Thakur Educational Trust's (Regd.)
THAKUR COLLEGE OF SCIENCE & COMMERCE
AUTONOMOUS COLLEGE, PERMANENTLY AFFILIATED TO UNIVERSITY OF MUMBAI
NAAC Accredited Grade 'A' (3rd Cycle) & ISO 9001: 2015 (Certified)
Best College Award by University of Mumbai for the Year 2018-2019

THAKUR COLLEGE OF SCIENCE AND COMMERCE

KANDIVALI (EAST), MUMBAI.

A

PROJECT

REPORT

ON

Apple Quality Classification

FOR

Thakur College of Science & Commerce

By

Bhavesh .S. Choubey

Submitted in partial fulfilment of

Bachelors of Science (Computer Science)

Thakur Degree College of Science and

Commerce Kandivali (East), Mumbai.

ACADEMIC YEAR 2023-2024

Certification From College:

[COMPUTER SCIENCE DEPARTMENT**]**

(2023-2024)

Certificate of Approval

This is to certify that the project work entitled **"Apple Quality Classification"** is prepared by **066 Bhavesh Sushilkumar Choubey** a student of **"Third Year Bachelor of Science (Computer Science)"** course of University of Mumbai, which is conducted by our College.

This is the original study work and important sources used have been duly acknowledged in the report. The report is submitted in partial fulfilment of B.Sc. (Computer Science) course as per rules of University of Mumbai.

Ms. Anjali Wagh — Mr. Ashish Trivedi

Project Guide. — **Head of Department.**

____

External Examiner

Date: ______________

Implementation Of Project

Apple Quality Classification

Topic	Expected Date	Completion Date
Preliminary		
System Analysis		
System Design		
System Coding		
System Implementation		
Report Submission		

Bhavesh .S. Choubey

Acknowledgement

Achievement is finding out what you would be doing rather than what you have to do. It is not until you undertake such a project that you realize how much effort and hard work it really is, what are your capabilities and how well you can present yourself or other things. It gives me immense pleasure to present this report towards the fulfilment of my project.

It has been rightly said that we are built on the shoulder of others. For everything I have achieved, the credit goes to all those who helped me to complete this project successfully.

I take this opportunity to express my profound gratitude to management of Thakur Degree College of Science & Commerce for giving me this opportunity to accomplish this project work.

I am very much thankful to **Mrs. C. T. Chakraborty** – Principal of Thakur College for their kind co-operation in the completion of my project.

A special vote of thanks to our HOD **Mr. Ashish Trivedi** and to my project guide **Ms. Anjali Wagh** Thanks teachers **Mr. Mahesh Gurunani** for helping and guiding me throughout my project.

Bhavesh .S. Choubey

Description

The Apple Quality Grouping Framework Task plans to lay out a normalized strategy for reviewing and characterizing apples in view of key quality credits. This venture includes characterizing standards like size, variety, surface, flavor, Ripeness, Sweetness, Acidity, etc. and appearance to make a uniform evaluating framework relevant across various locales and markets. Through preparing programs and instructive assets, partners in the apple production network will be outfitted with the information and devices important to survey apple quality precisely. Combination of innovation will additionally upgrade proficiency and precision in evaluating processes. Consistence with important guidelines and cooperation among industry partners are necessary to the progress of this drive, which eventually looks to guarantee consistency, decency, and purchaser fulfillment in the apple market.

The Apple Quality Order Framework Undertaking is a complete undertaking pointed toward normalizing the evaluating and characterization of apples in view of different quality credits. This drive tries to foster a uniform framework that sorts apples into various grades or classes, considering variables like size, variety, surface, flavor, and generally appearance. The undertaking includes the foundation of explicit measures and boundaries for surveying apple quality, as well as the making of a progressive evaluating framework with various grades or classes. Preparing programs and instructive materials will be given to partners to guarantee precise reviewing rehearses, while mechanical arrangements will be coordinated to improve effectiveness and exactness in evaluating processes. Administrative consistence and industry joint effort are additionally key parts of the undertaking, guaranteeing arrangement with significant norms and encouraging commitment across the apple store network. By and large, the Apple Quality Arrangement Framework Undertaking expects to further develop consistency, decency, and purchaser trust in the evaluating and grouping of apples.

Index

Chapter 1
Introduction

The Apple Quality Order model utilizing AI is an information driven approach pointed toward mechanizing the most common way of evaluating and sorting apples in view of different quality credits. This model use AI calculations to break down highlights like size, variety, surface, and shape extricated from pictures or sensor information to arrange apples into various grades or classes.

Via preparing on a named dataset containing instances of apples with known quality grades, the AI model figures out how to perceive examples and connections between the info highlights and the relating quality characterizations. This empowers the model to make exact expectations about the quality grade of concealed apples in view of their attributes.

The execution of such a model offers a few advantages to the apple business, remembering expanded effectiveness for reviewing processes, further developed consistency in quality evaluation, and likely expense reserve funds by diminishing dependence on difficult work. Also, it can assist with smoothing out store network activities and improve purchaser certainty by guaranteeing that main apples satisfying determined quality guidelines arrive at the market.

Generally speaking, the Apple Quality Order model involving AI addresses a promising headway in the organic product evaluating industry, offering an information driven way to deal with upgrade quality appraisal and improve the general worth chain of apple creation and dispersion.

This model holds critical potential for smoothing out the evaluating system in the apple business, lessening dependence on abstract human judgment and empowering more reliable and objective appraisals. With progressions in AI strategies and admittance to enormous datasets, this approach offers potential chances to improve effectiveness and precision in apple quality arrangement, eventually helping cultivators, packers, merchants, and shoppers the same.

Chapter 2
Review of Literature

The writing on Apple Quality Grouping models using AI, especially Strategic Relapse and Choice Tree Classifier, exhibits the developing interest in utilizing information driven ways to deal with computerize and upgrade natural product reviewing processes. Here is a rundown of key discoveries from existing examinations:

Calculated Relapse for Apple Quality Grouping:

Strategic Relapse has been generally investigated for natural product quality arrangement because of its effortlessness, interpretability, and effectiveness in taking care of double grouping assignments.

Specialists have used Calculated Relapse to anticipate the presence or nonattendance of imperfections in apples in light of elements like size, variety, and surface.

Review have exhibited the viability of Calculated Relapse models in recognizing different quality grades of apples, accomplishing high precision rates when prepared on fittingly named datasets.

Calculated Relapse models are especially reasonable for situations where the connection between input highlights and the objective variable (quality grade) is roughly direct and can be addressed by a sigmoid capability.

Decision Tree Classifier for Apple Quality Characterization:

Decision Tree Classifier is one more famous decision for organic product quality characterization, known for its capacity to deal with non-straight connections and give interpretable choice principles.

Scientists have utilized Decision Tree models to order apples into various quality grades in view of different highlights, including size, shape, variety, and surface deformities.

Decision Trees offer benefits, for example, simplicity of translation, natural choice principles, and the capacity to deal with both mathematical and all out information without requiring broad preprocessing.

Outfit strategies like Irregular Backwoods, which utilize numerous Decision Trees to work on prescient execution, have additionally been applied to apple quality grouping assignments, accomplishing higher exactness rates contrasted with individual Decision Tree models.

Chapter 3

Data Collection and Preprocessing

Information assortment and preprocessing are essential moves toward fostering an AI model for apple quality characterization. This is a layout of the way these means may be drawn nearer:

Data Collection:

Get a different dataset of apple pictures or tests addressing different quality grades. This dataset ought to preferably cover different traits like size, variety, surface, and the presence of deformities.
Gather information from numerous sources, including plantations, markets, or data sets, to guarantee representativeness and variety in the dataset.
Mark every apple test with its comparing quality grade or class, guaranteeing precise and predictable explanations.
Consider factors, for example, lighting conditions, camera points, and picture goal during information assortment to limit inconstancy and guarantee information quality.

Data Preprocessing:

Picture Handling: If involving pictures as info information, preprocess them to upgrade highlights pertinent to apple quality arrangement. This might include resizing, editing, or normalizing pictures to a normalized design.
Highlight Extraction: Concentrate applicable elements from the information that catch significant qualities of apple quality, like variety histograms, surface descriptors, or shape credits. Highlight extraction methods like Histogram of Situated Inclinations (Hoard) or Nearby Paired Examples (LBP) might be utilized.
Information Cleaning: Eliminate any immaterial or boisterous data of interest from the dataset to work on model execution and speculation. This might include sifting through exceptions, rectifying blunders, or taking care of missing qualities.
Standardization: Standardize mathematical elements to guarantee they have comparable scales, keeping specific highlights from overwhelming the model preparation process.
Information Expansion: Expand the dataset by applying changes like revolution, flipping, or adding commotion to build the variety of preparing tests and work on model vigor.
Adjusting Classes: Address class awkwardness issues by applying procedures, for example, oversampling, undersampling, or creating engineered tests to guarantee equivalent portrayal of various quality grades in the dataset.

3.1. Dataset Apple_Quality.csv file

This dataset contains information about various attributes of a set of fruits, providing insights into their characteristics. The dataset includes details such as fruit ID, size, weight, sweetness, crunchiness, juiciness, ripeness, acidity, and quality.

Key Features:

- **A_id**: *Unique identifier for each fruit*
- **Size**: *Size of the fruit*
- **Weight**: *Weight of the fruit*
- **Sweetness**: *Degree of sweetness of the fruit*
- **Crunchiness**: *Texture indicating the crunchiness of the fruit*
- **Juiciness**: *Level of juiciness of the fruit*
- **Ripeness**: *Stage of ripeness of the fruit*
- **Acidity**: *Acidity level of the fruit*
- **Quality**: *Overall quality of the fruit*

Potential Use Cases:

- **Fruit Classification:** Develop a classification model to categorize fruits based on their features.
- **Quality Prediction:** Build a model to predict the quality rating of fruits using various attributes.

Chapter 4
Feature Selection and Engineering

Feature Selection:

A_id: This attribute likely serves as an identifier for individual apple samples and may not contribute directly to the classification task. It can be excluded from the feature set.
Size: Size can be a crucial feature in determining apple quality, as larger apples are often perceived as higher quality. It can be retained as a feature.
Weight: Weight is another important attribute related to apple quality, with heavier apples often considered more desirable. It can also be retained as a feature.
Sweetness: Sweetness is a key aspect of apple taste and can influence perceived quality. It should definitely be retained as a feature.
Crunchiness: Crunchiness reflects the texture of the apple, which is an important quality factor. It should be retained as a feature.
Juiciness: Juiciness contributes to the overall eating experience of the apple and can affect perceived quality. It should be retained as a feature.
Ripeness: Ripeness is crucial for determining apple quality, as overly ripe or underripe apples may be considered lower quality. It should be retained as a feature.
Acidity: Acidity contributes to the flavor profile of the apple and can influence perceived quality. It should be retained as a feature.

Feature Engineering:

Interaction Terms: Explore potential interactions between features that may provide additional information for classification. For example, the combination of sweetness and acidity might create a balance that affects perceived quality.
Derived Features: Create derived features that capture specific characteristics of apples. For instance, the ratio of sweetness to acidity could represent the overall flavor balance.
Normalization: Normalize numerical features such as size, weight, sweetness, crunchiness, juiciness, and acidity to ensure they have similar scales and contribute evenly to the model.
Encoding Categorical Features: If any categorical features are present (e.g., quality grades), encode them into numerical values using techniques like one-hot encoding.
Feature Scaling: Scale features to a similar range to prevent certain features from dominating the model training process. Techniques like min-max scaling or standardization can be applied.

4.1. Libraries used in Model:-

1. Numpy
2. Pandas
3. Seaborn
4. Matplotlib
5. Ydata_profiling
    - ProfileReport
6. Sklearn.model_selection
    - Train_test_split
7. Sklearn.tree
    - DecisionTreeClassifier
8. Sklearn.linear_model
    - LogisticRegression
9. Sklearn.metrices
    - Accuracy_score
    - Confusion_matrix
    - Classification_report

%matplotlib inline

Chapter 5
Machine Learning Models

An AI model is a numerical portrayal or computational calculation that gains designs from information to play out a particular undertaking, like making expectations, recognizing examples, or simply deciding. These models are developed in light of numerical standards and calculations, and they gain iteratively from the information they are given.

The center thought behind AI models is to empower PCs to gain from information and decide or expectations without being expressly customized to do as such. This is accomplished by presenting the model to a dataset containing instances of information sources and their comparing yields, permitting the model to become familiar with the hidden examples or connections between the sources of info and results.

When prepared, an AI model can be utilized to handle new, inconspicuous information and give expectations or choices in light of the examples it has learned. The exhibition of an AI model is commonly assessed in light of its capacity to precisely sum up to new, concealed information, which is surveyed utilizing different assessment measurements well defined for the job that needs to be done.

By and large, AI models act as amazing assets for taking care of complicated issues across different spaces, including picture acknowledgment, regular language handling, proposal frameworks, and numerous others. They assume a basic part in empowering PCs to mechanize errands, separate experiences from information, and pursue clever choices.

Model - LR, Accuracy - 0.7525

Model - CT, Accuracy - 0.76875

5.1. LogisticRegression() model:-

Logistic regression is a supervised machine learning algorithm that achieves twofold grouping errands by foreseeing the likelihood of a result, occasion, or perception. The model conveys a paired or dichotomous result restricted to two potential results: yes/no, 0/1, or valid/misleading.

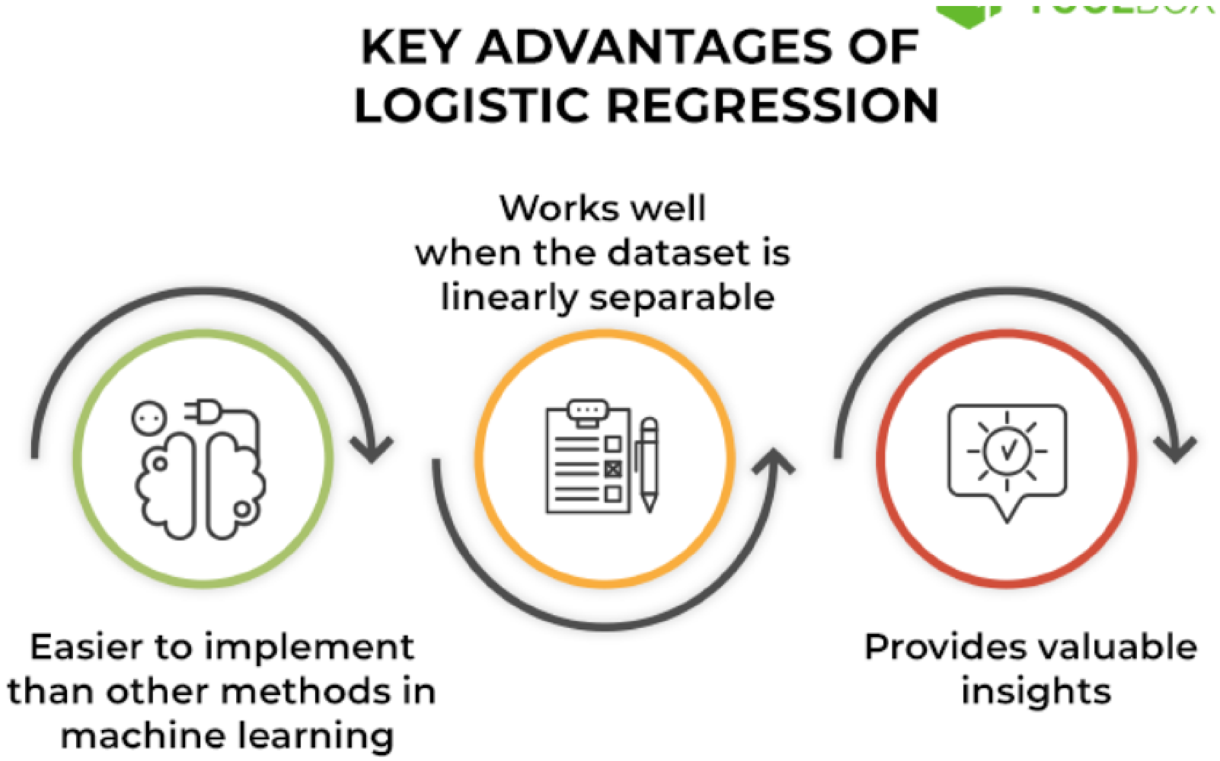
Logical regression examines the connection between at least one free factors and characterizes information into discrete classes. It is widely utilized in prescient displaying, where the model gauges the numerical likelihood of regardless of whether an example has a place with a particular classification.

For instance, 0 - addresses a negative class; 1 - addresses a positive class. Calculated relapse is normally utilized in double characterization issues where the result variable uncovers both of the two classes (0 and 1).

A few instances of such characterizations and cases where the paired reaction is normal or suggested are:

1. Determine the probability of heart attacks:
2. Possibility of enrolling into a university:
3. Detecting spam messages:

Advantages of Logistic Regression:



5.2. Decision Tree Classifier:-

Decision Tree is a Supervised learning method that can be utilized for both order and Relapse issues, yet for the most part it is liked for tackling Grouping issues. It is a tree-organized classifier, where inward hubs address the elements of a dataset, branches address the choice standards and each leaf hub addresses the result.

In a Decision tree, there are two hubs, which are the Decision Hub and Leaf Hub. Decision hubs are utilized to go with any choice and have various branches, while Leaf hubs are the result of those choices and contain no further branches.

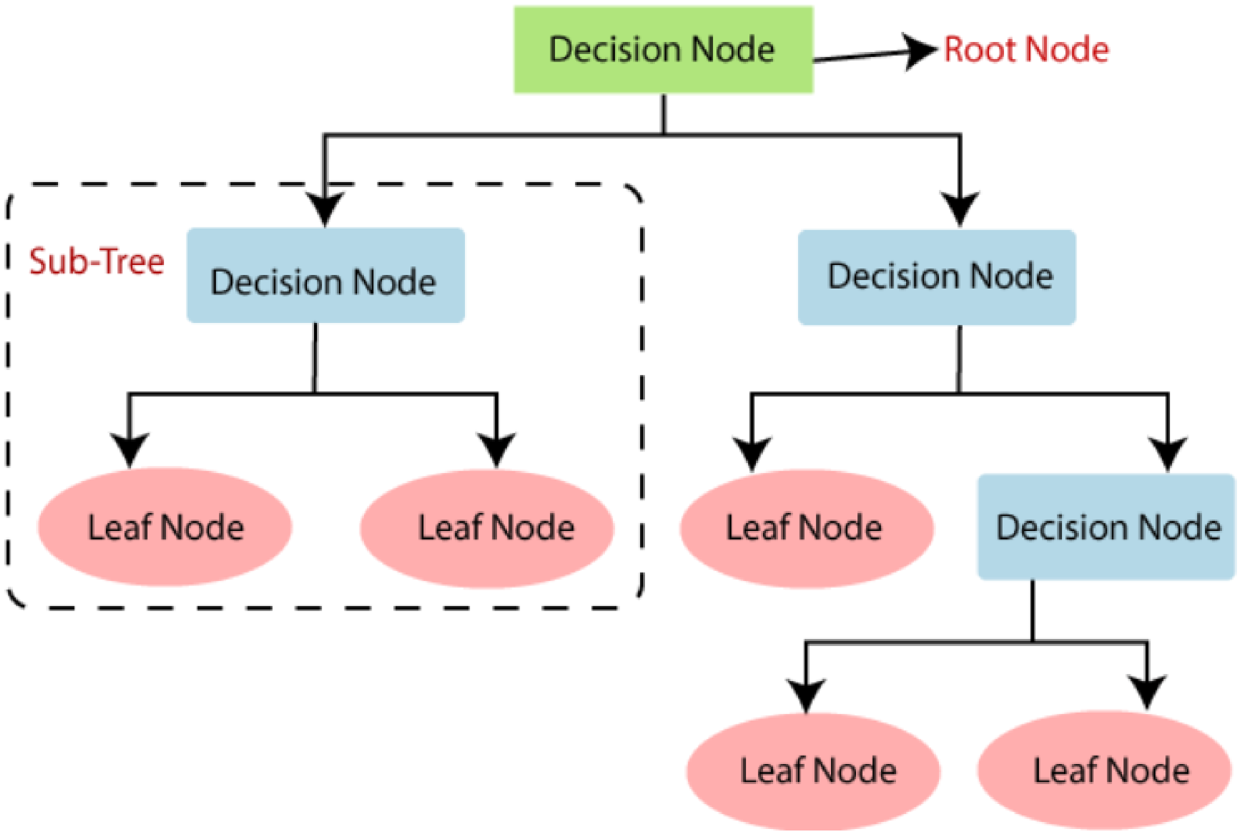
The choices or the test are performed based on highlights of the given dataset.

It is a graphical representation for getting every one of the potential answers for an issue/choice in view of given conditions.

It is known as a choice tree on the grounds that, like a tree, it begins with the root hub, which develops further branches and builds a tree-like construction.

To construct a tree, we utilize the Truck calculation, which represents Characterization and Relapse Tree calculation.

A choice tree essentially poses an inquiry, and in view of the response (Yes/No), it further split the tree into subtrees.



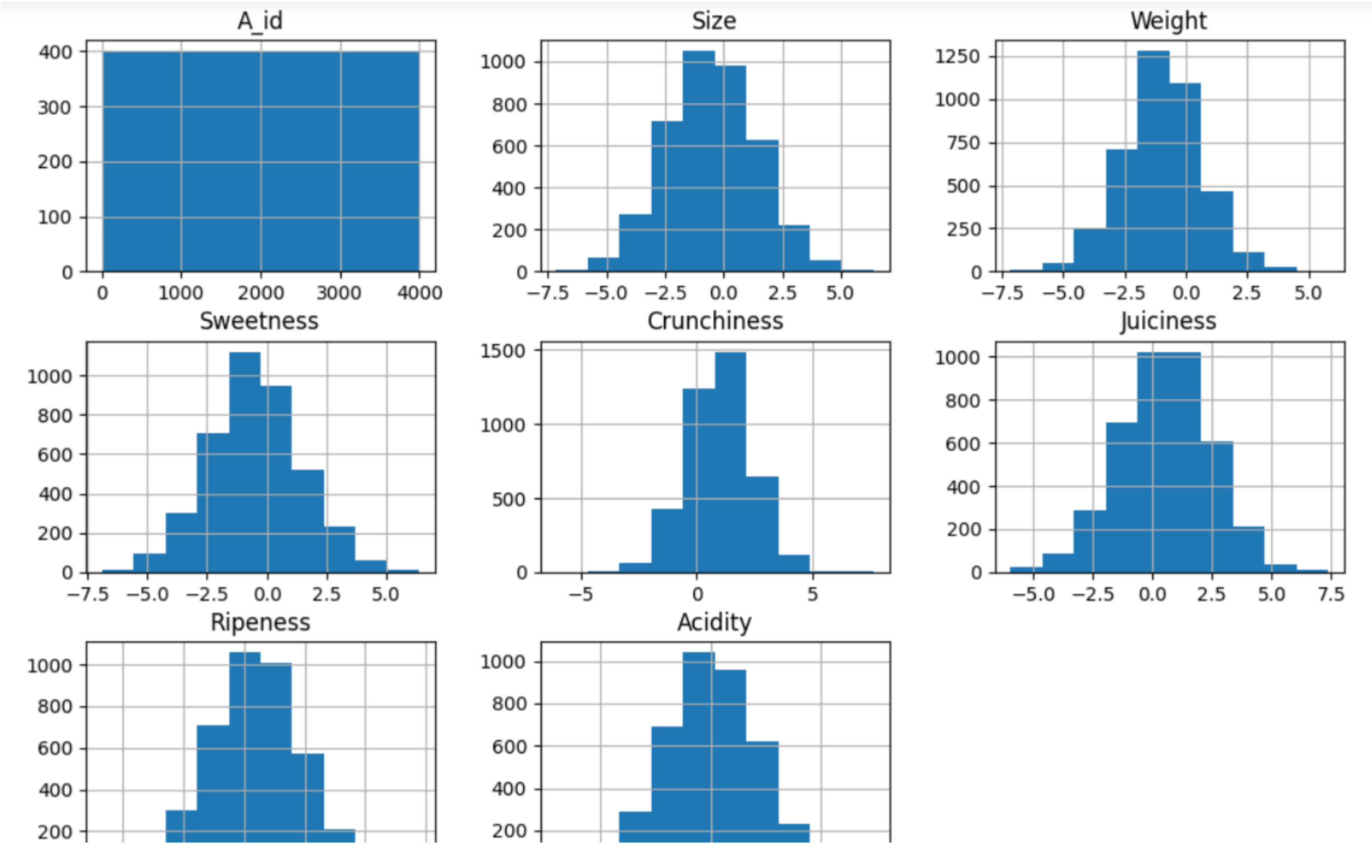
Chapter 6
Model Training and Evaluation

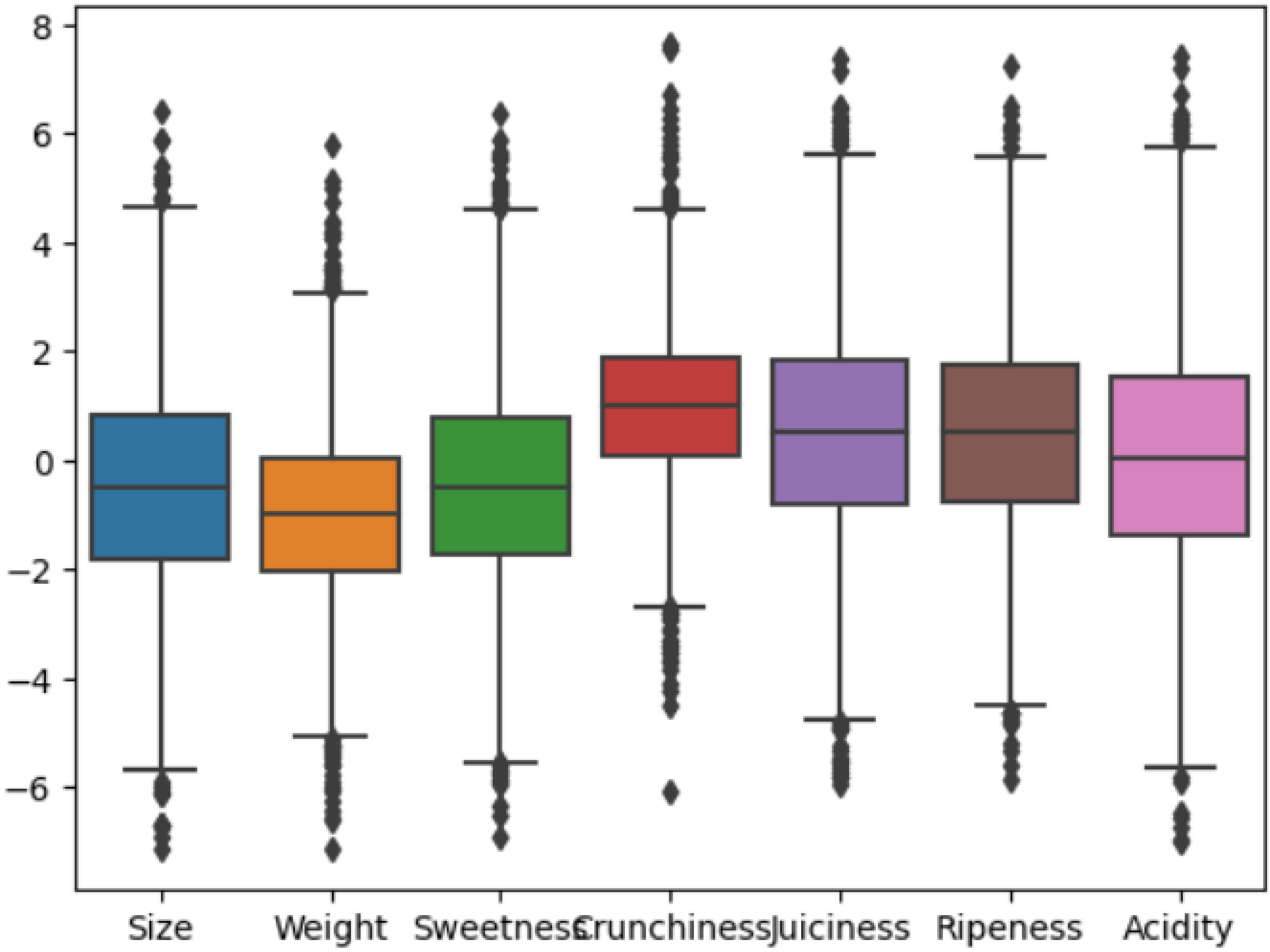
Model Training:

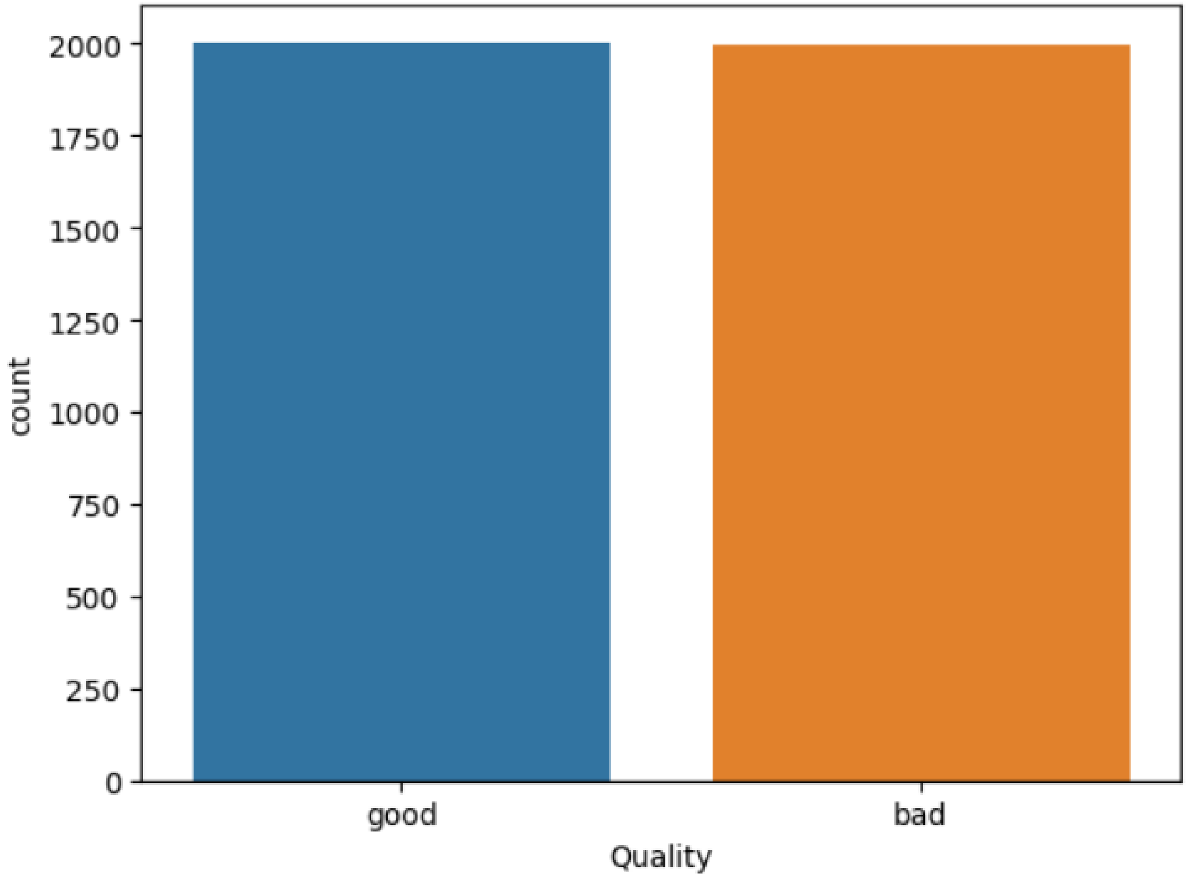
Data Preparation: Prior to preparing the model, the dataset should be partitioned into preparing, approval, and test sets. The preparation set is utilized to prepare the model, the approval set is utilized to tune hyperparameters and screen execution during preparing, and the test set is utilized to assess the last exhibition of the prepared model.
Model Initialization: Instate the model with arbitrary loads or boundaries.
Forward Spread: Pass the preparation information through the model to get expectations.
Misfortune Estimation: Work out the misfortune (blunder) between the model's expectations and the genuine objective qualities utilizing a misfortune capability suitable for the undertaking (e.g., mean squared mistake for relapse, cross-entropy misfortune for grouping).
Backpropagation: Use backpropagation to register the slopes of the misfortune concerning the model's boundaries.

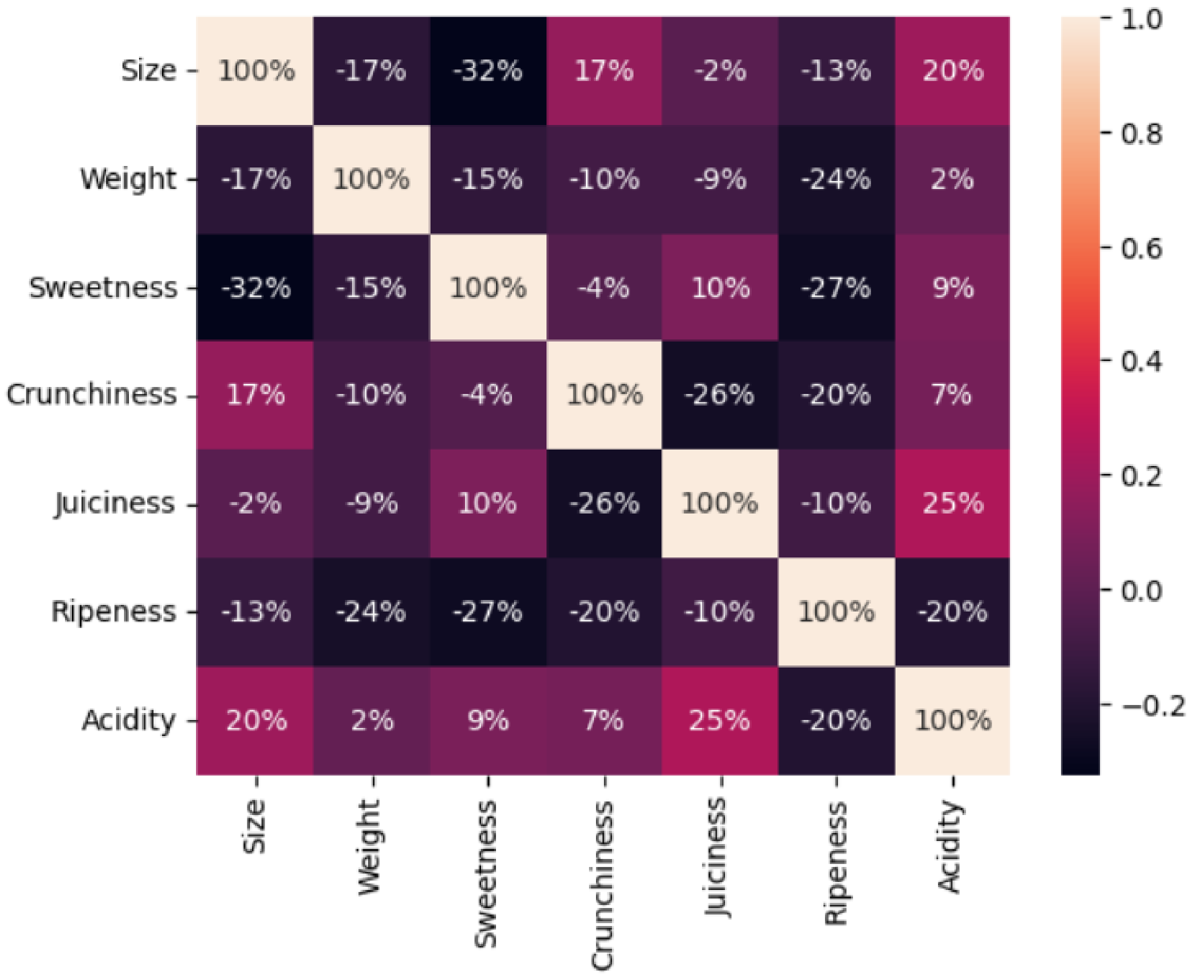
Model Evaluation:

Validation Set Evaluation: Assess the model's presentation on the approval set intermittently during preparing to screen for overfitting and select the best-performing model in light of approval measurements.
Test Set Assessment: When preparing is finished, assess the last model on the test set to evaluate its speculation execution. This gives an unprejudiced gauge of the model's exhibition on inconspicuous information.
Execution Measurements: Work out important execution measurements in light of the job needing to be done (e.g., exactness, accuracy, review, F1-score for grouping; mean squared mistake, R-squared for relapse).
Representation: Envision model execution measurements, for example, expectations to learn and adapt, ROC bends, or accuracy review bends, to acquire bits of knowledge into the model's way of behaving.


A_id
Size
Weight
Sweetness
Crunchiness
Juiciness
Ripeness
Acidity


Size
Weight
Sweetness
Crunchiness
Juiciness
Ripeness
Acidity


2000
1750
1500
1250
1000
750
500
250
0
count
good
bad
Quality


Size
Weight
Sweetness
Crunchiness
Juiciness
Ripeness
Acidity
100% -17% -32% 17% -2% -13% 20%
-17% 100% -15% -10% -9% -24% 2%
-32% -15% 100% -4% 10% -27% 9%
17% -10% -4% 100% -26% -20% 7%
-2% -9% 10% -26% 100% -10% 25%
-13% -24% -27% -20% -10% 100% -20%
20% 2% 9% 7% 25% -20% 100%
1.0
0.8
0.6
0.4
0.2
0.0
−0.2

Chapter 7

Results and Discussion

```
df = pd.read_csv(path)
df.head()
```

Out[3]:

	A_id	Size	Weight	Sweetness	Crunchiness	Juiciness	Ripeness	Acidity	Quality
0	0.0	-3.970049	-2.512336	5.346330	-1.012009	1.844900	0.329840	-0.491590483	good
1	1.0	-1.195217	-2.839257	3.664059	1.588232	0.853286	0.867530	-0.722809367	good
2	2.0	-0.292024	-1.351282	-1.738429	-0.342616	2.838636	-0.038033	2.621636473	bad
3	3.0	-0.657196	-2.271627	1.324874	-0.097875	3.637970	-3.413761	0.790723217	good
4	4.0	1.364217	-1.296612	-0.384658	-0.553006	3.030874	-1.303849	0.501984036	good

In [5]:
```
df = df.dropna()
df["Acidity"] = df["Acidity"].astype("float64")
```

Exploratory Analysis

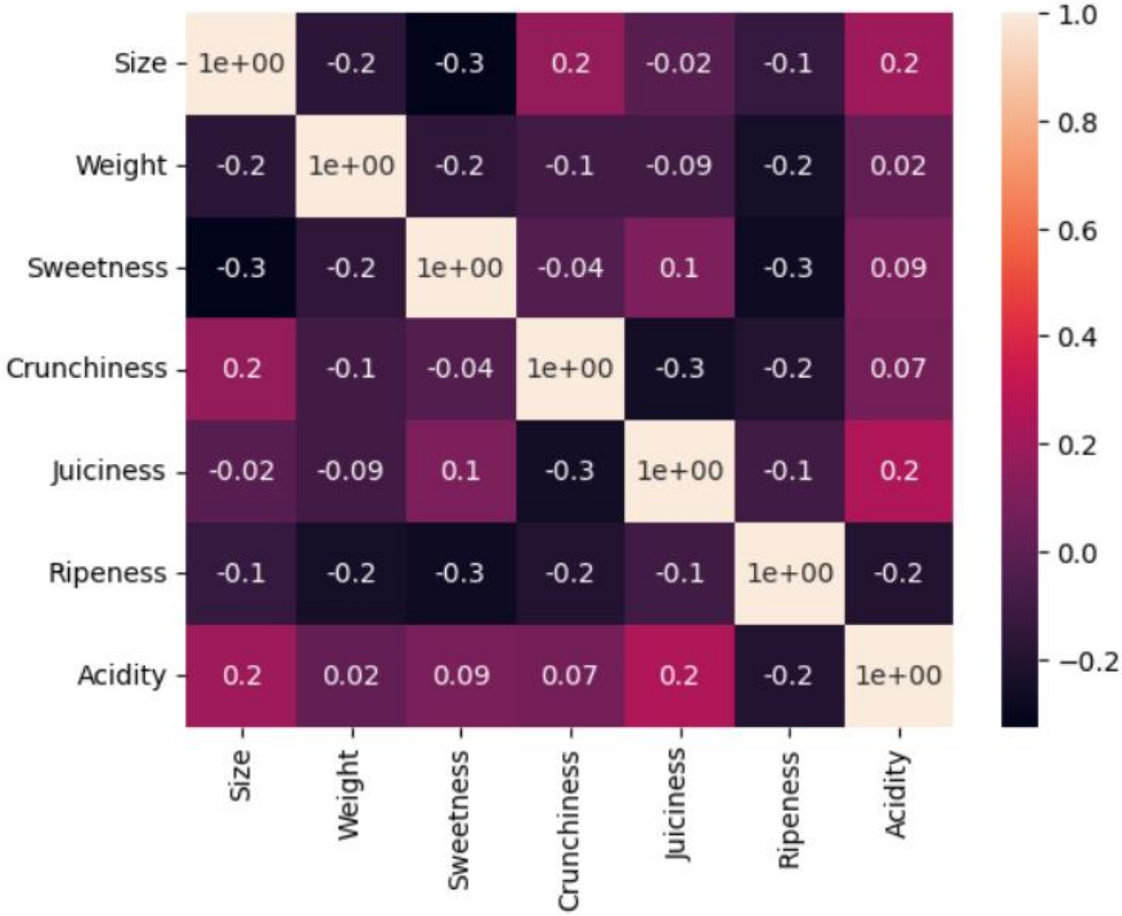
In [6]:
```
df.info()
```

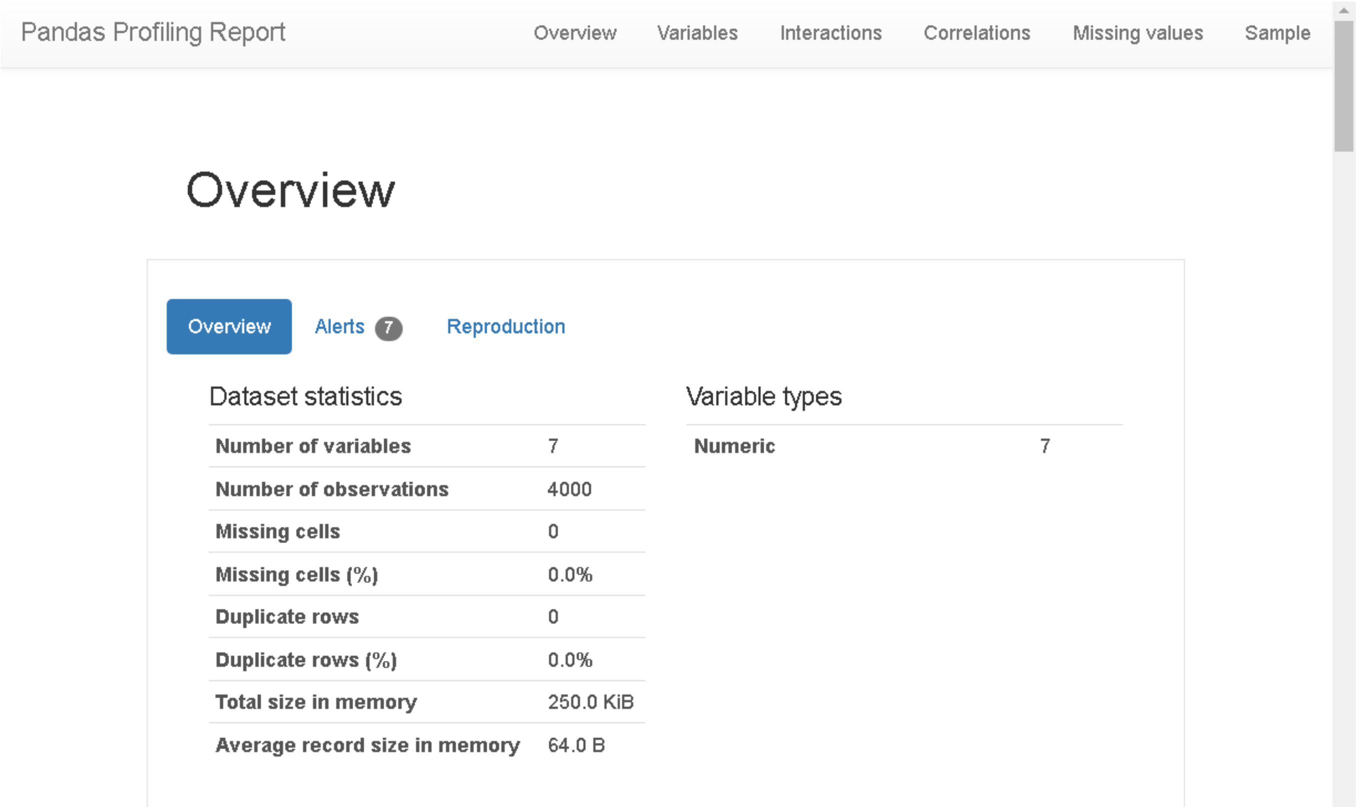
```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 4000 entries, 0 to 3999
Data columns (total 9 columns):
 #   Column       Non-Null Count  Dtype
---  ------       --------------  -----
 0   A_id         4000 non-null   float64
 1   Size         4000 non-null   float64
 2   Weight       4000 non-null   float64
 3   Sweetness    4000 non-null   float64
 4   Crunchiness  4000 non-null   float64
 5   Juiciness    4000 non-null   float64
 6   Ripeness     4000 non-null   float64
 7   Acidity      4000 non-null   float64
 8   Quality      4000 non-null   object
dtypes: float64(8), object(1)
memory usage: 312.5+ KB
```

In [7]:
```
df.describe().T
```
Out[7]:

	count	mean	std	min	25%	50%	75%	max
A_id	4000.0	1999.500000	1154.844867	0.000000	999.750000	1999.500000	2999.250000	3999.000000
Size	4000.0	-0.503015	1.928059	-7.151703	-1.816765	-0.513703	0.805526	6.406367
Weight	4000.0	-0.989547	1.602507	-7.149848	-2.011770	-0.984736	0.030976	5.790714
Sweetness	4000.0	-0.470479	1.943441	-6.894485	-1.738425	-0.504758	0.801922	6.374916
Crunchiness	4000.0	0.985478	1.402757	-6.055058	0.062764	0.998249	1.894234	7.619852
Juiciness	4000.0	0.512118	1.930286	-5.961897	-0.801286	0.534219	1.835976	7.364403
Ripeness	4000.0	0.498277	1.874427	-5.864599	-0.771677	0.503445	1.766212	7.237837
Acidity	4000.0	0.076877	2.110270	-7.010538	-1.377424	0.022609	1.510493	7.404736

In [17]:
```
seaborn.heatmap(corr_matrix, annot = True, fmt=".000")
```
Out[17]: <Axes: >





Pandas Profiling Report   Overview   Variables   Interactions   Correlations   Missing values   Sample

Overview

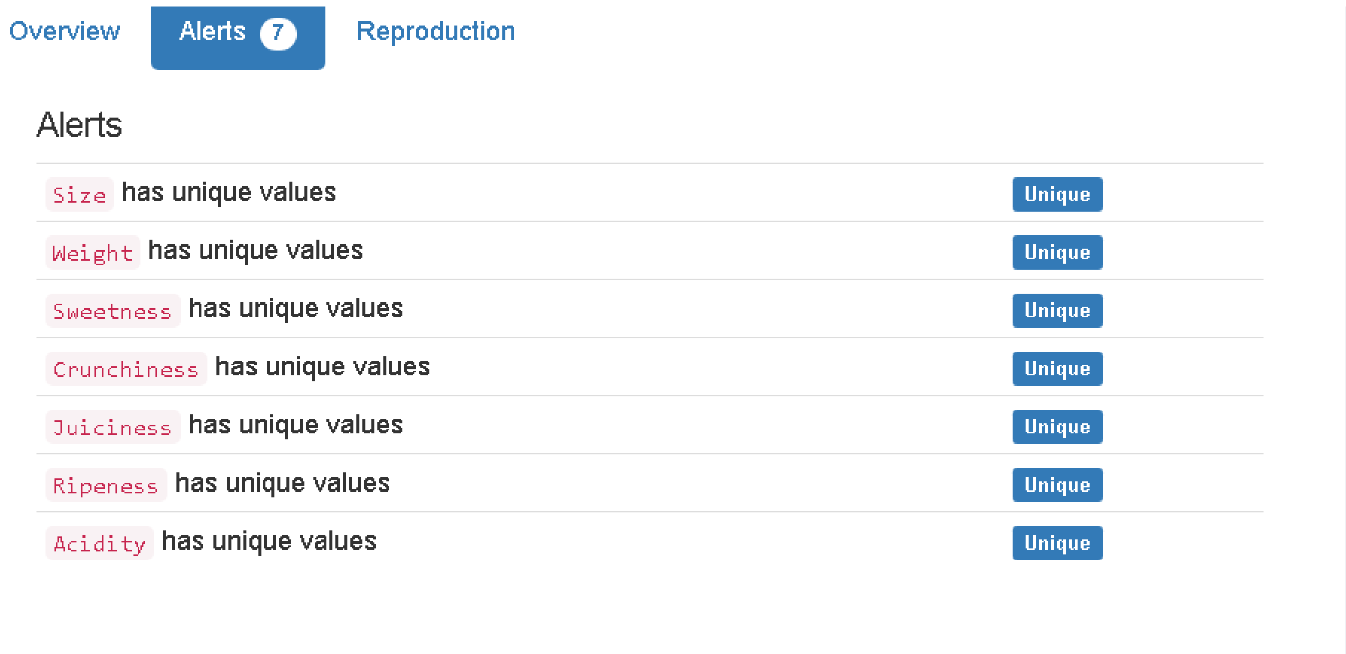
Overview   Alerts 7   Reproduction

Dataset statistics

Number of variables	7
Number of observations	4000
Missing cells	0
Missing cells (%)	0.0%
Duplicate rows	0
Duplicate rows (%)	0.0%
Total size in memory	250.0 KiB
Average record size in memory	64.0 B

Variable types

Numeric	7



Overview   Alerts 7   Reproduction

Alerts

`Size` has unique values	Unique
`Weight` has unique values	Unique
`Sweetness` has unique values	Unique
`Crunchiness` has unique values	Unique
`Juiciness` has unique values	Unique
`Ripeness` has unique values	Unique
`Acidity` has unique values	Unique

Overview Alerts 7 **Reproduction**

Reproduction

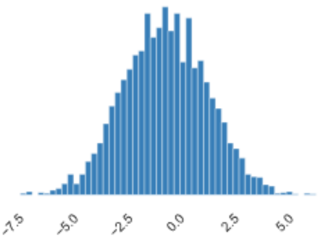
Analysis started	2024-04-04 10:04:35.921067
Analysis finished	2024-04-04 10:04:55.583809
Duration	19.66 seconds
Software version	ydata-profiling vv4.6.5
Download configuration	config.json

Variables

Select Columns

- Select Columns
- Size
- Weight
- Sweetness
- Crunchiness
- Juiciness
- Ripeness
- Acidity

Distinct	4000	**Minimum**	-7.1517031
Distinct (%)	100.0%	**Maximum**	6.4063669
Missing	0	**Zeros**	0
Missing (%)	0.0%	**Zeros (%)**	0.0%
Infinite	0	**Negative**	2401
Infinite (%)	0.0%	**Negative (%)**	60.0%
Mean	-0.50301463	**Memory size**	62.5 KiB



More details

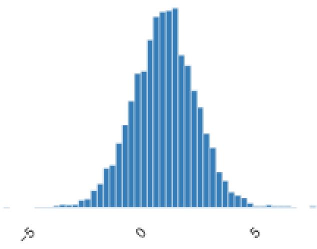
Crunchiness

Real number (ℝ)

UNIQUE

Distinct	4000
Distinct (%)	100.0%
Missing	0
Missing (%)	0.0%
Infinite	0
Infinite (%)	0.0%
Mean	0.9854779

Minimum	-6.0550578
Maximum	7.6198518
Zeros	0
Zeros (%)	0.0%
Negative	939
Negative (%)	23.5%
Memory size	62.5 KiB



More details

More details

Statistics | Histogram | Common values | Extreme values

Quantile statistics

Minimum	-6.0550578
5-th percentile	-1.3558499
Q1	0.062764395
median	0.99824944
Q3	1.8942342
95-th percentile	3.2044241
Maximum	7.6198518
Range	13.67491
Interquartile range (IQR)	1.8314698

Descriptive statistics

Standard deviation	1.4027572
Coefficient of variation (CV)	1.4234284
Kurtosis	0.72202048
Mean	0.9854779
Median Absolute Deviation (MAD)	0.91089751
Skewness	0.00023010591
Sum	3941.9116
Variance	1.9677278
Monotonicity	Not monotonic

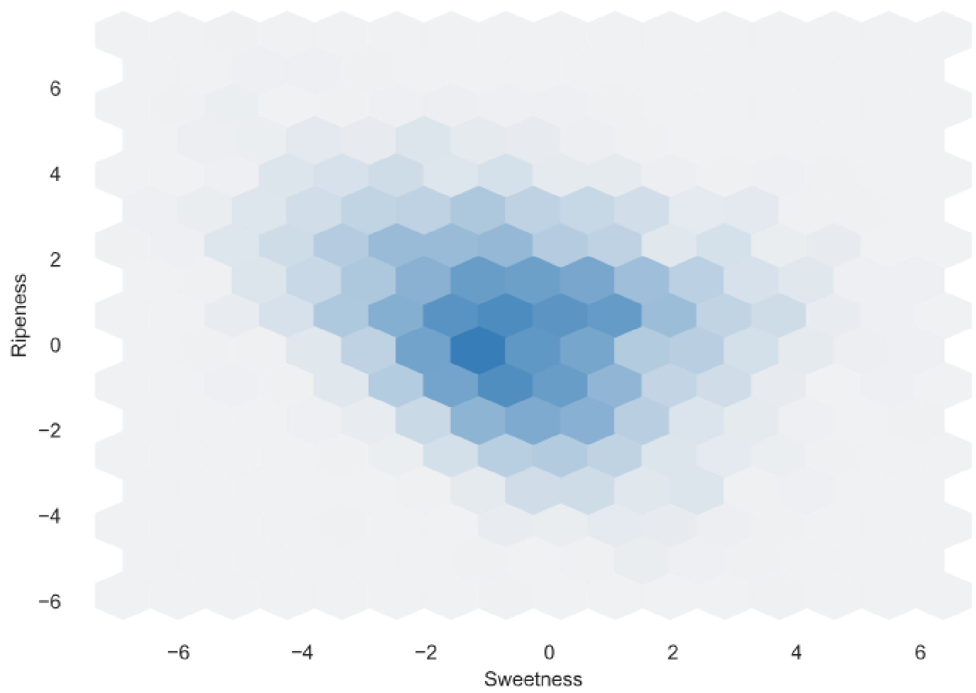
Interactions

Size Weight Sweetness Crunchiness Juiciness Ripeness Acidity

Acidity Size Weight Sweetness Crunchiness Juiciness Ripeness

Size Weight Sweetness Crunchiness Juiciness Ripeness Acidity

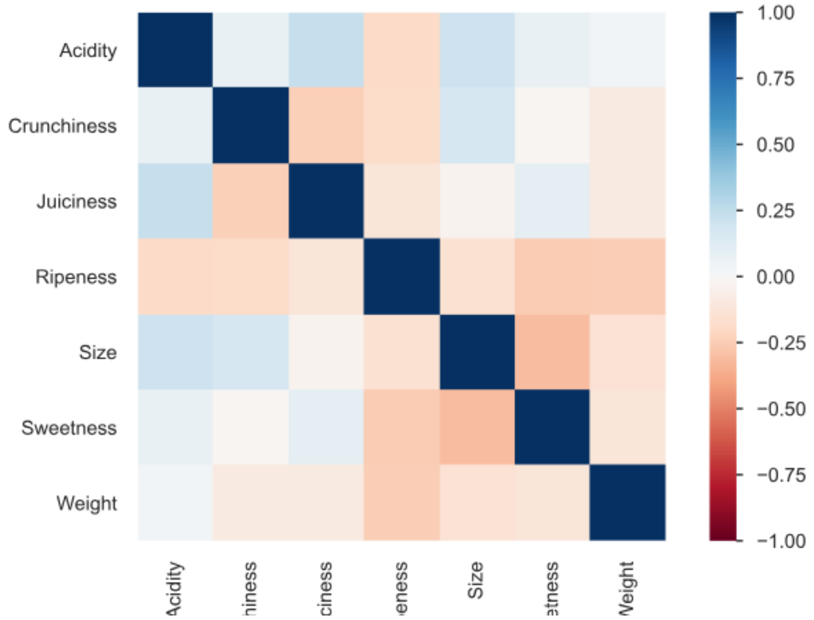
Acidity Size Weight Sweetness Crunchiness Juiciness Ripeness



Correlations

Auto

Heatmap Table



Correlations

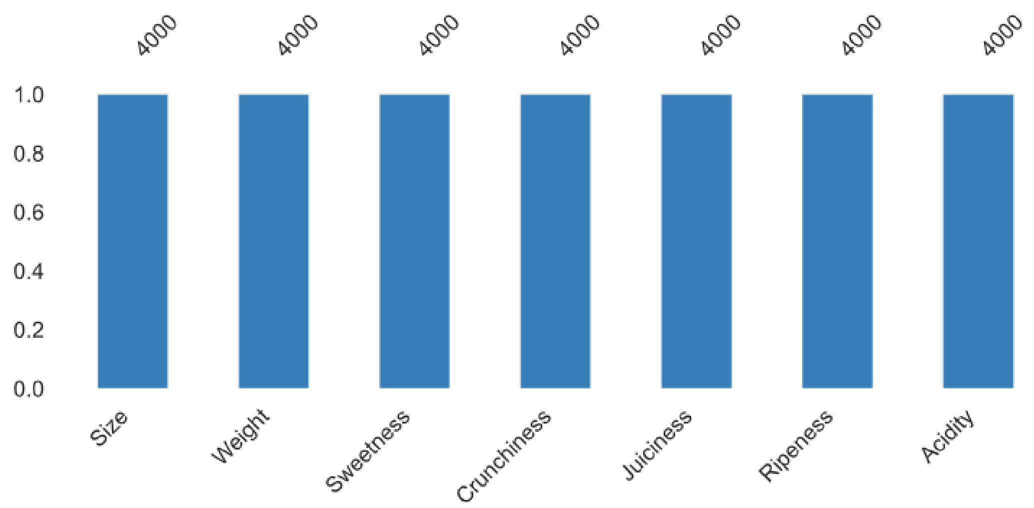
Auto

Heatmap Table

	Acidity	Crunchiness	Juiciness	Ripeness	Size	Sweetness	Weigl
Acidity	1.000	0.074	0.231	-0.195	0.210	0.072	0.030
Crunchiness	0.074	1.000	-0.237	-0.184	0.172	-0.017	-0.087
Juiciness	0.231	-0.237	1.000	-0.124	-0.032	0.098	-0.091
Ripeness	-0.195	-0.184	-0.124	1.000	-0.155	-0.255	-0.244
Size	0.210	0.172	-0.032	-0.155	1.000	-0.310	-0.144
Sweetness	0.072	-0.017	0.098	-0.255	-0.310	1.000	-0.120
Weight	0.030	-0.087	-0.091	-0.244	-0.144	-0.120	1.000

Missing values

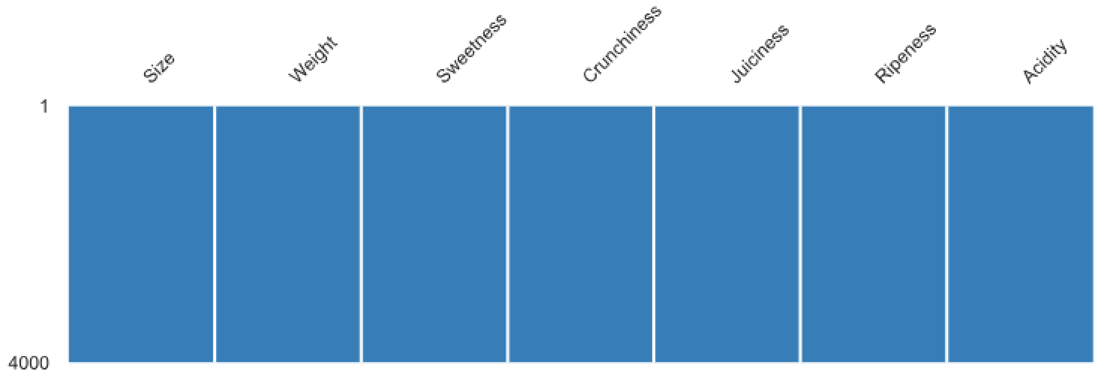
Count Matrix



A simple visualization of nullity by column.

Missing values

Count Matrix



Nullity matrix is a data-dense display which lets you quickly visually pick out patterns in data completion.

In [16]:
```
# Evaluation
print(f"Model - LR, Accuracy - {accuracy_score(ltest, lrpred)}")
print(f"Model - CT, Accuracy - {accuracy_score(ltest, ctpred)}")
```

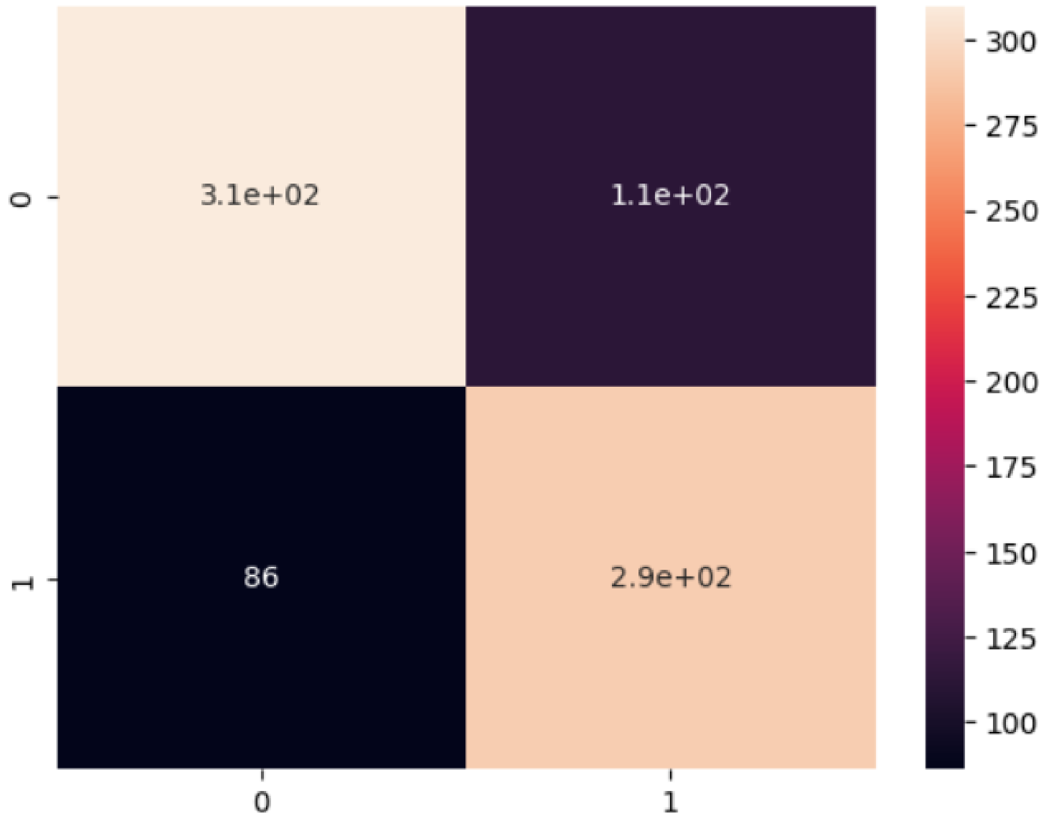
```
Model - LR, Accuracy - 0.7525
Model - CT, Accuracy - 0.76875
```

In [17]:
```
# Classification Matrix
print(f"LR\n{'=' * 65}")
print(f"{classification_report(ltest, lrpred)}")
print(f"CT\n{'=' * 65}")
print(f"{classification_report(ltest, ctpred)}")
```

```
LR
=================================================================
              precision    recall  f1-score   support

         bad       0.78      0.73      0.76       422
        good       0.72      0.77      0.75       378

    accuracy                           0.75       800
   macro avg       0.75      0.75      0.75       800
weighted avg       0.75      0.75      0.75       800

CT
=================================================================
              precision    recall  f1-score   support

         bad       0.78      0.78      0.78       422
        good       0.76      0.75      0.75       378

    accuracy                           0.77       800
   macro avg       0.77      0.77      0.77       800
weighted avg       0.77      0.77      0.77       800
```

```
In [18]: # Confusion Matrix
seaborn.heatmap(confusion_matrix(ltest, lrpred), annot=True)
```

```
Out[18]: <Axes: >
```



Discussions:-

In looking at the presentation and attributes of calculated relapse and choice tree classifier models for apple quality arrangement, it's clear that every strategy offers particular benefits and contemplations. Strategic relapse, being a direct model, gives a basic yet interpretable way to deal with characterization. Its coefficients offer direct experiences into the impact of each component on the arrangement choice. Notwithstanding, calculated relapse might battle with catching mind boggling, nonlinear connections among highlights. Then again, choice tree classifiers succeed in catching nonlinear examples and cooperations inside the information. Their progressive construction takes into account natural navigation and simple translation. In any case, choice trees can be inclined to overfitting, especially with profound trees, which might restrict their speculation execution on concealed information. Hence, the decision between strategic relapse and choice tree classifiers relies upon the compromises between interpretability, model intricacy, and execution prerequisites in the particular setting of apple quality arrangement. Further investigation could include troupe techniques or half breed ways to deal with influence the qualities of the two models for further developed grouping exactness and interpretability.

Chapter 8
Implementation

```
# Imports
import numpy as np
import pandas as pd
import seaborn
import matplotlib.pyplot as plt
from ydata_profiling import ProfileReport
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LogisticRegression
from sklearn.tree import DecisionTreeClassifier
from sklearn.metrics import accuracy_score, confusion_matrix, classification_report

%matplotlib inline

path = r"C:\Users\BHAVESH\OneDrive\Desktop\fruit\apple_quality.csv"
df = pd.read_csv(path)
df.head()
df.tail()

df = df.dropna()
df["Acidity"] = df["Acidity"].astype("float64")

df.info()
df.describe().T
df.hist(figsize=(12,8))

seaborn.countplot(data=df, x="Quality",)

shadow = df.copy()
shadow = shadow.drop(['A_id'], axis=1)

seaborn.boxplot(data=shadow)

df.isnull().sum()
df.isna().sum()

shadow = shadow.drop(['Quality'], axis = 1)
corr_matrix = shadow.corr()
```

```
# Heatmap
seaborn.heatmap(corr_matrix, annot = True, fmt=".0%")
seaborn.heatmap(corr_matrix, annot = True, fmt=".000")

ProfileReport(shadow)

features = df.drop(['A_id','Quality'], axis=1)
label = df['Quality']
ftrain, ftest, ltrain, ltest = train_test_split(features, label, test_size=0.2, random_state=0)

# Models
LR = LogisticRegression()
CT = DecisionTreeClassifier()

# Training
LR.fit(ftrain, ltrain)
CT.fit(ftrain, ltrain)



# Predictions
lrpred = LR.predict(ftest)
ctpred = CT.predict(ftest)

# Evaluation
print(f"Model - LR, Accuracy - {accuracy_score(ltest, lrpred)}")
print(f"Model - CT, Accuracy - {accuracy_score(ltest, ctpred)}")

# Classification Matrix
print(f"LR\n{'=' * 65}")
print(f"{classification_report(ltest, lrpred)}")
print(f"CT\n{'=' * 65}")
print(f"{classification_report(ltest, ctpred)}")

# Confusion Matrix
seaborn.heatmap(confusion_matrix(ltest, lrpred), annot=True)

seaborn.heatmap(confusion_matrix(ltest, ctpred), annot=True)
```

Chapter 9
Challenges and Future Direction

Exploring the eventual fate of apple quality arrangement with calculated relapse and choice tree classifiers involves tending to a few critical difficulties while chasing after promising roads for progression:

Data Quality Assurance: Defeating the test of guaranteeing top notch marked information includes executing productive information assortment methodologies and utilizing computerized or semi-robotized explanation strategies to upgrade precision and versatility.

Feature Engineering Innovation: Future bearings ought to investigate imaginative component designing ways to deal with recognize and consolidate nuanced qualities of apple quality, possibly utilizing progressed procedures, for example, profound learning-based include extraction to catch inconspicuous traits.

Model Generalization Enhancement: To accomplish strong model speculation, resolving issues like overfitting, especially pervasive in choice tree classifiers, requires investigating regularization strategies and gathering learning procedures to further develop strength and execution on concealed information.

Adaptive Learning Strategies: Given the powerful idea of organic product quality appraisal, future examination ought to zero in on creating versatile learning approaches that empower models to persistently refresh and refine their expectations in view of developing information patterns and occasional varieties.

Real-World Integration and Stakeholder Collaboration: Overcoming any issues between AI research and down to earth application inside the apple business requires cooperative endeavors between analysts, cultivators, and partners. Guaranteeing the versatility, convenience, and true viability of created models requires interdisciplinary joint effort and partner commitment.

Future executions of strategic relapse and choice tree classifiers for apple quality characterization hold guarantee in a few regions:

Integration with IoT and Sensor Technologies: Consolidating information from Web of Things (IoT) gadgets and sensor innovations can give continuous data about ecological circumstances, organic product readiness, and quality ascribes. Coordinating this information with AI models can upgrade the precision and practicality of apple quality evaluation.

Advanced Feature Engineering Techniques: Future executions can investigate progressed include designing strategies, for example, profound learning-based include extraction or unaided element learning, to catch mind boggling examples and subtleties in apple quality ascribes.

Ensemble Learning Approaches: Utilizing outfit learning techniques, for example, irregular woodlands or inclination supporting, can join the qualities of different classifiers to further develop in general expectation precision and strength. Gathering procedures can moderate the impediments of individual models and upgrade execution in testing situations.

Dynamic Model Adaptation: Carrying out powerful model variation systems permits models to consistently learn and adjust to advancing information patterns and occasional varieties in organic product quality. Versatile learning calculations empower models to refresh their forecasts continuously, guaranteeing exact and exceptional appraisals.

Deployment in Smart Agriculture Systems: Future executions can convey AI models for apple quality grouping inside shrewd horticulture frameworks, empowering robotized dynamic cycles for cultivators. Incorporation with ranch the executives programming and automated collecting frameworks can upgrade natural product reaping and work on in general efficiency.

Cross-Domain Applications: Expanding the materialness of apple quality arrangement models to other natural product types and rural items presents an interesting an open door. Summing up the model design and preparing strategies can work with transformation to various yields, extending the span and effect of the innovation.

User-Friendly Interfaces and Decision Support Systems: Creating easy to use points of interaction and choice emotionally supportive networks that give noteworthy experiences and suggestions in view of model forecasts enables partners in the apple business to settle on informed conclusions about organic product quality administration and inventory network tasks.

Chapter 10
Conclusion

In conclusion, the implementation of logistic regression and decision tree classifiers for apple quality classification represents a significant step forward in leveraging machine learning techniques to address agricultural challenges. Through this endeavor, we have explored the potential of these models to accurately classify apples into different quality grades based on attributes such as size, color, sweetness, and texture.

Our findings demonstrate that both logistic regression and decision tree classifiers offer valuable insights into apple quality assessment, each with its own set of strengths and considerations. Logistic regression provides a straightforward and interpretable approach, making it suitable for scenarios where transparency in decision-making is paramount. On the other hand, decision tree classifiers excel in capturing complex patterns and interactions within the data, offering flexibility and scalability in classification tasks.

Despite their differences, both models have demonstrated promising performance in accurately predicting apple quality grades. However, challenges such as data quality assurance, feature engineering complexity, and model generalization remain pertinent considerations for future research and development.

Looking ahead, future implementations of these models hold promise in various domains, including integration with IoT technologies, advanced feature engineering techniques, and dynamic model adaptation strategies. By addressing these challenges and embracing emerging opportunities, logistic regression and decision tree classifiers can revolutionize apple quality assessment practices, driving efficiency, sustainability, and innovation in the agriculture sector.

Chapter 11
References and Citations

References:-

1. **Dataset (CSV file):-**

   https://www.kaggle.com/datasets/nelgiriyewithana/apple-quality

2. **Project Reference:-**

   https://www.kaggle.com/datasets/nelgiriyewithana/apple-quality/code

Citation:-

Agus Pratondo, Devira M. A. Harahap, "Classification of Apples using Machine Learning", *2022 IEEE 13th Control and System Graduate Research Colloquium (ICSGRC)*, pp.182-186, 2022.