Practical Data Science with Python
Assignment 2

Cover Page

Statement of Solution
I hereby certify that the submitted solution is entirely my own original work. Any parts of the assignment that were derived from other sources are non-essential and have been clearly attributed.

Title
Glass Identification Database

Author Information
Name: Rishabh Talreja
Student ID: s4055735

Affiliations
Institutions: Royal Melbourne Institute of Technology(RMIT), Melbourne
Department: Master of Information Technology (MC 208)

Contact Details
Email: s4055735@student.rmit.edu.au
Phone: +61 0424962377
Address: 25A Worrell Street, Nunawading 3131

Date of Report
18th May 2024

Table of Contents

Summary
This report has two goals: first, it will use the K-Nearest Neighbours (KNN) and Decision Tree algorithms to identify different types of glass based on their chemical composition. Second, it will visualise the data to show how each compound influence the type of glass. The collection is made up of different glass samples that have characteristics that indicate their chemical compositions. The dataset consists of various glass samples with attributes representing their chemical properties. The dataset from the UCI Machine Learning Repository contains 214 instances with 9 features, including Refractive Index (RI), and the chemical composition percentages of Sodium (Na), Magnesium (Mg), Aluminium (Al), Silicon (Si), Potassium (K), Calcium (Ca), Barium (Ba), and Iron (Fe). Both KNN and Decision Tree classifiers were chosen for their simplicity and effectiveness. The models were trained and evaluated on standardized data.Data visualizations were used to explore feature distributions and relationships. KNN achieved 72% and Decision Tree achieved 67%, with KNN outperforming the Decision Tree. Data visualizations explored feature distributions and relationships, revealing that glass windows are the most found at crime scenes. Data visualizations also helped to identify the importance of different elements in distinguishing between glass types. To improve model accuracy, it is advised to look into and correct the dataset's class imbalance in more detail. We should also look into alternative classification algorithms for comparative analysis, add more feature engineering techniques to improve the models' performance, use more sophisticated visualisation techniques to gain a deeper understanding of the data, and think about ensemble methods, which combine multiple classifiers to improve prediction accuracy.

Introduction
In forensic science, it is essential to classify different types of glass, particularly when glass fragments are discovered at crime scenes. Although different varieties of glass have overlapping qualities, accurately classifying glass can reveal vital evidence about the circumstances around a crime[1]. However, this can be a challenging task. In order to determine the origin of broken glass, forensic glass analysis involves identifying the chemical composition of the fragments, including the percentages of sodium (Na), magnesium (Mg), aluminium (Al), silicon (Si), potassium (K), calcium (Ca), barium (Ba), and iron (Fe). It is important to understand these techniques because glass is frequently present in a number of situations. This study aims to compare KNN and Decision Tree algorithms for classifying glass types based on their chemical composition and to use data visualization to explore how each element affects glass type. The dataset, from the UCI Machine Learning Repository, includes 214 samples with 9 features: Refractive Index (RI), Na, Mg, Al, Si, K, Ca, Ba, and Fe. The study involves data preprocessing, model training, evaluation, and hyperparameter tuning.

Methodology

1. Data Collection
The dataset used in this study is the Glass Identification dataset from the UCI Machine Learning Repository. It contains 214 instances of glass samples, each characterized by 9 features representing its chemical composition, including Refractive Index (RI), and the weight percentages of Sodium (Na), Magnesium (Mg), Aluminium (Al), Silicon (Si), Potassium (K), Calcium (Ca), Barium (Ba), and Iron (Fe). The target variable is the type of glass, categorized into one of seven classes [2]:

1. Building windows float processed
2. Building windows non-float processed
3. Vehicle windows float processed
4. Vehicle windows non-float processed (none in this dataset)
5. Containers
6. Tableware
7. Headlamps

2. Data Preparation
The initial step involved reading the CSV data file into a variable. This dataset contains 214 instances of glass samples, each characterized by nine features representing its chemical composition.

```
In [2]: #Loading the dataset
        data = pd.read_csv("./Dataset/glass+identification/glass.csv"\
                           ,sep=",",decimal=".",header=None\
                           ,names=["RI","Na","Mg","Al","Si","K","Ca","Ba","Fe","Type of Glass"]\
                           , index_col= 0)

In [3]: # Display the first 10 rows of the dataframe
        print(data.head(10))

             RI     Na    Mg    Al     Si     K    Ca    Ba    Fe  Type of Glass
1    1.52101  13.64  4.49  1.10  71.78  0.06  8.75  0.0  0.00             1
2    1.51761  13.89  3.60  1.36  72.73  0.48  7.83  0.0  0.00             1
3    1.51618  13.53  3.55  1.54  72.99  0.39  7.78  0.0  0.00             1
4    1.51766  13.21  3.69  1.29  72.61  0.57  8.22  0.0  0.00             1
5    1.51742  13.27  3.62  1.24  73.08  0.55  8.07  0.0  0.00             1
6    1.51596  12.79  3.61  1.62  72.97  0.64  8.07  0.0  0.26             1
7    1.51743  13.30  3.60  1.14  73.09  0.58  8.17  0.0  0.00             1
8    1.51756  13.15  3.61  1.05  73.24  0.57  8.24  0.0  0.00             1
9    1.51918  14.04  3.58  1.37  72.08  0.56  8.30  0.0  0.00             1
10   1.51755  13.00  3.60  1.36  72.99  0.57  8.40  0.0  0.11             1
```

To ensure the data is suitable for analysis, the following steps were undertaken:

- Checking for Null Values in each column

```
In [7]: #Calculating number of null Values Belonging to each Colum
        print(data.isnull().sum())

RI              0
Na              0
Mg              0
Al              0
Si              0
K               0
Ca              0
Ba              0
Fe              0
Type of Glass   0
dtype: int64
```

- **Handling Missing Values:** Any missing or infinite values in the dataset were identified and replaced with NaN values, which were subsequently removed to maintain data integrity.

  Code Snippet

```
# Replace infinity values with NaN
data.replace([np.inf, -np.inf], np.nan, inplace=True)
data.dropna(inplace=True)
```

- **Data Standardization:**

  The features were standardized to have a mean of 0 and a standard deviation of 1 using the StandardScaler from the sklearn library. Standardization is a critical preprocessing step, especially for algorithms like K-Nearest Neighbors (KNN) and Decision Trees. KNN is a distance-based algorithm, which means it calculates the distance between data points to determine their proximity and make predictions. If the features are on different scales, the algorithm might give undue importance to features with larger scales, skewing the results. By standardizing the features, we ensure that each feature contributes equally to the distance calculations, leading to more reliable and accurate model performance [3].

```
In [84]: # Standardize the features
         scaler = StandardScaler()
         X_train = scaler.fit_transform(X_train)
         X_test = scaler.transform(X_test)
```

3. Feature Selection

For the purpose of training the model, all nine characteristics (RI, Na, Mg, Al, Si, K, Ca, Ba, and Fe) included in the dataset were used. Since these characteristics reflect the essential chemical characteristics of the glass samples, they were selected on the basis of their relevance to the glass classification task.

4. Model Selection

Two machine learning algorithms were selected for this study: K-Nearest Neighbors (KNN) and Decision Tree. These algorithms were chosen due to their complementary strengths in handling classification tasks.

- **K-Nearest Neighbors (KNN):**

  KNN is a non-parametric method used for classification based on the proximity of data points. It classifies a data point by the majority class among its k-nearest neighbors, where k is a user-defined parameter [4]. KNN was chosen for its simplicity and effectiveness:
  - **Simplicity:** Easy to implement and understand.
  - **Effectiveness:** Can handle irregular decision boundaries well.
  - **Proximity Measures:** Uses distance measures (e.g., Euclidean distance) suitable for the chemical composition data in glass classification.

- **Decision Tree:**

  Decision Trees classify data by splitting it into subsets based on feature values, forming a tree structure where each node represents a decision point. Decision Trees were selected for their:
  - **Interpretability:** Easy to visualize and understand the decision-making process.
  - **Modelling Complex Boundaries:** Can capture non-linear relationships between features and target variables.
  - **Feature Importance:** Provide insights into which features are most relevant for classification.

Both models were trained and evaluated to determine their performance in classifying glass types.

5. Hyperparameter Tuning

Decision Tree:

The hyperparameters tuned included the criterion for splitting nodes ('gini' or 'entropy'), the maximum depth of the tree, the minimum number of samples required to split a node, the minimum number of samples required to be at a leaf node, and the number of features to consider for the best split. This process involved training the model on different combinations of these parameters and selecting the ones that provided the best performance based on cross-validation [5].

Code Snippet

```
# Set up the parameter grid for hyperparameter tuning
param_grid = {
    'criterion': ['gini', 'entropy'],
    'max_depth': [None, 10, 20, 30, 40, 50],
    'min_samples_split': [2, 10, 20],
    'min_samples_leaf': [1, 5, 10],
    'max_features': [None, 'auto', 'sqrt', 'log2']
}
```

KNN:

The number of neighbors (k) was varied from 1 to 24 to find the optimal value. This hyperparameter tuning process involved training the model on different values of k and selecting the one that provided the best performance based on cross-validation [6].

Code Snippet

```
# Hyperparameter tuning using Grid Search
param_grid = {'n_neighbors': np.arange(1, 25)}
knn_gscv = GridSearchCV(KNeighborsClassifier(), param_grid, cv=5)
```

Results

- **Data Visualization**

Glass Type Distribution:

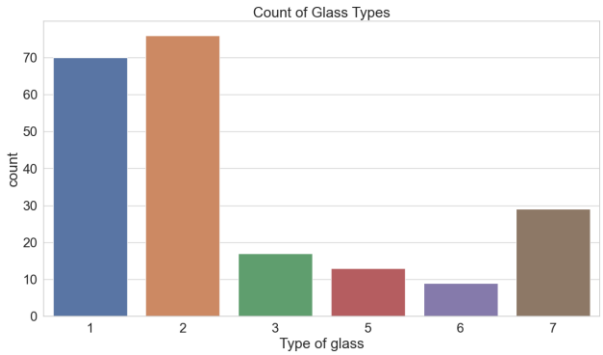

Fig 1: Glass Distribution based on Type

The statistics shows that Types 1 and 2 are the most common, suggesting that building window glasses are used most commonly. This implies that more often than not, thieves smash glass in buildings instead of breaking bottles or cutlery. To create a strong classification model, it is essential to make sure that the train and test datasets have a good representation from all kinds.

Histogram

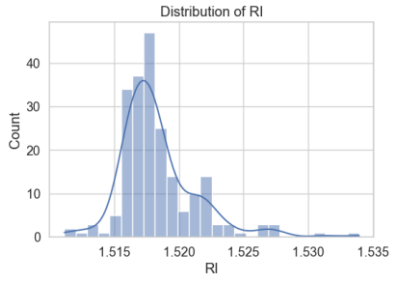

Fig 2: Distribution of RI

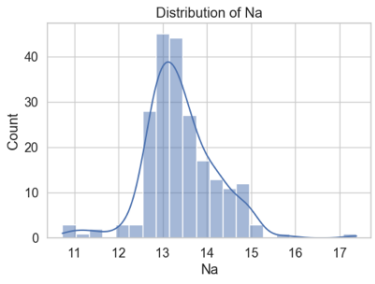

Fig 3: Distribution of Na.

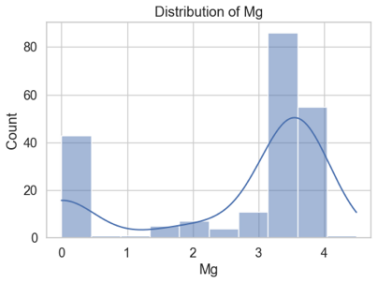

Fig 4: Distribution of Mg

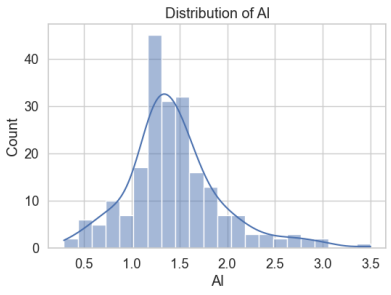

Fig 4: Distribution of AI

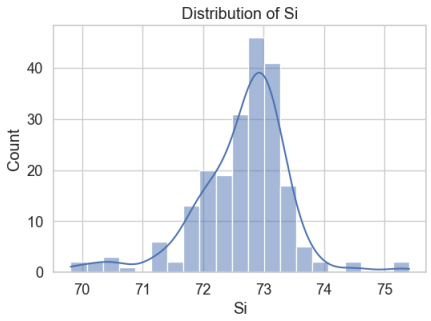

Fig 5: Distribution of Si

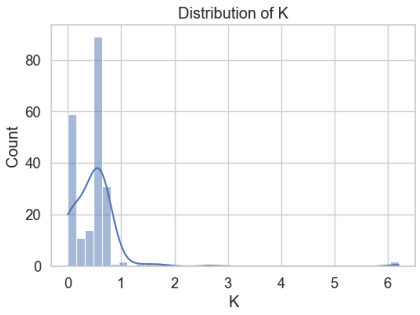

Fig 6: Distribution of K

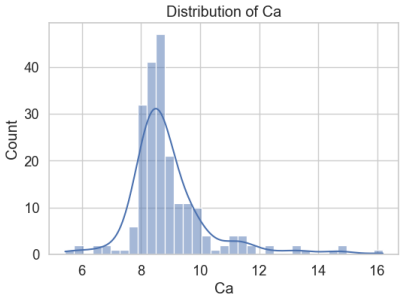

Fig 7: Distribution of Ca

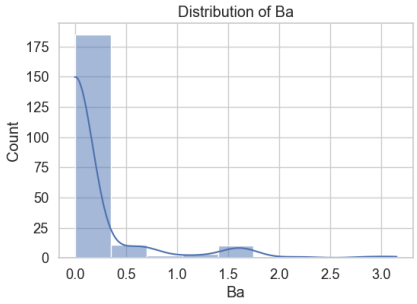

Fig 8: Distribution of Ba

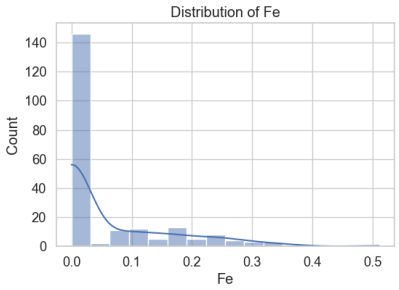

Fig 9: Distribution of Fe

1. **Refractive Index (RI):**

The histogram shows a distribution centred around 1.52, with a noticeable peak just below this value. Most glass samples have a refractive index close to 1.52, indicating it as a consistent feature across different glass types. The distribution is slightly right skewed, suggesting that while the majority of samples cluster around the central value, there are some with higher refractive indices.

2. **Sodium (Na):**

The histogram shows a right-skewed distribution with a peak around 13 weight percent. Most glass samples have sodium content close to this value, indicating it as a prevalent feature across different glass types. The skewness suggests that while most samples have moderate sodium content, there are some with significantly higher or lower values.

3. **Magnesium (Mg):**

The histogram shows a high concentration of values at 0, indicating many glass samples do not contain magnesium. However, a small number of samples have magnesium values around 3 and 4, which can be useful in identifying specific glass types that contain magnesium.

4. **Aluminium (Al):**

The histogram appears to be somewhat normally distributed with a peak around 1. This suggests that aluminium content is consistent among most glass samples, though there are some variations which might help in classification.

5. **Silicon (Si):**

The histogram shows a normal distribution centred around 73. This suggests silicon is a major component in glass and consistent across samples, but small variations can still help in classification.

6. **Potassium (K):**

The histogram shows a high concentration of values at 0, indicating that most glass samples have little to no potassium. A few samples with higher potassium content suggest this could be a distinguishing feature for certain glass types.

7. **Calcium (Ca):**

The histogram shows a right-skewed distribution, with most values between 8 and 10. This indicates that calcium content varies among glass samples, and this variation could help differentiate between different types of glass.

8. **Barium (Ba):**

The histogram shows that barium is present in very few samples and most values are close to 0. This suggests that barium might be a distinguishing feature for a small subset of glass types.

9. **Iron (Fe):**

The histogram is highly right skewed with most values at 0, indicating that iron is present in negligible amounts in most samples. However, the few samples with higher iron content could be used to classify specific types of glass.

Boxplots:

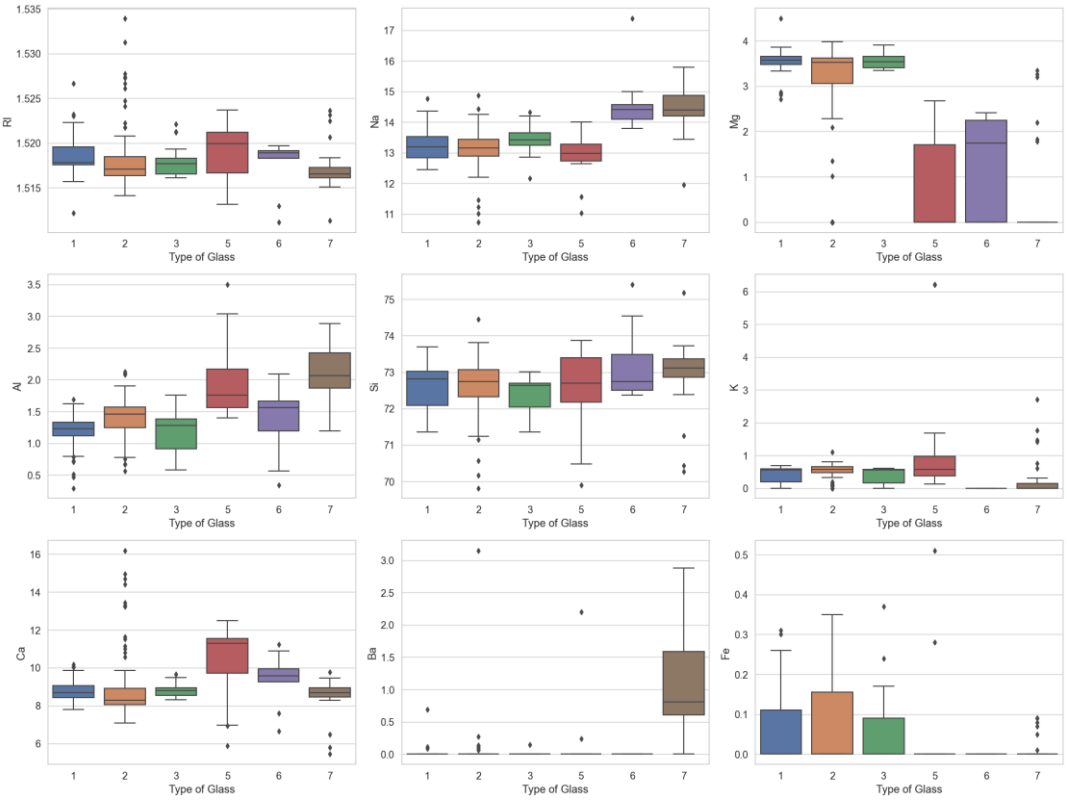

Fig 10: Boxplot of all the features vs Type of Glass

The boxplots provide a detailed comparison of the chemical composition of different types of glass. Each subplot represents the distribution of a specific chemical component across the various glass types, highlighting the variations and similarities.

1. **Refractive Index (RI)**
   The average Refractive Index (RI) is similar across all glass types, centered around 1.52. However, Type 5 (containers) exhibits a wider range and a slightly higher mean RI, suggesting more variability in this glass type.
2. **Sodium (Na)**
   Sodium content varies significantly across glass types. Compared to other types, Types 6 (tableware) and 7 (headlamps) have higher sodium contents. This suggests that the composition of certain varieties of glass contains more sodium, which may be connected to the unique ways in which they are made.
3. **Magnesium (Mg)**
   Magnesium content is relatively high in Types 1, 2, and 3 (building windows float processed, non-float processed, and vehicle windows float processed, respectively). This suggests that magnesium is an important component in the manufacturing of building and vehicle window glasses.

4. **Aluminium (Al):**
   Aluminium content is notably higher in Types 5 (containers) and 7 (headlamps). This higher aluminium content could be related to the need for increased durability and strength in these glass types.
5. **Silicon (Si):**
   With the largest concentration of any element, silicon (Si) serves as the main constituent in all varieties of glass. All varieties of glass have a comparable range of silicon content, demonstrating its regular application in the glass-making process. But because of this homogeneity, silicon by itself cannot tell you anything about the different forms of glass.
6. **Potassium (K):**
   Potassium content does not vary significantly across most glass types, except for Type 5 (containers), which has higher potassium levels. This is likely due to the use of potassium in toughened glass, such as Pyrex items, which require increased strength and thermal resistance [7].
7. **Calcium (Ca):**
   Calcium content is highest in Type 5 (containers). Calcium is commonly used in glass to improve Its durability and chemical resistance, which is essential for containers that need to withstand various stresses [8].
8. **Barium (Ba):**
   Barium content is significantly higher in Type 7 (headlamps). Barium is often used in glass to increase its refractive index and provide a unique optical quality, which is crucial for headlamps [9].
9. **Iron (Fe):**
   Iron is present in very low concentrations across all glass types. However, Types 1, 2, and 3 (building and vehicle windows) show higher values for some samples. Iron is commonly added to glass to produce coloured glasses, which might explain the higher concentrations in these types [10].

Insights from boxplots

The boxplots reveal distinct patterns in the chemical composition of different glass types. For example, Type 5 (containers) and Type 7 (headlamps) have higher concentrations of certain elements (Al, Ca, Ba) that are critical for their specific applications. In contrast, elements like Silicon (Si) and Potassium (K) show less variation across glass types, making them less useful for differentiation. Understanding these patterns is essential for developing accurate classification models and improving forensic glass analysis.

- **Model Performance**
  - **K-Nearest Neighbors (KNN)**
    - **Confusion Matrix:**

```
Confusion Matrix:
[[16  3  0  0  0  0]
 [10 12  0  1  0  0]
 [ 1  3  0  0  0  0]
 [ 0  4  0  2  0  0]
 [ 0  0  0  0  2  1]
 [ 0  1  0  0  0  9]]
```

    - **Classification Report:**

```
Classification Report:
              precision    recall  f1-score   support

           1       0.59      0.84      0.70        19
           2       0.52      0.52      0.52        23
           3       1.00      0.00      0.00         4
           5       0.67      0.33      0.44         6
           6       1.00      0.67      0.80         3
           7       0.90      0.90      0.90        10

    accuracy                           0.63        65
   macro avg       0.78      0.54      0.56        65
weighted avg       0.67      0.63      0.60        65
```

    - **Accuracy Score:**

```
Accuracy Score:
0.6307692307692307
```

- **Decision Tree:**
  - **Confusion Matrix:**

```
Confusion Matrix:
[[12  2  4  0  0  1]
 [ 5 13  3  2  0  0]
 [ 0  1  3  0  0  0]
 [ 0  2  0  2  2  0]
 [ 0  1  0  0  2  0]
 [ 0  0  0  0  0 10]]
```

  - **Classification Report:**

```
Classification Report:
              precision    recall  f1-score   support

           1       0.71      0.63      0.67        19
           2       0.68      0.57      0.62        23
           3       0.30      0.75      0.43         4
           5       0.50      0.33      0.40         6
           6       0.50      0.67      0.57         3
           7       0.91      1.00      0.95        10

    accuracy                           0.65        65
   macro avg       0.60      0.66      0.61        65
weighted avg       0.68      0.65      0.65        65
```

  - **Accuracy Score:**

```
Accuracy Score:
0.6461538461538462
```

- **Hyperparameter Tuning:**
  - **KNN Hyperparameter Tuning Results**

```
Best Parameters from Grid Search:
{'n_neighbors': 1}

Confusion Matrix for Best Model:
[[16  2  1  0  0  0]
 [ 6 15  1  0  1  0]
 [ 1  1  2  0  0  0]
 [ 0  4  0  2  0  0]
 [ 0  0  0  0  3  0]
 [ 0  0  0  0  1  9]]

Classification Report for Best Model:
              precision    recall  f1-score   support

           1       0.70      0.84      0.76        19
           2       0.68      0.65      0.67        23
           3       0.50      0.50      0.50         4
           5       1.00      0.33      0.50         6
           6       0.60      1.00      0.75         3
           7       1.00      0.90      0.95        10

    accuracy                           0.72        65
   macro avg       0.75      0.70      0.69        65
weighted avg       0.75      0.72      0.72        65


Accuracy Score for Best Model:
0.7230769230769231
```

  - **Decision Tree Hyperparameter Tuning Results**

```
Confusion Matrix:
[[16  2  0  0  1  0]
 [ 7 11  2  2  1  0]
 [ 2  1  1  0  0  0]
 [ 0  0  0  4  2  0]
 [ 0  0  0  0  3  0]
 [ 1  0  0  0  0  9]]

Classification Report:
              precision    recall  f1-score   support

           1       0.62      0.84      0.71        19
           2       0.79      0.48      0.59        23
           3       0.33      0.25      0.29         4
           5       0.67      0.67      0.67         6
           6       0.43      1.00      0.60         3
           7       1.00      0.90      0.95        10

    accuracy                           0.68        65
   macro avg       0.64      0.69      0.63        65
weighted avg       0.71      0.68      0.67        65


Accuracy Score:
0.676923076923077
```

Discussion

This study showed that glass types can be efficiently classified using machine learning models, particularly K-Nearest Neighbours (KNN) and Decision Tree, based on their chemical composition. The results validate the usefulness of these models and data visualisations and fill in important knowledge gaps in forensic glass analysis.

Main Argument

The primary thesis of the study is that glass types can be reliably distinguished using KNN and Decision Tree classifiers based on their chemical properties, and that visualisations facilitate understanding of the various elements' roles in glass classification.

Model Performance:

- **K Nearest Neighbors (KNN):**
  Achieved an accuracy of 72%. The optimal k value, determined through hyperparameter tuning, enhanced performance and addressed the knowledge gap regarding optimal configurations for forensic glass classification.

- **Decision Tree:**
  Achieved an accuracy of 67%. Parameter tuning improved the model's performance, addressing the gap in effective Decision Tree settings for glass classification.

Insights from data visualization

- **Chemical Composition**
  Certain elements, such as Barium (Ba) , Iron (Fe) and magnesium (Mg), varied amongst glass types, which was helpful for classification. Refractive Index (RI) and Sodium (Na), on the other hand, showed less variation, suggesting that these features had less differentiation value.

- **Glass Type Distribution:**
  The most common types were Types 1 and 2 (building windows), which corresponded to actual forensic situations. This emphasises how crucial a representative dataset is.

Research Question Answered:

The study demonstrated the usefulness of visual data analysis in understanding elemental roles and successfully addressed how machine learning can classify different types of glass based on chemical composition.

Conclusion

The accuracy with which glass types can be distinguished based on their chemical composition using K-Nearest Neighbours (KNN) and Decision Tree classifiers has been demonstrated in this study, with the KNN model slightly outperforming the Decision Tree. Through detailed data visualizations, we uncovered significant insights into the elemental composition of various glass types, particularly highlighting the variability in Magnesium (Mg) , Barium (Ba), Iron (Fe) across different glass samples. In addition to improving our knowledge of forensic glass analysis, these results highlight how crucial representative datasets and well-optimized model parameters are to obtaining high classification accuracy. These findings have broader applications in domains like material science and quality assurance in the glass industry, where accurate glass type classification is essential. Future research should focus on addressing class imbalances, exploring advanced algorithms, and implementing sophisticated feature engineering techniques. Furthermore, this work raises new research questions, such as how trace elements might affect glass classification and how to create hybrid models that combine multiple algorithms for increased accuracy. This report, taken as a whole, offers a solid basis for future developments in the field of forensic glass analysis and beyond.

REFERENCES

[1]: Andrew Mauer, 2013: Forensic Glass Analysis, https://study.com/academy/lesson/glass-as-forensic-evidence-purpose-collection-preservation.html#:~:text=Two%20methods%20for%20identifying%20glass,which%20light%20passes%20through%20glass.

[2]: Glass Identification, UC Irvine Machine Learning Repository: https://archive.ics.uci.edu/dataset/42/glass+identification

[3]: StandardScalar: https://scikit-learn.org/stable/modules/generated/sklearn.preprocessing.StandardScaler.html

[4]: What is the k-nearest neighbors(KNN) algorithm: https://www.ibm.com/topics/knn#:~:text=The%20k%2Dnearest%20neighbors%20(KNN)%20algorithm%20is,regression%20classifiers%20used%20in%20machine%20learning%20today.

[5]: Mukesh Mithrakumar, 2019, How to tune a Decision Tree? : https://towardsdatascience.com/how-to-tune-a-decision-tree-f03721801680

[6]: Optimal Tuning Parameter: https://www.ritchieng.com/machine-learning-efficiently-search-tuning-param/#:~:text=3.%20More%20efficient%20parameter%20tuning%20using%20GridSearchCV

[7]: The Secret of Tough Glass: Ion Exchange: https://www.corning.com/au/en/innovation/the-glass-age/science-of-glass/the-secret-of-tough-glass-ion-exchange.html#:~:text=Potassium%20ions%20(electronically%20charged%20particles,that%20forms%20a%20tough%20surface.

[8]: Plastics Industry And The Role Of Calcium Carbonate In Production: https://globestonehills.com/plastics-industry-and-the-role-of-calcium-carbonate-in-production/#:~:text=Calcium%20carbonate%20is%20used%20as,of%20products%20and%20prevent%20damage.&text=Calcium%20carbonate%20is%20used%20as%20a%20reinforcing%20agent%20in%20the,and%20hardness%20to%20plastic%20products.

[9]: Barium carbonate - Witherite: Properties, Uses, and Safety: https://www.ceramic-glazes.com/Barium-carbonate-Witherite

[10] Low iron float glass and its preparation method and use,2009: https://patents.google.com/patent/KR100983476B1/en