=3
	Average Silhouette width	0.1106	0.1715	0.1017
	total within-cluster sum of square	8776.723	11166.540	8776.723

Findings:

Initially, we selected a dataset representing football teams' performance metrics with the goal of understanding patterns and characteristics that contribute to match outcomes (such as "Away win" or "Home win + Draw"). To ensure effective analysis and reliable predictions, several data preprocessing techniques were applied to enhance data efficiency and improve accuracy.

We utilized various **visualization methods** such as **box plots**, **scatter plots**, and **histograms** to clarify the data and facilitate its interpretation. These visualizations enabled us to identify outliers and make informed decisions about necessary data cleaning steps. We removed all **missing values**, **empty entries**, and **outliers** that could potentially affect the results negatively.

Furthermore, **data transformations** including **normalization** and **feature partitioning** were implemented to streamline the dataset, ensuring equal weight distribution across features and optimizing data processing for the mining tasks.

Consequently, we conducted data mining tasks, encompassing classification and partitioning. For classification, we employed the Gini index and information gain metrics. Experimenting with three different sizes of training and testing data allowed us to achieve optimal results for both model construction and evaluation. Here are our findings:

- Information Gain:

Evaulate the Models of Information Gain

Train/Test split	70/30 split	60/40 split	80/20 split
Accuracy	0.5599	0.5637	0.6042
Error Rate	0.4401	0.4363	0.3958
Sensitivity	0.1858	0.238	0.3152
Specificity	0.9489	0.9134	0.9138
Precision	0.7907	0.7468	0.7959

- Accuracy:** Among the different splits, the model trained on the 80% training set and 20% testing set achieved the highest accuracy of 60.42%, followed by the 60/40 split model at 56.37%, and the 70/30 split model at 55.99%. This indicates that the 80/20 split model has a slightly better predictive performance in terms of overall accuracy.
- Error Rate:** The 80/20 split model also exhibited the lowest error rate at 39.58%, suggesting it has fewer misclassifications compared to the other models. The error rate increases with the 60/40 split at 43.63% and reaches its highest at 44.01% for the 70/30 split.
- Sensitivity (Recall):** The 80/20 split model shows the best sensitivity at 31.52%, indicating a slightly better performance in correctly identifying 'Home win + Draw' outcomes. The sensitivity drops to 23.8% for the 60/40 split and is lowest for the 70/30 split at 18.58%.
- Specificity:** The specificity of the models remains relatively high across all splits. The 60/40 split model and 80/20 split model have similar values around 91.38%, while the 70/30 split model achieves a slightly higher specificity at 94.88%. This shows that all models are effective at correctly identifying 'Away win' outcomes.

- **Precision:** The 80/20 split model has the highest precision at 79.59%, followed by the 70/30 split at 79.07%, and the 60/40 split at 74.68%. This suggests that, among the positive predictions, the 80/20 split model is slightly better at predicting true positive cases.

Summary:

The model trained using the 80% training set and 20% testing set generally shows the best balance in terms of accuracy, error rate, sensitivity, and precision, making it a strong candidate for predicting match outcomes. However, the 70/30 split model has the highest specificity, indicating a better performance in correctly identifying 'Away win' outcomes. Overall, the choice of the best model depends on the specific requirements and goals, such as maximizing sensitivity for capturing positive outcomes or achieving high specificity for minimizing false positives.

- Gini index:

Train/Test split	70/30 split	60/40 split	80/20 split
Accuracy	0.5626	0.5512	0.5917
Error Rate	0.4373	0.4488	0.4083
Sensitivity	0.1912	0.1774	0.2903
Specificity	0.9488	0.9523	0.9137
Precision	0.7954	0.8	0.7826

Based on the presented results the 80-20 split model is considered the best . Here are some reasons why this model outperforms the others:

- **Highest Accuracy:** The **80/20 split** achieved the highest accuracy.
- **Lowest Error Rate:** The **80/20 split** also had the lowest error rate, indicating it made fewer incorrect predictions compared to the other models.
- **Highest Sensitivity:** The **80/20 split** had the highest sensitivity, meaning it was better at identifying **‘Home win + Draw’** outcomes.
- **Strong Specificity:** While the **60/40 split** had the highest specificity, the **80/20 split** still performed well in identifying **‘Away win’** outcomes, making it highly effective.
- **Best Precision:** The **80/20 split** had high precision, coming close to the **70/30** split, but overall performed better in correctly predicting positive cases.

In summary, the 80-20 model strikes a good balance between classification accuracy, specificity, and sensitivity, which is why it is considered the best based on the provided results.

- The best model between information gain and the Gini index:

Evaluate the Best Model between Information Gain and Gini Index

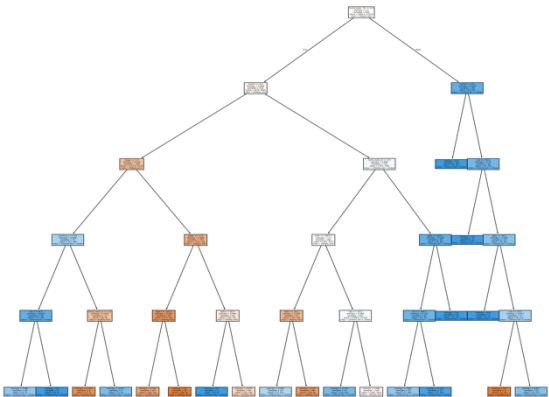
The comparison of the models trained with an 80/20 split that have resulted in the greatest accuracies:

Model	Information Gain	Gini Index
Accuracy	0.6042	0.5917
Error Rate	0.3958	0.4083
Sensitivity	0.3145	0.2903
Specificity	0.9138	0.9137
Precision	0.7959	0.7826

- **Accuracy and Error Rate:** The **Information Gain** model achieved slightly higher accuracy and a lower error rate compared to the **Gini Index** model, indicating it performs better overall in classifying instances correctly.
- **Sensitivity:** The **Information Gain** model showed better sensitivity, making it more effective at identifying ‘**Home win + Draw**’ outcomes.
- **Specificity:** Both models had similar specificity, with the **Information Gain** model being marginally better, indicating strong performance in identifying ‘**Away win**’ outcomes.
- **Precision:** The **Information Gain** model had slightly higher precision, suggesting it was more accurate in predicting positive outcomes.

Conclusion: The **Information Gain** model slightly outperforms the **Gini Index** model across key metrics, making it more effective at identifying positive instances while maintaining a strong balance in **specificity** and **precision**.

This was the decision tree associated with this division:



- **Root Node:** The root node is the first decision point. It splits the data based on the feature that gives the most information about the match outcome. This helps to make the first, most important decision.
- **Internal Nodes:** As the tree moves down, each internal node represents a decision point where the data is split further. The tree looks at features like **team performance** and **match conditions** to separate the data and make more detailed predictions.
- **Branches:** The branches show the possible outcomes of each decision at an internal node. They lead either to another internal node or to the final decision at a **leaf node**.
- **Leaf Nodes:** The leaf nodes at the end of the tree give the final prediction. They tell us whether the match is likely to end

in an **Away win** or a **Home win + Draw**. Each leaf node also shows how many matches reached that point and the probability of each outcome.

- **Path:** A path from the root node to a leaf node represents a decision rule. For example, "**If team performance is high and match conditions are favorable, predict 'Home win + Draw'.**" This shows how the tree makes predictions based on the decisions made at each node.

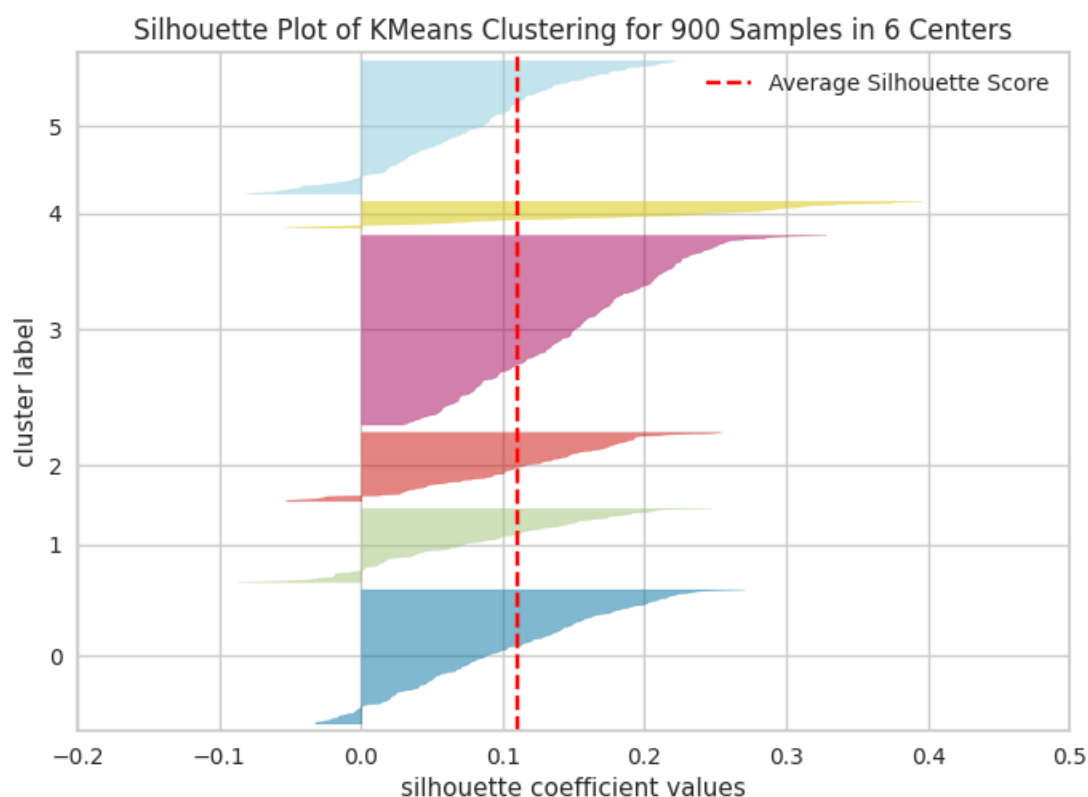
For Clustering, we used K-means algorithm with 3 different K to find the optimal number of clusters, we calculated the average silhouette width for each K, and we concluded the following results:

Metric	K=3	K=6	K=2
Average Silhouette width	0.1016929619176413	0.11058957961188719	0.17145331666962568
WSS	10446.48671611567	8776.722812008877	11166.539537119663

We've decided that **K=6** is the best choice for our clustering model based on the metrics we've analyzed (WSS, Average Silhouette Width, and Visualization of K-means). This choice is because **K=6** provides the best balance between cohesion and separation of the clusters. It has the lowest **WSS** value, indicating more compact and well-defined clusters compared to **K=3** and **K=2**.

Additionally, the silhouette plot for **K=6**, with a higher average silhouette score, further supports our choice, indicating that it creates distinct and well-separated clusters.

And this was the corresponding chart:



From the graph of **KMeans Clustering** for **900 samples** in **6 centers**, the fact that most of the silhouette scores are positive reinforces the idea that the samples are well-matched to their respective clusters and are well-separated from neighboring clusters. This indicates that the clustering solution has effectively divided the data into distinct and cohesive groups.

However, while the majority of silhouette scores being positive is a good sign, it doesn't necessarily mean that the clustering solution is "perfect." There may still be some overlap or ambiguity between clusters, particularly in cases where there are samples with silhouette scores close to **0** or even negative values, indicating areas where clusters are less distinct or where samples might be borderline between clusters.

Finally, while clustering models provide valuable insights into the underlying structure of the data, **supervised learning models** (such as classification) are more accurate and suitable for applications where class labels are known. Since our dataset includes class labels for outcomes (e.g., predicting match results), **supervised learning** methods would be more appropriate for tasks where the expected output is already known and can guide the model for better predictions.

7. References:

- BRENDA_L, " FIFA World Cup 2022 Dataset", Kaggle, Available:
<https://www.kaggle.com/datasets/brenda89/fifa-world-cup-2022>
- "Labs and Lecture Slides," College of Computer Science, Department of Information Technology, King Saud University.