Assessment Report

on

“Classify Vegetables Based on Nutritional Content”

submitted as partial fulfillment for the award of

BACHELOR OF TECHNOLOGY DEGREE

SESSION 2024-25

in

CSE(AIML)

By

Name: Aman Yadav

Roll Number: 202401100400028

Section: A

Under the supervision of

“BIKKI KUMAR”

KIET Group of Institutions, Ghaziabad

April, 2025

---

1. **Introduction**
   Classification and Segmentation of Vegetables Based on Nutritional Content

---

2. Problem Statement

In the field of nutrition science and diet planning, it is essential to classify vegetables based on their nutritional content to support healthy eating, dietary analysis, and food recommendations. However, due to the variety and overlap in nutritional values among different vegetable types, manually classifying and grouping vegetables can be inconsistent and inefficient.

This project aims to develop a machine learning-based system that can **automatically classify vegetables into categories** such as *leafy, root, cruciferous*, etc., based on their nutritional features. Additionally, it explores **unsupervised learning techniques** to segment and cluster vegetables, revealing hidden patterns or similarities that are not captured by traditional classification methods.

By combining both supervised and unsupervised approaches, the project provides a robust solution for organizing, analyzing, and visualizing vegetable data, which can be valuable for dieticians, nutritionists, app developers, and food recommendation systems.

---

3. Objectives

To build a machine learning system that can:

- **Classify** vegetables into categories (e.g., leafy, root, etc.) based on their nutritional values.
- **Cluster/segment** vegetables using unsupervised learning to explore natural groupings.

---

4. Methodology

1. Preprocessing:

- **Encoded labels using LabelEncoder.**
- **Standardized feature values using StandardScaler.**

2. Classification:

- **Model: RandomForestClassifier.**
- **Metrics: Accuracy, Precision, Recall**
- **Visualization: Confusion Matrix heatmap**

3. Clustering & Segmentation:

- **Model: KMeans clustering.**
- **Visualization: PCA-based 2D scatter plot showing segmented clusters.**

---

5. Code

```
#Import Required Libraries
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import LabelEncoder, StandardScaler
from sklearn.ensemble import RandomForestClassifier
```

```
from sklearn.metrics import confusion_matrix, classification_report, accuracy_score,
precision_score, recall_score

from sklearn.cluster import KMeans

from sklearn.decomposition import PCA


# Load Dataset from Google Drive

df = pd.read_csv('/content/drive/MyDrive/Colab
Notebooks/vegetable_nutrition_dataset.csv')

print("✅ Dataset Loaded Successfully!")


# Step 4: Classification


## Prepare data

X = df.drop(["Name", "Category"], axis=1)

y = df["Category"]


## Encode target labels

le = LabelEncoder()

y_encoded = le.fit_transform(y)


## Scale features

scaler = StandardScaler()

X_scaled = scaler.fit_transform(X)
```

```
## Split into train/test
X_train, X_test, y_train, y_test = train_test_split(X_scaled, y_encoded, test_size=0.3, random_state=42)

## Train classifier
clf = RandomForestClassifier(random_state=42)
clf.fit(X_train, y_train)

## Predict and Evaluate
y_pred = clf.predict(X_test)

print("\n🔍 Evaluation Metrics")
print(f"Accuracy: {accuracy_score(y_test, y_pred):.2f}")
print(f"Precision: {precision_score(y_test, y_pred, average='weighted'):.2f}")
print(f"Recall: {recall_score(y_test, y_pred, average='weighted'):.2f}")

print("\n📊 Classification Report:")
print(classification_report(y_test, y_pred, target_names=le.classes_))

## Confusion Matrix Heatmap
plt.figure(figsize=(8,6))
sns.heatmap(confusion_matrix(y_test, y_pred),
            annot=True, fmt='d', cmap='Blues',
```

```
        xticklabels=le.classes_, yticklabels=le.classes_)
plt.title("Confusion Matrix")
plt.xlabel("Predicted")
plt.ylabel("Actual")
plt.show()

# Step 5: Clustering and Segmentation

## Apply KMeans
kmeans = KMeans(n_clusters=4, random_state=42)
cluster_labels = kmeans.fit_predict(X_scaled)

## Attach clusters to dataframe
df['Cluster'] = cluster_labels

## PCA for visualization
pca = PCA(n_components=2)
X_pca = pca.fit_transform(X_scaled)

## Plot clusters
plt.figure(figsize=(8,6))
sns.scatterplot(x=X_pca[:,0], y=X_pca[:,1], hue=cluster_labels, palette='Set2')
plt.title("K-Means Clustering Segmentation of Vegetables")
```

plt.xlabel("PCA Component 1")

plt.ylabel("PCA Component 2")

plt.legend(title="Cluster")

plt.show()

---

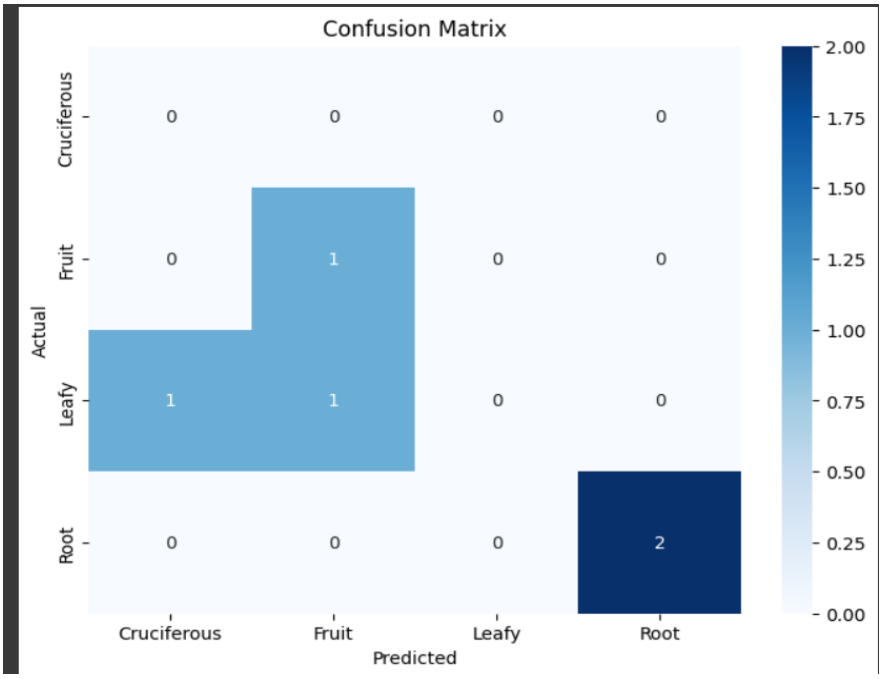
6. Results and Analysis

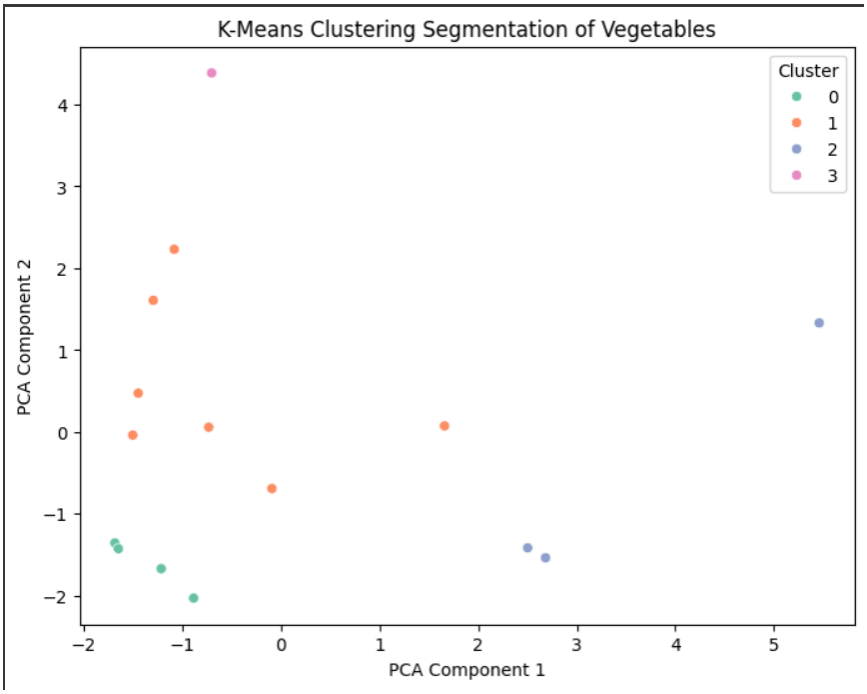
```
🔍 Evaluation Metrics
Accuracy: 0.60
Precision: 0.50
Recall: 0.60

📊 Classification Report:
              precision    recall  f1-score   support

 Cruciferous       0.00      0.00      0.00         0
       Fruit       0.50      1.00      0.67         1
       Leafy       0.00      0.00      0.00         2
        Root       1.00      1.00      1.00         2

    accuracy                           0.60         5
   macro avg       0.38      0.50      0.42         5
weighted avg       0.50      0.60      0.53         5
```





---

10. References

- scikit-learn documentation

- pandas documentation
- Seaborn visualization library
- Research articles on credit risk prediction
