ORIGINAL ARTICLE

A C G
publications

Rec. Agric. Food. Chem. 3:2 (2023) 45-54

records of agricultural and
food chemistry

An Application of Data Mining Problem by Python: A Case Study on Fruit Classification

Oğuz Akpolat [1*] and Gonca Ertürk [1]

[1] *Muğla Sıtkı Koçman University, Science Faculty, Department of Chemistry, Muğla, Turkiye*

(Received November 27, 2023; Revised December 21, 2023; Accepted December 24, 2023)

Abstract: Fruits contain vitamins and dietary fibers, a vital source of the human diet. According to fruit production statistics, millions of metric tons of fruits was produced worldwide, and the advanced agricultural fruit recognition system with a simple camera or sensor will play an excellent role for farmers and general people. For the classification of fruits and vegetables, the features that come to mind first are sizing, color, and smell. The other ones are physical, chemical, and biological properties, and their tastes are important for identification, too. Such features also provide a perfect environment for creating imbalanced and incomplete datasets under the open-set protocol. The traditional methods that include a physical inspection of each fruit are less efficient, resulting in the development of more efficient and effective classification algorithms. Data mining algorithms for classification is operated in some software without being directly coded, KNIME, or can be modeled in code-able software such as PYTHON. It is a widely used high-level, general-purpose, interpreted, dynamic programming language, and pandas, numpy, matplotlib, and scikit-learn packages is frequently used as the basis for programming with PYTHON. This work aims to distinguish between different types of fruits using a simple dataset for the task of training a classifier. In this study prepared using the data, 59 fruit samples with the species name, subspecies name, weight, width, height, and color properties were analyzed using Decision Tree (DTR), K-Nearest Neighbors (KNN), Linear Discriminant Analysis (LDA), Gaussian Navie Bayes (GNB) and Support Vector Machine (SVM) classifiers. Accuracy values for each classifier were defined and compared with each other, and then the classifier with the highest accuracy was examined in detail. In the decision tree of samples classified according to fruit names, it is understood that height is the most effective measure in classifying fruits, and they are in the Bin 1 class (15 pieces). The second effective one is in the Bin 1 class (15 units) with height. The third effective one is in the Bin 1 class with width (12 peaces).

Keywords: Fruit; Classification; Python; decision tree; Accuracy.

1. Introduction

Consumption of fruits and vegetables is important for human health because these foods are primary sources of some essential nutrients and contain phytochemicals that may lower the risk of chronic disease. Because the many fruits and vegetables available to the population vary in composition for nutrients and phytochemicals, classifying fruits and vegetables is important to researchers who attempt to assess relationships among diet, health, and disease. The proposed classifications for fruits and vegetables is offered to nutrition professionals as a means to group fruits and vegetables more accurately based on food components of public health significance. For the classification of fruits and vegetables, the features that come to mind first are sizing, colour, and smell. The other ones are physical, chemical, and biological properties, and their tastes are important for identification, too [1].

[*] Corresponding author: e-Mail: oakpolat@gmail.com

The article was published by ACG Publications
https://www.acgpubs.org/journal/records-of-agricultural-and-food-chemistry July-December 2023 EISSN:2792-0763
DOI: http://doi.org/10.25135/rfac.18.2311.2980
Available online: December 28, 2023

An application of data mining problem

Classification of fruits and vegetables is a complex task based on a set of highly variable attributes, i.e., texture, colour, shape, and size. These attributes were used as features, and their variability poses significant challenges for a classification task. Due to this variance, it is very impractical to obtain an exhaustive dataset to train a classifier for fruit and vegetable classification. Such features also provide a perfect environment for creating imbalanced and incomplete datasets under the open-set protocol. Training on an imbalanced dataset can constrain the performance of a classifier to readily available samples. For example, a fruit and vegetable dataset will usually have more images of a fruit with normal colour, texture, shape, and size than a fruit with irregular colour, shape, or size. In the past several years, Deep Learning (DL) approaches have achieved state-of-the-art classification performance but are prone to errors when trained using imbalanced and complex datasets. Deep Convolutional Neural Networks (DCNNs) have achieved a performance boost for different applications, e.g., object recognition, face recognition, and verification [2].

Fruits contain vitamins and dietary fibers, a vital source of the human diet. Different types of more than 2,000 fruits are found worldwide, but most people are familiar with only 10%. According to fruit production statistics, millions of metric tons of fruits were produced worldwide in 2021, with the largest producing countries being China, India, and Brazil. The advanced agricultural fruit recognition system with a simple camera or sensor will play an excellent role for farmers and general people. In this modern era of technological advancements, fruit classification, and recognition systems can be used for kids' educational purposes, which interests them greatly. The latest advanced computer vision technology utilizing deep neural networks can be used for object discovery and semantic picture division. Thanks to computer vision and advanced image processing techniques, it has been successfully employed in various works for automatic fruit recognition and classification. The traditional methods that include a physical inspection of each fruit are less efficient, resulting in the development of more efficient and effective classification algorithms. Each article is analysed for design principles, mathematical model, Algorithms for Classification, features considered and extracted for Classification, and algorithm efficiency compared to others. The current state-of-the-art techniques and algorithms used for fruit classification are analysed thoroughly. The analysis assists in deciding the algorithm to be used for Classification based on the type of fruit and fruit features taken into consideration [3, 4].

Data analysis techniques, in which potentially useful and understandable information held among colossal data volumes, the meaning of which has not been discovered before, is extracted, and which includes database management systems, statistics, artificial intelligence, machine learning, parallel and distributed processes in the background, are called data mining. Many different techniques are used in this process, such as classification, clustering, data summarization, learning classification rules, finding dependency networks, variability analysis, and abnormal detection. Classification is one of the main data mining methods and is based on a learning algorithm. It is applied to discover a hidden pattern in large-scale data. Within the scope of data mining, the pattern is recorded digitally for an entity; It is expressed as observable, measurable, and repeatable information. Classification algorithms applied to obtain the desired information enable the data set to be divided into certain classes according to the common characteristics of the data it contains. Following this process, a classification model is obtained. The resulting classification model is applied to a new data set, and the presence of similar classes in the data set determined by the model is investigated. The process in question is also called "pattern recognition." Nowadays, a lot of data is produced in analysis devices in the field of chemistry and can be stored digitally. It is possible to evaluate these data, reveal the relationships between them, and make predictions about the future with the help of new data measured based on these relationships, thanks to data mining algorithms [5].

Data mining is defined as obtaining previously unknown, valid, and applicable information from data stacks through a dynamic process. While classification and curve fitting are defined as data mining prediction methods, such as clustering and association analysis are described as descriptive. The main classification methods are decision trees, Bayesian classification, artificial neural networks, and decision support machines. In short, classification examines the attributes of a new object and assigns this object to a predefined class. The important thing here is that the characteristics of each class are determined in advance. Clustering is the grouping of data according to their closeness or distance to each other. There are no predetermined group boundaries, but it can be optimized by giving the number of groups. Data mining software is divided into two groups: commercial and open source. Data mining algorithms are operated in some software without being directly coded or can be modelled in code-able software such as PYTHON. PYTHON is a widely used high-level, general-purpose, interpreted, dynamic programming language. The design philosophy emphasizes the readability of code, and the syntax allows programmers to express concepts in fewer lines of code than is possible in languages such as C++ or Java. The language provides constructs intended to enable clear programs at both small and large scales. Pandas, numpy, matplotlib, and scikit-learn packages are frequently used as the basis for programming with PYTHON [5, 6, 7, 8].

In a study, Li (2017) implemented several machine-learning algorithms in Python using Scikit-learn, the most popular machine-learning tool for Python. A simple dataset was used for the task of training a classifier to distinguish between different types of fruits. This post aims to identify the machine learning algorithm best suited for the problem at hand and compare various algorithms, selecting the best-performing one. The fruits dataset was created by Dr. Iain Murray from the University of Edinburgh in this work. He bought a few dozen oranges, lemons, and apples of different varieties and recorded their measurements in a table. Then, the University of Michigan professors formatted the fruit data slightly, and it can be downloaded from there. In this study prepared using these data, 59 fruit samples with the species name, subspecies name, weight, width, height, and color properties were analyzed using Decision Tree (DTR), K-Nearest Neighbours (KNN), Linear Discriminant Analysis (LDA), Gaussian Navie Bayes (GNB) and Support Vector Machine (SVM) classifiers. Accuracy values for each classifier were defined and compared with each other, and then the classifier with the highest accuracy was examined in detail. Python was used in the study [9, 10, 11, 12] for programming.

2. Materials and Methods

Data mining models can be grouped under four main headings: prediction, clustering, connection analysis, and difference deviations. Predicting and clustering investigate each record's relationship to others, while objective and temporal connections can be examined in connection analysis. The most well-known classification techniques used for prediction are decision trees, statistical-based algorithms such as Bayesian and Regression, distance-based algorithms, and artificial neural networks. Of these, the classification can be mathematically defined as:

$$D = \{t_1, t_2, \dots, t_n\} \tag{1}$$

Let's have a database and let each t_i show a record.

$$C = C_1, C_2, \dots, C_m \tag{2}$$

Let m denote the set of classes consisting of classes.

$$f\colon D \to C \tag{3}$$

and each t_i should belong to a class. Here C_j is a separate class, each containing its own records. So, it can be shown in the form:

$$C_i = \left\{t_i / f(t_i) = C_j, 1 \le i \le n,\ \ and\ \ t_i \in D\right\} \tag{4}$$

Classification can also carry a class (discrete) and continuous value of the dependent variable with the class we have or its statistical definition. In this respect, it approaches regression or multi-term regression. Classification can also be defined as a supervised learning approach revealing hidden patterns within a certain range. The most common of these algorithms are ID3 and C4.5. Normalization is one of these algorithms' most frequently used data transformation processes. With Min-Max normalization, the most used data normalization technique, the original data are converted to the new data in range by a linear transformation. This data range is usually 0-1.

$$Newdata = \{(Rawdata - minRawdata)/(maxRawdata - minRawdata)\} \tag{5}$$

The principles of the decision tree method and steps of the decision tree algorithm are given below.

2.1. Basics of Decision Tree Method

1. Identification of the problem.
2. Drawing/structuring the decision tree.
3. Assigning the probabilities of the occurrence of events.
4. Calculation of the expected return (or benefit) for the corresponding chance point-backward, the transaction.
5. Assignment of the highest expected return (benefit) to the relevant decision point-backward comparison.
6. The submission of the proposal is based on its principles.

2.2. The Steps of The Decision Tree Algorithm

1. The learning set T is created.
2. The attribute that best separates the samples in the set T is determined.
3. A node of the tree is created with the selected attribute, and child nodes or leaves of the tree are created from this node. Determine the instances of the subset of child nodes.
4. For each sub-dataset created in step three:
   - If the samples all belong to the same class
   - If there is no qualification to divide the samples

If there is no sample with the remaining attribute value, the process is terminated. In the other case, the process is continued from the second step to separate the subset of data. The decision tree can be easily encoded in any programming language using IF-ELSE expressions. Decision tree classification is a classification method that creates a model in the form of a tree structure consisting of decision nodes and leaf nodes according to the property and goal. The decision tree algorithm is improved by dividing the data set into small and even smaller pieces. A decision node may contain one or more branches. The first node is called the root node. A decision tree can consist of both categorical and numerical data. The randomness, uncertainty, and probability of an unexpected situation occurring in the formation of any situation are defined by entropy. If all the samples are regular/homogeneous, their entropy becomes zero. Here, entropy is defined as follows:

$$E(S) = \sum_{i=1}^{c} - p_i \log_2 p_i \tag{6}$$

Entropy is not calculated only on the target. In addition, entropy can also be calculated on properties. But when calculating entropy on properties, it is considered in the target. In this case, entropy is defined as follows:

$$E(T,X) = \sum_{c \in X} P(c)E(c) \tag{7}$$

Information gain (Gain) is based on subtracting all entropy after dividing a dataset on a feature as follows). If the entropy is small, the importance of the feature increases for the Decision Tree algorithm ID3. On the other hand, as it gets closer to 1, the importance of the feature decreases. However, in information gain, the situation is the opposite; in this respect, it can be thought of as the inverse of entropy. While constructing the Decision Tree, the feature with the highest information gain is selected.

$$Gain(T,X) = Entropy(T) - Entropy(T,X) \tag{8}$$

Overfitting is an important problem for decision tree models and many other prediction models. Overfitting occurs when the training set continues to reduce errors in a way that affects the learning algorithm. To avoid overfitting in a decision tree construction, two approaches are usually used:

· Pre-pruning: Stopping the growth of the tree before the containment process.

· Post-pruning: first creating the whole tree and then removing the unnecessary parts from the tree.

Due to the difficulty in determining when pruning will be done in practice, the first approach is hardly used. The second approach is much more successful. Attention should be paid to the following steps in this approach.

· A different dataset than the training data is used to decide on the pruning process. This data set is called the validation dataset. The validation dataset is used to decide on unnecessary nodes.

· After obtaining a decision tree, using statistical methods such as error estimation and significance testing (Chi-Square Testing), it is decided whether there will be pruning and expansion (expanding – adding new nodes to the tree) on the training data.

· The Minimum Distance Description Principle is a measure between the Decision tree and the training dataset. When the size (tree) + Size (non-classifiable tree) is minimized, the tree growth is stopped.

In this section, the data were tested in the decision tree modelling performed using the Python programming language and measured for features of Fruit Samples and presented in Table 1. Here, the features of the fruits were physically defined as label, name, sub-type, mass, width, height, and color [8].

Table 1. Data set for features of fruit samples

No	**fruit_label**	**fruit_name**	**fruit_subtype**	**mass**	**width**	**height**	**color_score**
1	1	apple	granny_smith	192	8.4	7.3	0.55
2	1	apple	granny_smith	180	8.0	6.8	0.59
3	1	apple	granny_smith	176	7.4	7.2	0.60
4	2	mandarin	mandarin	86	6.2	4.7	0.80
Data File in the dimension of 59*8 (**fruit_data_with_colors.txt**)							
56	4	lemon	unknown	116	6.3	7.7	0.72
57	4	lemon	unknown	116	5.9	8.1	0.73
58	4	lemon	unknown	152	6.5	8.5	0.72
59	4	lemon	unknown	118	6.1	8.1	0.70

When it comes to the evaluation of your classifier, there are several different ways you can measure its performance. Classification accuracy is the simplest out of all the methods of evaluating accuracy and is the most commonly used. Classification accuracy is simply the number of correct predictions divided by all predictions or a ratio of correct predictions to total predictions. While it can give you a quick idea of how your classifier performs, it is best used when the number of observations/examples in each class is roughly equivalent. Because this doesn't happen often, you're probably better off using another metric. A confusion matrix is a table or chart representing the accuracy of a model concerning two or more classes. The model's predictions will be on the X-axis, while the outcomes/accuracy are located on the y-axis. The cells are filled with the number of predictions the model makes. Correct predictions can be found on a diagonal line moving from the top left to the bottom right. You can read more about interpreting a confusion matrix in Table 2. Accuracy is the rate of correct predictions in total data and is defined as follows:

$$Accuracy = (Total\ Correct\ Prediction)/N = (True\ Positive + True\ Negative)/N \qquad (9)$$

Table 2. Confusion matrix obtained by a classification algorithm

		Actual Class	
		1	**0**
	1	True Positive	False Negative
Predicted Class	**0**	False Negative	True Negative

The Python codes required for the analysis of the algorithm written for decision trees in data evaluation (Fruit_Simple_Classification_00_02_Decission_Tree.Tot.py) and its numerical output were given in Appendix 1. The graphs of the program outputs were given in Table 3, Figure 1, Figure 2, Figure 3, Figure 4, and Figure 5 as “Comparison of accuracies of applied classifiers,”, “Distribution of samples according to Fruit names,” “Display of sample distributions in box graphics,” “Histograms of distributions related to fruits’ futures,” “Correlation of fruit futures to each other” and “Decision tree of samples classified according to Fruit names,” respectively.

Table 3. Comparison of accuracies of applied classifiers

No	**Classifier**		**Accuracy**	
			on test set	**on training set**
1	Decision Tree	DT	1.00	0.67
2	K-Nearest Neighbors	K-NN	0.95	1.00
3	Linear Discrimant	LDA	0.86	0.67
4	Gaussian Navie Bayes	GNB	0.86	0.67
5	Support Vector Machine	SVM	0.91	0.80

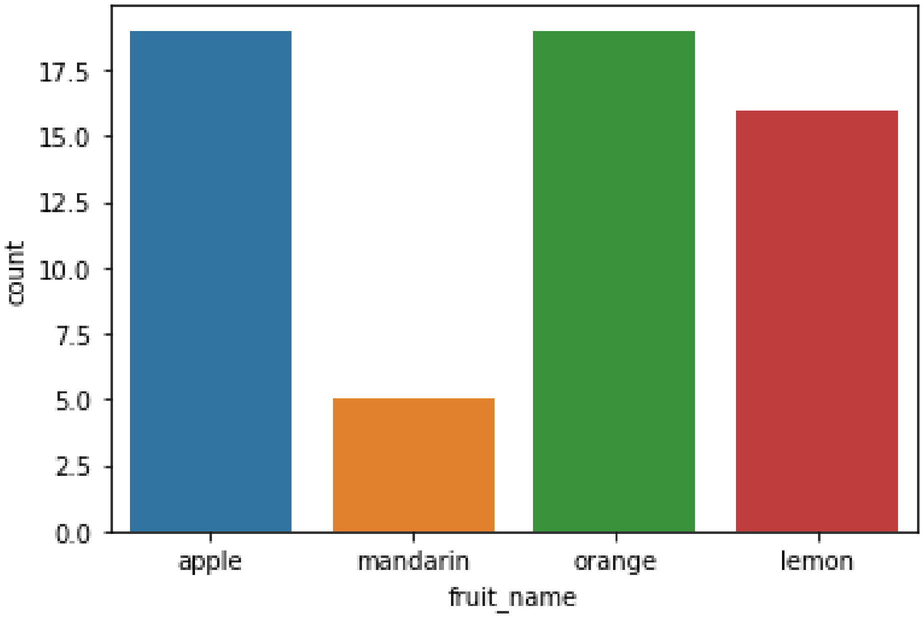

Figure 1. Distribution of samples according to fruit names

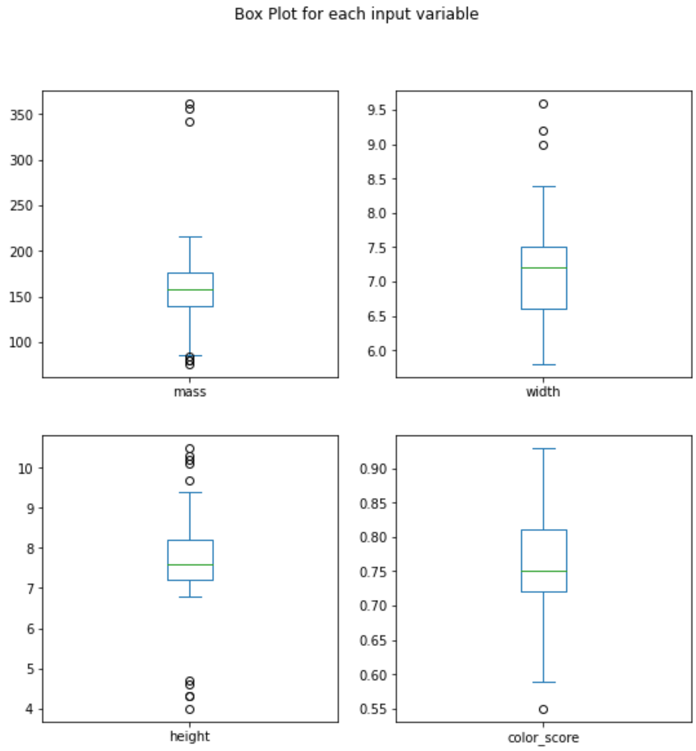

Figure 2. Showing the sample distributions in box graphics

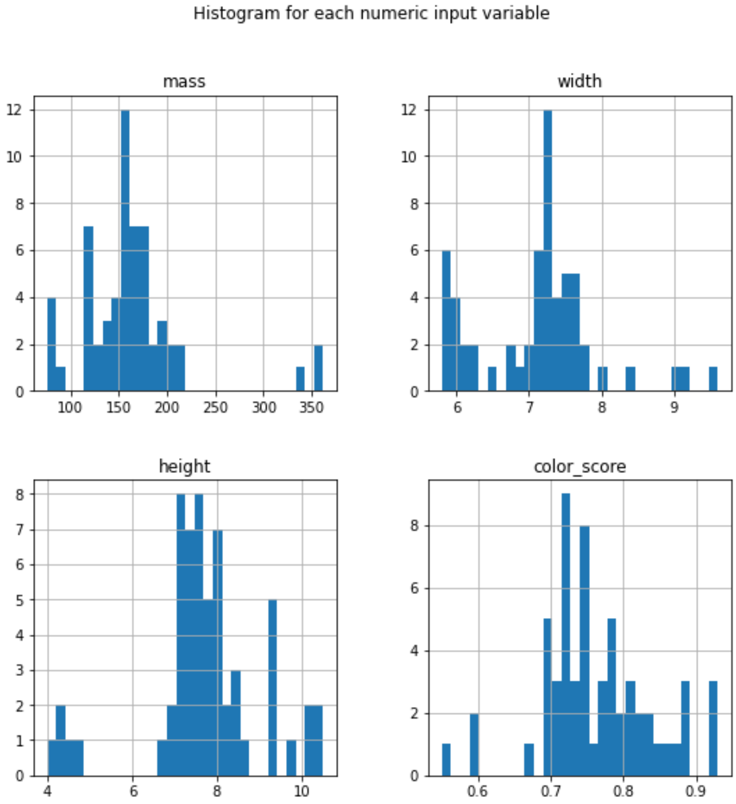

Figure 3. Histograms of distributions related to fruit futures

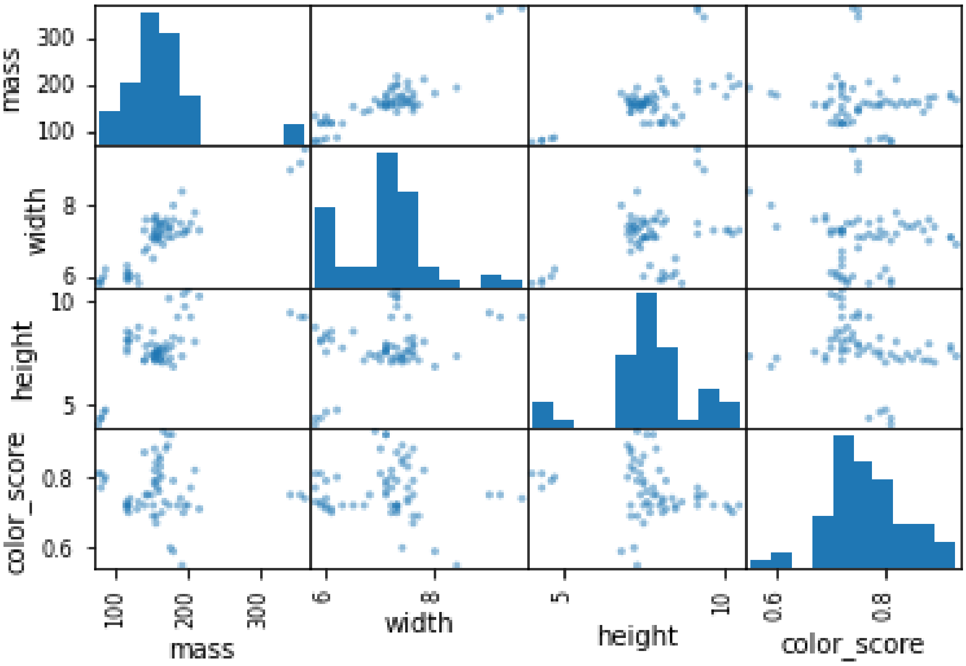

Figure 4. Correlation of fruit futures to each other

3. Results and Discussion

In this section, the outputs and results of calculated accuracies for some classifiers and the decision tree application selected for solving classification problems in detail have also been examined, and the numerical and % distribution of fruit samples are given in Table 4.

Table 4. Distribution of samples according to Fruit names

Fruit Name	Numeric value	Distribution (%)
Apple	19	32
Mandarin	5	9
Orange	19	32
Lemon	16	27

As seen in Table 3, firstly, accuracies of applied classifiers, Decision Tree, K-Nearest Neighbors, Linear Discriminant, Gaussian Navie Bayes, and Support Vector Machine were compared to each other for both test and training data sets. Then, the decision tree classifier with the highest accuracy value on the test set was chosen to examine in detail. But the other classifier also has higher accuracy values. The data tested in the decision tree modelling performed were measured for features of Fruit Samples and presented as given in Table 1. Here, the features of the fruits were physically defined as label, name, sub-type, mass, width, height, and colour. Figure 3 and Figure 4 show **h**istograms of distributions related to fruit futures and the correlation of fruit futures to each other.

As examined in Figure 5 and Table 5, the decision tree of samples classified according to fruit names, it is understood that the most effective measure in classifying fruits is height, and they are in the Bin 1 class (15 pieces). The second effective one is in the Bin 1 class (15 units) with height. The third effective one is in the Bin 1 class with width (12 units).

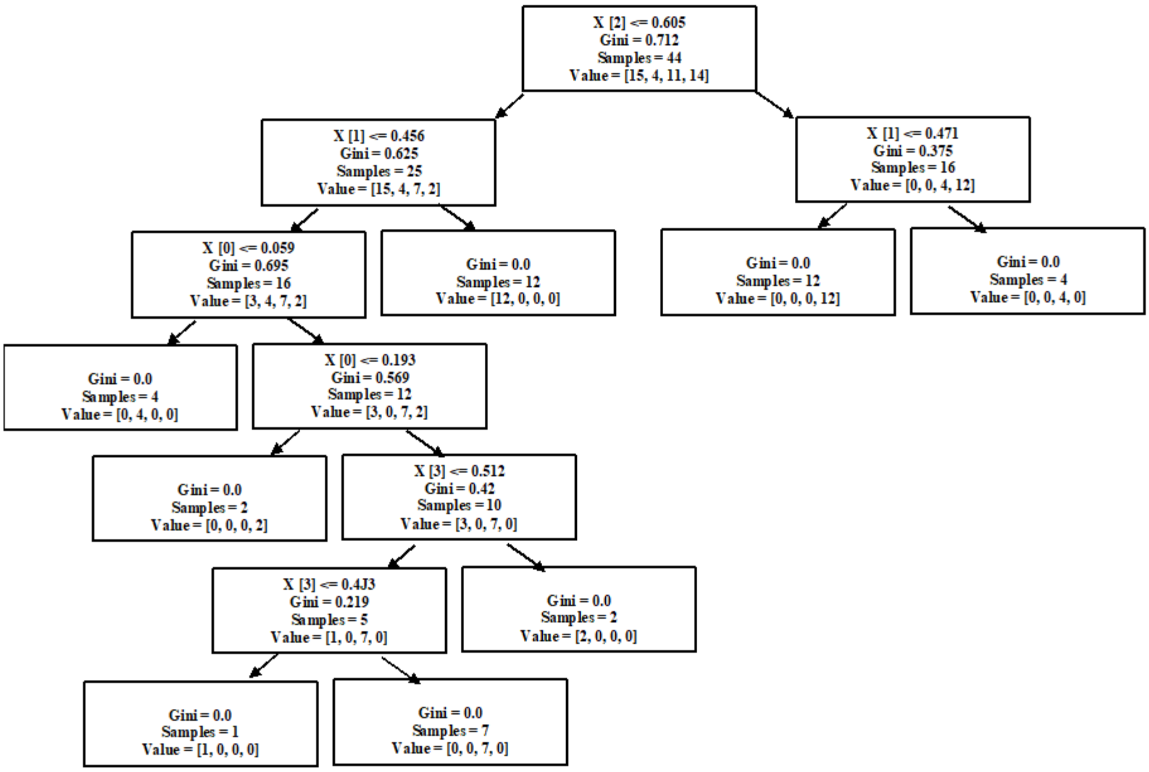

Figure 5. Decision tree of samples classified according to Ffruit names

Table 5. Detailed analysis values of decision tree

Root (1)	**Bin 1** (28/59) (X[2]<=0.605-**Height**, Gini Separation Index=0.712)						
	Mass (0)		**Width (1)**		**Height (2)**		**color_score (3)**
Bin 1					15	34	
Bin 2					4	9	
Bin 3					11	25	
Bin 4					14	32	
Total					**44**	**100**	
Root (1-1)	**Bin 1** (28/59) (X[1]<=0.456-**Height**, Gini Separation Index=0.625)						
Bin 1			15	0,54			
Bin 2			4	0,14			
Bin 3			7	0,25			
Bin 4			2	0,07			
Total			**28**	**100**			
Root (1-2)	**Bin 1** (12/59) (X[1]<=0.456-**Height**, Gini Separation Index=0.000)						
Bin 1			12	100			
Bin 2			0				
Bin 3			0				
Bin 4			0				
Total			**12**	**100**			

4. Conclusions

Classification of fruits and vegetables is a complex task based on a set of highly variable attributes, i.e., texture, colour, shape, and size. These attributes are used as features, and their variability poses significant challenges for a classification task. Due to this variance, it is very impractical to obtain an exhaustive dataset to train a classifier for fruit and vegetable classification. In this study prepared using these data, created by Dr. Iain Murray from the University of Edinburgh, 59 fruit samples with the species name, subspecies name, weight, width, height, and colour properties were analysed using Decision Tree (DTR), K-Nearest Neighbours (KNN), Linear Discriminant Analysis (LDA), Gaussian Navie Bayes (GNB) and Support Vector Machine (SVM) classifiers. Accuracy values for each classifier were defined and compared with each other, and then the classifier with the highest accuracy was examined in detail. For programming, Python was used in the study. Then, the decision tree classifier with the highest accuracy value on the test set was chosen to examine in detail but the other classifier also has higher accuracy values. This work can be used for a large-scale grading of fruits and vegetables, and it should not be forgotten much more samples and much more error in data mining works.

ORCID

Oğuz Akpolat: 0000-0002-6623-4323
Gonca Ertürk: 0000-0002-8821-0330

References

[1] J.A.T. Pennington and R.A. Fisher (2009). Classification of fruits and vegetables, *J. Food Comp. Anal.* 22S, 23–31.

[2] K. Hammed, D. Chai, and A. Rassau (2021). Class distribution-aware adaptive margins and cluster embedding for classification of fruit and vegetables at supermarket self-checkouts, *Neurocomputing* 461, 292-309.

[3] R.G. Brereton (2003). Chemometrics: Data analysis for the laboratory and chemical plant, John Wiley & Sons Ltd., Provide the city and Country, Chichester West Sussex, United Kingdom.

[4] N.E.A. Mimma, S. Ahmed, T. Rahman, and R. Khan (2022). Fruits Classification and Detection Application Using Deep Learning, *Hindawi Sci. Programm.* **16**, Article ID 4194874.

[5] P. Jaya Vineela. M. Sai Manvitha, P. Rishitha and U. Prabu (2022). a comprehensive study on fruit classification and grading techniques, *Proceedings of the Third International Conference on Electronics and Sustainable Communication Systems (ICESC 2022)* IEEE Xplore Part Number: CFP22V66-ART; ISBN: 978-1-6654-7971-4.

[6] G. Silahtaroğlu (2016), Veri madenciliği kavram ve algoritmaları, II. Baskı, Papatya Yayıncılık

[7] A. Çınar (2019). Veri Madenciliğinde sınıflandırma algoritmalarının performans değerlendirmesi ve r dili ile bir uygulama, *Marmara Üniv. Öneri Derg.* **14(51)**, 90-111.

[8] https://anaconda.org/anaconda/python, (2022). anaconda/packages/python3.10.6-540.

[9] S. Robinson (2022). Decision Trees in PYTHON with Scikit-Learn, https://stackabuse.com/decision-trees-in-python-with-scikit-learn/, StackΛBuse.

[10] D. Nelson (2022). Overview of Classification Methods in PYTHON with Scikit-Learn, https://stackabuse.com/overview-of-classification-methods-in-python-with-scikit-learn/, StackΛBuse.

[11] S. Li (2017). Solving A Simple Classification Problem with PYTHON — Fruits Lovers' Edition, https://towardsdatascience.com/solving-a-simple-classification-problem-with-python-fruits-lovers-edition-d20ab6b071d2.

[12] I. Lavrov, J. Domashova (2020). Constructor of compositions of machine learning models for solving classification problems, *Procedia Comput. Sci.* **169**,780-786.