İSTANBUL TİCARET ÜNİVERSİTESİ
FEN BİLİMLERİ DERGİSİ

İstanbul Commerce University Journal of Science

http://dergipark.org.tr/ticaretfbd

Research Article / Araştırma Makalesi

PERFORMANCE COMPARISON OF SMOTE-BASED MACHINE LEARNING MODELS ON UNBALANCED DATASETS: A STUDY ON DATE AND PISTACHIO FRUITS

SMOTE TABANLI MAKİNE ÖĞRENMESİ MODELLERİNİN DENGESİZ VERİ KÜMELERİ ÜZERİNDE PERFORMANS KARŞILAŞTIRMASI: HURMA VE ANTEP FISTIĞI MEYVELERİ ÜZERİNE BİR ÇALIŞMA

Fatih BAL[1] **Fatih KAYAALP**[2]

https://doi.org/10.55071/ticaretfbd.1597150

Corresponding Author (Sorumlu Yazar)	*Received (Geliş Tarihi)*	*Revised (Revizyon Tarihi)*	*Accepted (Kabul Tarihi)*
fatihbal@klu.edu.tr	*06.12.2024*	*20.03.2025*	*25.03.2025*

Abstract

Creating balanced datasets is a significant challenge that substantially affects the performance of machine learning models in the classification of agricultural products. In this research, we tried to overcome this challenge by using an unbalanced dataset containing information on 7 date palm (Phoenix dactylifera L.) and 2 pistachio (Pistacia vera L.) cultivars. The aim of the study is to compare the classification performance of machine learning models on an unbalanced dataset and a balanced dataset using the SMOTE technique. Initially, classification was performed on the unbalanced dataset using machine learning approaches. Among the machine learning models applied on the unbalanced dataset, the Linear-SVM model showed the highest accuracy rate with an accuracy rate of 92,62%. In the data set extended by applying the SMOTE technique, the RBF-SVM model again showed the highest accuracy rate with 95,55% accuracy rate. In summary, our study highlights the difficulties in machine learning-based agricultural crop classification due to data unbalances. Utilizing the SMOTE technique for oversampling was effective in overcoming this obstacle and improving classification accuracy.

Keywords: Machine learning, SMOTE, fruit classification, oversampling.

Öz

Dengeli veri kümeleri oluşturmak, tarımsal ürünlerin sınıflandırılmasında makine öğrenimi modellerinin performansını önemli ölçüde etkileyen önemli bir zorluktur. Yapılan bu araştırmada, 7 hurma (*Phoenix dactylifera L.)* ve 2 Antep fıstığı (*Pistacia vera L.)* çeşidine ait bilgileri içeren dengesiz bir veri kümesi kullanarak bu zorluğun üstesinden gelinmeye çalışılmıştır. Çalışmanın ana hedefi, makine öğrenmesi modellerinin dengesiz veri kümesi ve SMOTE tekniği ile dengelenmiş veri kümesi üzerindeki sınıflandırma başarısını karşılaştırmaktır. Başlangıç olarak, dengesiz veri kümesi üzerinde makine öğrenimi yaklaşımları kullanılarak sınıflandırma yapılmıştır. Dengesiz veri kümesinde uygulanan makine öğrenmesi modelleri içerisinde %92,62 doğruluk oranı ile en yüksek doğruluk oranını Linear-SVM modeli göstermiştir. SMOTE tekniği uygulanarak genişletilen veri kümesinde ise %95,55 doğruluk oranı ile en yüksek doğruluk oranını RBF-SVM modeli göstermiştir. Özetle, çalışmamız makine öğrenimi tabanlı tarımsal ürün sınıflandırmasında veri dengesizliklerinden kaynaklanan zorlukların altını çizmektedir. Aşırı örnekleme için SMOTE tekniğinden yararlanmak, bu engelin üstesinden gelmede ve sınıflandırma doğruluğunu artırmada etkili olmuştur.

Anahtar Kelimeler: Makine öğrenmesi, SMOTE, meyve sınıflandırma, aşırı örnekleme.

[1] Kırklareli University, Faculty of Engineering, Department of Software Engineering, Kırklareli, Türkiye.
fatihbal@klu.edu.tr

[2] Düzce University, Faculty of Engineering, Department of Computer Engineering, Düzce, Türkiye.
fatihkayaalp@duzce.edu.tr

1. INTRODUCTION

Dates (*Phoenix dactylifera L.*) are the fruit of the date palm tree. Being the dominant species of the Phoenix genus, these plants belong to the Aceraceae family, commonly known as the palm family, holding significant importance (Echegaray et al., 2023). The encompass around 200 types and over 2500 species worldwide (Koklu et al., 2021). Dates hold great agricultural significance in numerous parts of the world, particularly in the Middle East, North Africa, and Pakistan (Yıldız, 2019). Kirmizi Pistachio (*Pistacia vera L.*), also known as Antep Fistigi, is a variety of nut commonly grown in Mediterranean and Middle Eastern countries (Çağlar et al., 2017). It is a small tree or shrub that produces a greenish-yellow fruit, which is the actual nut. Enclosed within a hard, woody shell, the fruit splits open upon ripening to reveal the nut inside. The nut is oval or oblong-shaped, displaying a greenish-brown color with a wrinkled surface. Siirt Pistachio (*Pistacia vera L.*) native to the Siirt and Sanliurfa provinces in Turkey (Aydın, 2018), is another type of nut. It is also referred to as Siirt Pistachio or simply Siirt Pist. Although it bears similarities to Antep Fistigi, it is considered a distinct cultivar with a unique flavor profile. Like many other fruits, *date* and *pistachio* can be classified using machine learning (ML) techniques based on their genetic varieties. For instance, genetic variety features such as fruit eccentricity, roundness, compactness, solidity etc., can be utilized for fruit classification. The classification process primarily relies on using images of the fruits as the training dataset. Features are extracted from these images, and a classifier is trained using the features to classify the fruits. Machine learning (ML) techniques can be utilized to perform this classification task. To select the most suitable method, it is crucial to test the dataset. Additionally, factors such as dataset size, number of features, and data characteristics also influence the method selection.

This study aims to reduce the negative effects of classifying unbalanced datasets. The unbalanced dataset used in this study exhibits an unbalanced distribution of images of different fruit types. In machine learning, unbalanced datasets lead to problems such as poor generalization, reduced efficiency and inaccuracy. To remove the unbalance and achieve a balanced data distribution, the SMOTE oversampling technique was applied. The use of the SMOTE oversampling technique aims to balance the dataset by equalizing the number of samples from minority classes with the number of samples from the class with the highest number. The main objective of this research is to improve the classification and subclassification accuracy of ML models when working with unbalanced data sets. To study and improve the classification performance of machine learning models on both unbalanced and balanced datasets, the study consists of two phases. In the first phase, we investigate the performance of machine learning methods on unbalanced datasets. In the second stage, the performance of machine learning models on the balanced dataset after applying SMOTE is analyzed.

For the purpose of this study, data on the genetic characteristics of dates and pistachios were collected from various sources. In order to classify and compare these data, the performances of seven different machine learning models such as Decision Tree, Random Forest, Logistic Regression, Support Vector Machine, Multi-Layer Perceptron, K-Nearest Neighbor and Naive Bayes were examined. Performance measures such as accuracy, balanced accuracy, precision, sensitivity, specificity, F1-

score and ROC analysis of all models were calculated and compared to evaluate the effectiveness of ML models.

In the context of agricultural product classification, this study is one of the studies that analyze in detail the effect of imbalanced datasets on classification performance and how SMOTE balances this effect. Examining the effect of SMOTE on subclasses in a multi-class problem and verifying data integrity with JSD differentiates this study from the existing literature. While it is common in literature to use SMOTE for one-class or two-class datasets, this study performs a detailed analysis on the classification of 9 different fruit varieties.

Within the scope of the study, the research on similar studies is described in the second section. In the third section, the dataset used and the numerical and percentage distribution of the data according to the classes are given. The working principle of the SMOTE technique proposed in the study and the classification performances of the ML methods examined are also explained under this heading. Hyperparameter optimization is also presented in this section. In the fourth section, the criteria used to measure the classification performance of ML methods are explained. The classification performance results of ML methods are presented in detail in this section. In the fifth section, the results are evaluated. The architecture of the proposed model is summarized in Figure 1.

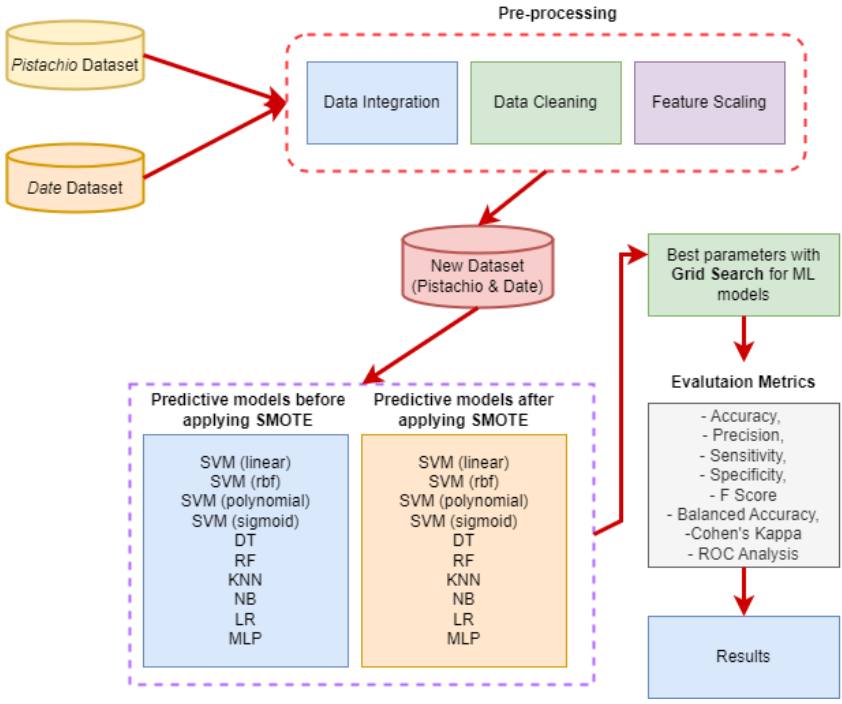

Figure 1. The Architecture of the Proposed Model for This Study.

2. LITERATURE REVIEW

Similar studies and applications are analyzed in this section of the study. The focus was on the performance of ML methods based on SMOTE. In a study conducted by Chemchem et al., wheat yield dataset was predicted using classical ML methods and ML methods combined with SMOTE over-sampling. The ML methods used in the study were Support Vector Machine (SVM), k-Nearest Neighbor (KNN), Gradient Boosting, Adaptive Boosting (AdaBoost), Decision Tree (DT), Random Forest (RF), Multi-Layer Perceptron (MLP), and Naïve Bayes (NB). The best result among the classical ML methods was achieved with RF, which had an accuracy ratio of 99,40%. Similarly, the best result for the combined SMOTE sampling based ML methods was also obtained with RF, measuring 99,22% (Chemchem et al., 2019). Xiao et al. studied corn disease identification using an improved Gradient Boosting Decision Tree (GBDT) method. They generated synthetic data by applying SMOTE sampling to the corn disease dataset. The performance of GBDT method was compared with other ML methods, such as Logistic Regression (LR), Linear SVM, Radial Basis Function (RBF) SVM, DT, and NB. The accuracy ratio of the GBDT method was measured as 92.51% (Xiao et al., 2019). In another study, Recurrent Neural Networks (RNN), Long Short-Term Memory (LSTM), Gated Recurrent Unit (GRU) were applied to tomato powdery mildew disease. The best accuracy among methods was achieved with RNN for the unbalanced dataset, measuring 89,80%. For the unbalanced dataset, both LSTM and GRU methods achieved the same accuracy of 56% (Varshney et al., 2021). A study conducted by Divakar et al., aimed to detect plant disease using a deep learning method. The unbalanced apple disease dataset was balanced using SMOTE over-sampling. The study examined the performance of several pre-trained CNN models for the apple disease classification. Among these models, DenseNet achieved the highest accuracy ratio of 92,88% (Divakar et al., 2021). Wang et al. proposed the prediction of chronic obstructive pulmonary disease (COPD) using unbalanced data. After evaluating the performance of classical ML methods, SMOTE over-sampling was applied to the unbalanced dataset. Among the classical ML methods, natural gradient boosting (NGBoost) achieved the highest accuracy rate of 91,1%. After applying SMOTE sampling, Extreme Gradient Boosting (XGBoost) achieved an accuracy rate of 80,5% (Wang et al., 2023). An application was developed by Bhardwaj et al., for the prediction of wine quality using ML methods. They generated 2381 samples from 12 original samples using the SMOTE method. RF, XGBoost, AdaBoost, KNN, and DT methods were used for the study. The best accuracy among the methods was achieved by AdaBoost and RF, both with an accuracy ratio of 100% (Bhardwaj et al., 2022). In another study conducted by Umer et al., the analysis of scientific paper citation was performed using textual features and SMOTE re-sampling techniques. ML models such as Extra Tree Classifier (ETC), DT, LR, AdaBoost, SVM, and RF were utilized in the study. The best accuracy ratio was achieved by ETC with 98,28% (Umer et al., 2021). Çelik et al., conducted a study on text classification using re-sampling techniques with SMS data. LR, KNN, SVM, DT, RF, XGBoost, and ANN models were used in the study. The dataset included spam, advertisement, and normal SMS data. LR yielded the best results when the SMOTE technique was applied to the SMS data. The accuracy ratio of LR was 80,07% (Çelik, 2020).

The summary of literature has presented the performances of classical ML methods using SMOTE techniques. The accuracy ratios of all examined studies were checked.

In this current study, ML methods based on SMOTE sampling were developed to classify different types of *dates* and *pistachios*.

3. MATERIALS AND METHODS

3.1. Materials

The data used for this study are the data shared by (Koklu et al., 2021). There are 34 genetic varieties of 7 different date fruits, which include Kirmizi Pistachio *(K_PISTA),* Siirt Pistachio *(S_PISTA)*, Barhee (*Berhi*), Deglet Nour (*Deglet*), Sukkary (*Dokol*), Rotab Mozafati (*Iraqi*), Ruthana (*Rotana*), Safawi (Safavi) and Sagai (*Sogay*), resulting in a total of 3046 date and pistachio fruit data. The number of data points and the percentage of the classes for date and pistachio fruits in the unbalanced dataset are given in Table 1. The list of features used for genetic varieties of fruits is given in Table 2. The plot showing the unbalanced distribution of the dataset is displayed in Figure 2. To examine the performance of ML methods, 80% of the data is allocated for training, and 20% is reserved for testing.

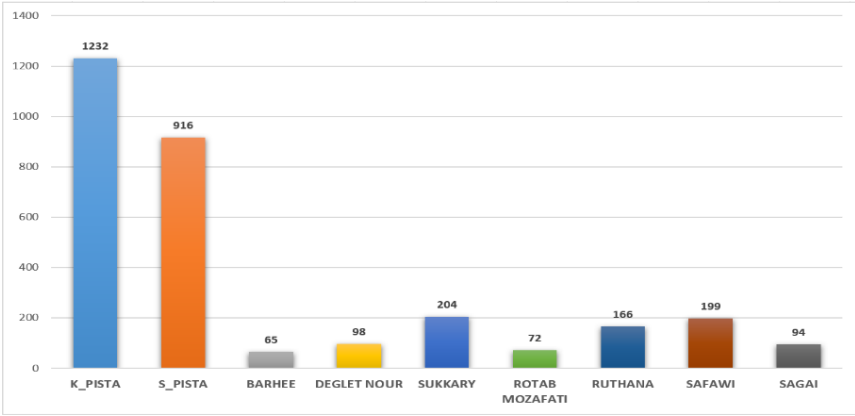

Figure 2. The Data Distributions of *Dates* and *Pistachios* on the Unbalanced Dataset.

Table 1. Number of Fruit Species Data in the Unbalanced Dataset.

Fruit Names	Total Data Number	The Percentage of the Data
Barhee	65	0,021339
Deglet Nour	98	0,032173
Sukkary	204	0,066973
Rotab Mozafati	72	0,023637
Ruthana	166	0,054497
Safawi	199	0,065331
Sagai	94	0,030860
Kirmizi Pistachio (K_PISTA)	1232	0,404464
Siirt_Pistachio (S_PISTA)	916	0,300722

TOTAL	***3046***	***1,000000***

Table 2. The Feature Names of Fruits' Genetic Varieties

Major Axis	Skew RR	Kurtosis RR
Minor Axis	Skew RG	Kurtosis RG
Eccentricity	Skew RB	Kurtosis RB
Equivalent Diameter	Roundness	Mean RR
StdDev RR	Compactness	Mean RG
StdDev RG	Shape Factor 1	Mean RB
StdDev RB	Shape Factor 2	Extent
Solidity	Shape Factor 3	Aspect Ratio
Perimeter	Shape Factor 4	

3.2. Hyperparameter Tuning of ML Methods

Optimizing hyperparameters in a machine learning (ML) algorithm is an essential part of the training process and is considered a fundamental practice for achieving effective implementation (Belete & Huchaiah, 2022). There are several hyperparameter optimization (HPO) algorithms. Some of these include Grid Search, Random Search, Bayesian Search, and Genetic Algorithms. In this study, the Randomized Search algorithm (RSA) has been used. The main purpose of RSA is to sample a certain number of value configurations in the hyperparameter space. This technique allows better exploration of the hyperparameter space when the number of combinations can be very large (Sher et al., 2023). RSA reduces the computational burden associated with combinations of values in the hyperparameter space and reaches the optimum values by random sampling (Anugerah Simanjuntak et al., 2024). This random sampling helps to avoid local optima that trap deterministic methods such as Grid Search (Boyd et al., 2024).

Significant changes were observed in the performance measures of the models after hyperparameter optimization. Especially for SVM, C and Gamma parameters significantly affected the accuracy rates by determining the decision boundaries of the model. The RBF kernel function gave the most successful result on the multi-class imbalanced dataset. In the Random Forest model, the number of trees and the splitting criterion (Gini/Entropy) are important factors, and it was observed that the Gini criterion gave better results after SMOTE. In the KNN model, the number of neighbors, distance criterion (Manhattan) and weighting (Distance) choices increased the success of the model. The number of hidden layers and the activation function (Tanh) were critical variables in the MLP model. The analysis shows that hyperparameter optimization plays an important role in improving model accuracy and generalization ability.

In this study, since the combination of values in the hyperparameter space is large, the optimal hyperparameters are determined by Random Search. Additionally, cross validation was applied during the HPO of all ML models, and a cross-validation value of 5 was chosen. HPO was conducted on a machine equipped with 64 GB RAM, and NVIDIA GeForce RTC 3060 graphics cards, and running on the Ubuntu operating system.

3.3. Proposed Method

Classifying unbalanced datasets has long been a significant challenge in machine learning. One effective approach to convert unbalanced datasets into balanced ones is through the application of the SMOTE technique. The Synthetic Minority Over-Sampling Technique is a widely adopted statistical method to handle class unbalance in machine learning (Özdemir et al., 2021). This technique was first introduced by (Chawla et al., 2002). and was inspired by a method developed by Ha and Bunke in 1997 (Ha & Bunke, 1997), which was successful in handwriting recognition tasks. Rather than using traditional data augmentation techniques like image rotation, as proposed by Ha & Bunke, SMOTE focuses on augmenting each individual sample from the minority class. The SMOTE method helps balance class distributions by generating synthetic instances for the minority class (Yavaş et al., 2020). These synthetic samples are created through interpolation between existing minority class samples and incorporated into the balanced dataset. This technique helps to balance the class distribution and reduces the model's bias toward the majority class. The steps of the SMOTE method are outlined in Table 3, and its formula (1) is provided below.

$$x_{new} = x_i + (x_j - x_i) * \beta \quad (1)$$

SMOTE technique has some disadvantages. One of them is that the synthetic data is also produced within the majority class region (Sağlam & Cengiz, 2022). Another issue is the creation of new instances that do not actually exist, solely based on existing instances (Özdemir et al., 2021). Finally, it can lead to a distortion of the true class distribution. In this study, the over-sampling method was chosen to preserve the samples in the balanced dataset. The dataset used in this study consists of unbalanced data with different classes. Referring to Table 1, *K_PISTA* accounts for 40% of the dataset, while *Barhee* of Date only represents 2%. In the balanced dataset, the class with the highest number of data points is Kirmizi Pistachio (K_PISTA), which has 1232 data points.

Table 3. The Working Principles of SMOTE Technique.

Process	Definition
P1:	The k nearest neighbors are looking for every feature belonging to minority class.
P2:	Taken the difference between a feature belonging to the minority class and its k nearest neighbor
P3:	A random number β is chosen between 0 and 1. Then, multiply this number by the value found in ***STEP P2***.
P4:	New features are obtained using Equation 1 (Yavaş et al., 2020).
P5:	Steps 1, 2, 3 and 4 are repeated to generate number of features.

After implementing SMOTE, the total number of data points in the balanced dataset decreased from 3046 to 2250. The values for all classes were adjusted to align with the class that had the highest count, which is K_PISTA. The balance achieved in the dataset after applying the SMOTE technique is shown in Table 4. Additionally, Figure 3

presents a plot depicting the balanced distribution of the dataset following the application of SMOTE.

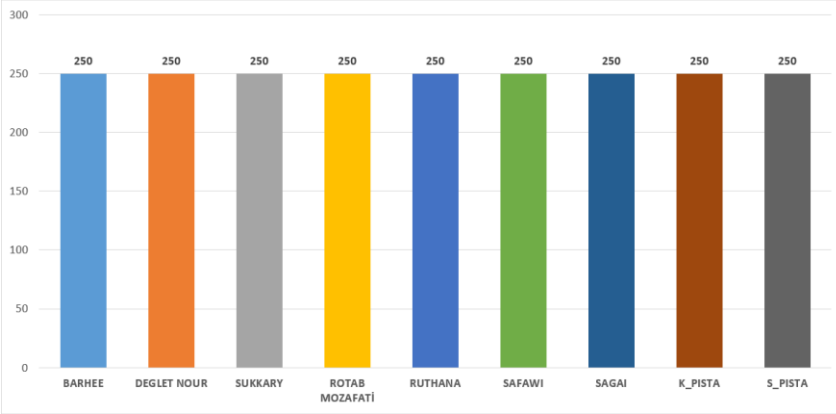

Figure 3. The Data Distributions of *Date* and *Pistachio* Dataset After Applying SMOTE Technique.

This data distribution leads to misleading results during classification with machine learning models. To obtain more accurate results in the classification, the data distribution of the dataset should be balanced.

Table 4. Number of Fruit Species Data in the Dataset After Applying SMOTE.

Fruit Names	**Total Data Number**	**The Percentage of the Data**
Barhee	250	0,111111
Deglet Nour	250	0,111111
Sukkary	250	0,111111
Rotab Mozafati	250	0,111111
Ruthana	250	0,111111
Safawi	250	0,111111
Sagai	250	0,111111
Kirmizi Pistachio (K_PISTA)	250	0,111111
Siirt_Pistachio (S_PISTA)	250	0,111111
TOTAL	***2250***	***1,000000***

There are several methods to achieve data balance in the dataset, and one of them is the SMOTE technique. After applying SMOTE, the similarity relationship between the original and synthetic data was analyzed using Jensen-Shannon Divergence (JSD). JSD is an explicit measure of the similarity between two probability distributions characterized by their symmetric and bounded properties (Menéndez et al., 1997). The distributions between synthetic and original data are shown in Figure 4 with the T-distributed Stochastic Neighbor Embedding (t-SNE) algorithm. The original and synthetic data overlap in some regions. However, in some regions the synthetic data appears to have different distributions than the original data. The red dots appear to be

clustered separately from the blue dots, which may indicate that the synthetic data contains deviations from the original distribution or forms new patterns. These deviations were analyzed with Jensen-Shannon Divergence Distance (JSD) and the deviation values are shown in Table 5.

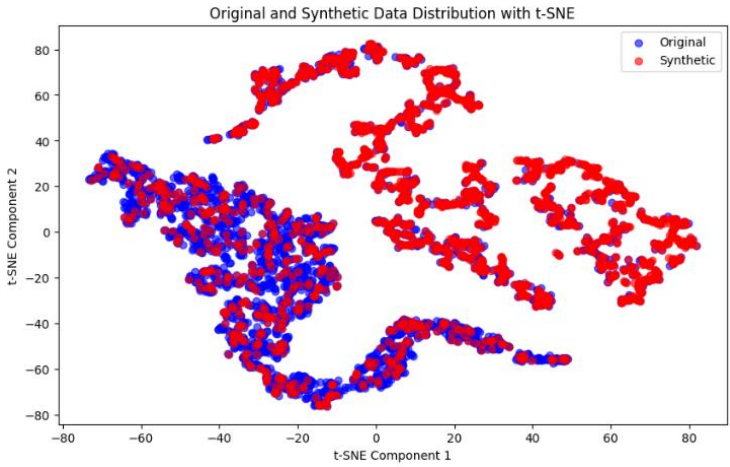

Figure 4. Representing the Distributions Between Synthetic and Original Data With t-SNE.

Those with a JSD distance greater than 0,3 between the synthetic and original data were not included in the training of the models because they showed significant separation. Values greater than 0,3 are marked with "*" symbol in Table 5.

Table 5. Jensen-Shannon Distance of Synthetic Data Generated With SMOTE to the Original Data.

Feature Names	**Jensen-Shannon Divergence Distance**	**Feature Names**	**Jensen-Shannon Divergence Distance**
Perimeter	0,2568	Mean RR*	0,3589
Major Axis	0,3420	Mean RG*	0,3588
Minor Axis*	0,3527	Mean RB*	0,3574
Eccentricity	0,2781	StdDev RR	0,2522
Equivalent Diameter*	0,3513	StdDev RG	0,1502
Extent	0,1166	StdDev RB	0,1852
Aspect Ratio	0,0021	Skew RR	0,2093
Roundness	0,3173	Skew RG	0,2613
Compactness	0,2953	Skew RB	0,2149
Shapefactor_1	0,0021	Kurtosis RR	0,0975
Shapefactor_2	0,3443	Kurtosis RG	0,1629
Shapefactor_3	0,2976	Kurtosis RB	0,1589
Shapefactor_4	0,2243	Solidity	0,2354

3.3.1. Support Vector Machine

Support Vector Machine (SVM) is a supervised machine learning method initially proposed by Vapnik (Cortes & Vapnik, 1995). It is designed to tackle quadratic optimization challenges and is widely employed for classification purposes (Cortes & Vapnik, 1995). The main goal of SVM is to create a decision boundary that effectively divides the various classes. This boundary is represented by a hyperplane, which is placed to maximize the margin between the different classes. The support vectors, which are the data points nearest to this decision boundary, are critical in determining the class separation. One key advantage of SVM is its efficient use of memory during the classification phase, as it works with a subset of the training data. The hyperparameter settings for the SVM model are listed in Table 6.

Table 6. The Best Parameter for SVM Model

ML Models	The Best Hyperparameter for the Unbalanced Dataset	The Best Hyperparameter after applying SMOTE Technique
RBF-SVM	**C**: 8180,247659224931 **Gamma**: 6,952230531190703 **Class Weight**: None **Decision Function Shape**: Ovo **Optimization Time**: 62,45651459693909 seconds	**C**: 8180,247659224931 **Gamma**: 6,952230531190703 **Class Weight**: None **Decision Function Shape**: Ovo **Optimization Time**: 22,32699489593506 seconds
Linear-SVM	**C**: 359,5227379674209 **Gamma**: 542,6447347075766 **Class Weight**: None **Decision Function Shape**: Ovo **Optimization Time**: 30,011558771133423 seconds	**C**: 113,63644767419068 **Gamma**: 6,952230531190703 **Class Weight**: Balanced **Decision Function Shape**: Ovr **Optimization Time**: 9,646587371826172 seconds
Polynomial-SVM	**C**: 7282,263486118596 **Gamma**: 632,3059305935794 **Class Weight**: Balanced **Decision Function Shape**: Ovo **Optimization Time**: 1075,8052697181702 seconds	**C**: 6334,137565104235 **Gamma**: 803,6721768991144 **Class Weight**: Balanced **Decision Function Shape**: Ovr **Optimization Time**: 7,240705251693726 seconds
Sigmoid-SVM	**C:** 3745,501188473625 **Gamma**: 731,994041811405 **Class Weight**: None **Decision Function Shape**: Ovo **Optimization Time**: 30,27118468284607 seconds	**C:** 3745,501188473625 **Gamma**: 731,994041811405 **Class Weight**: None **Decision Function Shape**: Ovo **Optimization Time**: 14,745770692825317 seconds

3.3.2. Random Forest

Random Forest (RF) is a supervised machine learning technique introduced by Breiman to solve classification and regression challenges (Breiman, 2001). It utilizes ensemble learning by creating a collection of decision trees, which are generated randomly and aggregated together. This method improves the model's classification performance by utilizing the combined insights from all the trees. The hyperparameter values for the RF model are provided in Table 7.

Table 7. The Best Parameters for RF Model.

ML Model	The Best Hyperparameter for the Unbalanced Dataset	The Best Hyperparameter after applying SMOTE Technique
Random Forest	**N Estimators:** 230 **Maximum Depth:** 17 **Minimum Samples Leaf:** 6 **Minimum Samples Split:** 4 **Max Features:** Sqrt **Bootstrap:** True **Criterion:** Entropy **Class Weight:** None **Maximum Samples:** 0,9997459059575176 **CCP Alpha:** 0,0013671964826997285 **Optimization Time:** 93,80823493003845 seconds	**N Estimators:** 165 **Maximum Depth:** 17 **Minimum Samples Leaf:** 3 **Minimum Samples Split:** 2 **Max Features:** Log2 **Bootstrap:** True **Criterion:** Gini **Class Weight:** Balanced **Maximum Samples:** 0,9699893371393026 **CCP Alpha:** 0,001545661652886743 **Optimization Time:** 68,85771036148071 seconds

3.3.3. K-Nearest Neighbor

The K-Nearest Neighbor (KNN) classifier is a method used to categorize samples by assigning them to the class of their closest, previously labeled neighbors (Cover & Hart, 1967). It relies on two key factors: distance and neighborhood (K) number. The distance measures the proximity between the predicted point and other points, while K determines the number of nearest neighbors considered for the prediction. The hyperparameter values of the KNN model are presented in Table 8.

Table 8. The Best Parameter for KNN Model.

ML Model	The Best Hyperparameter for the Unbalanced Dataset	The Best Hyperparameter after applying SMOTE Technique
K-Nearest Neighbor	**N Neighbors**: 15 **Weights**: Distance **Metric**: Manhattan **Algorithm**: Brute **Optimization Time**: 2,3644299507141113 seconds	**N Neighbors**: 11 **Weights**: Distance **Metric**: Manhattan **Algorithm**: Kd_Tree **Optimization Time**: 2,491448163986206 seconds

3.3.4. Decision Tree

Decision trees, consisting of root nodes, intermediate nodes, branches and leaves (Bakan & Kanbay, 2024), are a supervised machine learning algorithm that divides similar groups into smaller subsets (Quinlan, 1986). DTs are one of the easy-to-construct and understand models. A tree structure consists of leaf nodes labelled with different classes and internal nodes connected to two or more child nodes, forming a test node (Quinlan, 1996). The hyperparameter values of the DT model are presented in Table 9.

Table 9. The Best Parameter for DT Model.

ML Model	The Best Hyperparameter for the Unbalanced Dataset	The Best Hyperparameter after applying SMOTE Technique
Decision Tree	**Maximum Depth:** 13 **Minimum Samples Leaf:** 4 **Minimum Samples Split:** 15 **Max Features:** Log2 **Criterion:** Entropy **Class Weight:** None **Splitter:** Best **Maximum Leaf Nodes:** 12 **CCP Alpha:** 0,028181963210002724 **Optimization Time:** 1,9292397499084473 seconds	**Maximum Depth:** 11 **Minimum Samples Leaf:** 10 **Minimum Samples Split:** 16 **Max Features:** None **Criterion:** Gini **Class Weight:** Balanced **Splitter:** Best **Maximum Leaf Nodes:** 16 **CCP Alpha:** 0,0039236219609223855 **Optimization Time:** 6,359211444854736 seconds

3.3.5. Naïve Bayes

Naive Bayes (NB) is a straightforward probabilistic classifier based on Bayes' theorem (Aggarwal & Kaur, 2013). NB assumes that all features in a data set are independent. It is called "*naïve*" because of this assumption. However, this assumption is often not true in real world data. In a Naive Bayes classifier, the class label represents the hypothesis, while the feature values serve as the evidence. The classifier aims to determine the class label with the highest likelihood based on the given feature values. The hyperparameter settings for the NB model are in Table 10.

Table 10. The Best Parameter for NB Model.

ML Model	The Best Hyperparameter for the Unbalanced Dataset	The Best Hyperparameter after applying SMOTE Technique
Naïve Bayes	**Var Smoothing:** None **Priors**: 0,0007455074367977082 **Optimization Time**: 0,6440117359161377 seconds	**Var Smoothing:** None **Priors**: 0,004592489919658672 **Optimization Time**: 0,5791840553283691 seconds

3.3.6. Multi-Layer Perceptron

A Multilayer Perceptron (MLP) is a type of neural network model consisting of several layers of artificial neurons, commonly known as perceptron's. The input data is fed into the network and passed through multiple hidden layers before reaching the output layer (Shubhangi & Pratibha, 2021). Each layer performs a non-linear transformation on the input, allowing the network to capture complex and non-linear relationships between the input and output. The hyperparameter settings for the MLP model are provided in Table 11.

Table 11. The Best Parameter for MLP Model.

ML Model	The Best Hyperparameter for the Unbalanced Dataset	The Best Hyperparameter after applying SMOTE Technique
Multi-Layer Perceptron	**Hidden Layer Sizes**: (194, 138) **Activation Function**: Tanh **Solver**: Adam **Alpha**: 0,024598491974895578 **Learning Rate**: Invscaling **Maximum Iteration**: 100 **Optimization Time**: 510,3250524997711 seconds	**Hidden Layer Sizes**: (50, 105) **Activation Function**: Tanh **Solver**: Lbfgs **Alpha**: 0,09248643902204486 **Learning Rate**: Adaptive **Maximum Iteration**: 100 **Optimization Time**: 110,45194363594055 seconds

3.3.7. Logistic Regression

Logistic regression (LR) was introduced by David Cox in 1958 (Cox, 1958). LR is a classification technique typically used when the dependent variable has two possible outcomes (Prasetio & Harlili, 2016). It is a straightforward and efficient algorithm that is easy to implement and can efficiently handle particularly large datasets with minimal computational effort. To prevent overfitting, LR can be regularized by adding a penalty term to the cost function. While it is primarily used for binary classification, LR can also be adapted for multi-class classification problems using methods such as One-vs-All or SoftMax Regression. The hyperparameter settings for the LR model are shown in Table 12.

Table 12. The Best Parameter for MLP Model.

ML Model	The Best Hyperparameter for the Unbalanced Dataset	The Best Hyperparameter after applying SMOTE Technique
Logistic Regression	**C**: 8,075401551640624 **Penalty**: L1 **Solver**: Saga **Maximum Iteration**: 1000 **Optimization Time**: 77,50988411903381 seconds	**C**: 8,075401551640624 **Penalty**: L1 **Solver**: Saga **Maximum Iteration**: 1000 **Optimization Time**: 60,024758100509644 seconds

4. RESULTS

This section of the study presents the evaluation metrics and descriptions of the model's classification performances, including the classification performance results of the models before and after applying SMOTE.

4.1. Evaluation Metrics

In this section of the paper, the performance measures and results of the proposed model are presented in detail. The confusion matrix, accuracy, balanced accuracy, sensitivity, specificity, precision, F-score, Cohen's kappa score, and ROC accuracy score have been calculated for all models. A confusion matrix (Townsend, 1971) is depicted in Figure 5.

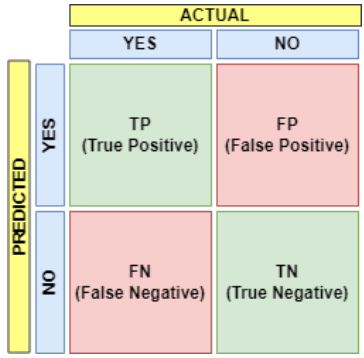

Figure 5. Confusion Matrix.

Accuracy (**A**): It measures the number of actual data instances over the total number of data instances.

$$\mathrm{A} = \frac{TP + TN}{TP + TN + FP + FN} \tag{2}$$

Precision (**P**): It measures how many of the positive predictions are made correctly.

$$P = \frac{TP}{TP + FP} \tag{3}$$

Sensitivity (**S**): It measures the number of positive cases predicted from all positive predictions.

$$\mathrm{S} = \frac{TP}{TP + FN} \tag{4}$$

Specificity (**SP**): It measures how many negative predictions are made correctly.

$$SP = \frac{TN}{TN + FP} \tag{5}$$

F1-Score (**F**): It measures harmonic average precision and sensitivity.

$$F = 2 * \frac{P * R}{P + R} \tag{6}$$

Balanced Accuracy (**BA**): It measures arithmetic meaning of sensitivity and specificity.

$$BA = \frac{R + S}{2} \tag{7}$$

ROC Accuracy Score (**ROC AUC**): It measures the area under the curve and compares the relationship via Precision and specificity.

Cohen's Kappa (**K**): It is a statistical method that measures the agreement between two values.

As a result of hyperparameter optimization, significant improvements in model performance were observed. For example, optimizing C and Gamma values in the RBF-SVM model improved accuracy from 92,62% to 95,55%. Similarly, optimizing the number of trees and splitting criterion in the Random Forest model resulted in a 4,5% increase in accuracy. The Linear-SVM model, which performed the best before SMOTE, was the least affected by data imbalance. However, the increase in the accuracy of the RBF-SVM model after SMOTE shows that it can better model non-linear relationships.

4.2. Results of Unbalanced Dataset

In this part of the study, the performance evaluations of ML models are included. A, P, S, SP and F results for the unbalanced dataset in all models are given in Table 13. BA, ROC AUC and K scores are given in Table 14. Among the models, Linear-SVM emerged as the top performer. The confusion matrix of Linear-SVM is presented in Figure 6. The ROC curve for all presented is shown in Figure 7 and the sub classification results are presented in Table 15.

Table 13. The Performance Evaluations of the Unbalanced Dataset.

ML Methods	A	P	S	SP	F
RBF SVM	0,9163	0,9186	0,9163	0,9672	0,9167
Linear SVM	***0,9262***	***0,9265***	***0,9262***	***0,9715***	***0,9260***
POLY SVM	0,9016	0,9034	0,9016	0,9655	0,9016
Sigmoid SVM	0,4327	0,1873	0,4327	0,5672	0,2614
RF	0,8983	0,8979	0,8983	0,9618	0,8976
DT	0,8229	0,7926	0,8229	0,9393	0,8050
KNN	0,9094	0,9104	0,9098	0,9679	0,9085
NB	0,8688	0,8784	0,8688	0,9548	0,8701
MLP	0,9032	0,9036	0,9032	0,9644	0,9026
LR	0,9260	0,9257	0,9262	0,9702	0,9253

Table 14. Other Performance Evaluations of the Unbalanced Dataset.

ML Methods	BA	ROC AUC	K
RBF SVM	0,9057	0,9463	0,8845
Linear SVM	***0,8853***	***0,9370***	***0,8978***
POLY SVM	0,8536	0,9194	0,8642
Sigmoid SVM	0,1111	0,5000	0,0000
RF	0,8464	0,9154	0,8594
DT	0,6823	0,8279	0,7547
KNN	0,8398	0,9131	0,8753
NB	0,8234	0,9019	0,8197
MLP	0,8500	0,9290	0,8663
LR	0,8947	0,9416	0,8978

Table 13 and Table 14 shows that Linear-SVM has the highest accuracy rate (92,62%). It is also the most successful model overall with the highest sensitivity (92,62%), specificity (97,15%) and Cohen's Kappa value (89,78%). The RBF-SVM model is also very successful and shows the second-best performance with an accuracy of 91,63%. The ROC AUC value (94,63%) is one of the highest. The sigmoid SVM model gave the worst result. With an accuracy of 43,27% and an ROC AUC of 50,00%, it behaves almost like a random prediction model. The Decision Tree (DT) model performed relatively poorly. Its accuracy is 82,29%, which is significantly lower than the other models.

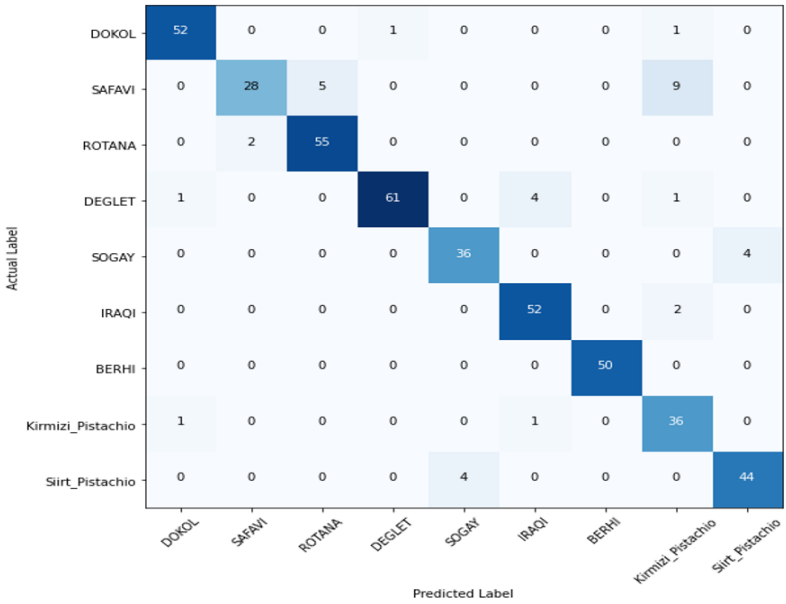

Figure 6. The Confusion Matrix of the Best ML Model (Linear-SVM) in the Unbalanced Dataset.

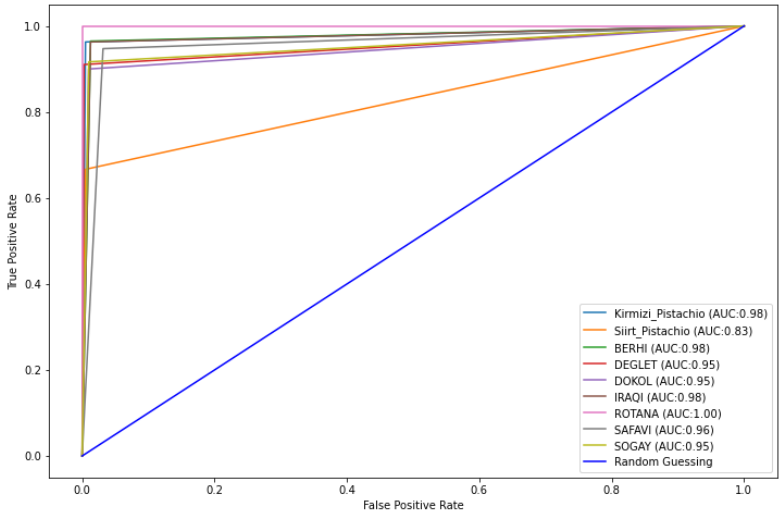

Figure 7. The ROC Curve of the Best ML Model (Linear-SVM) in the Unbalanced Dataset.

When Table 15 is analyzed, the precision and sensitivity rates are relatively low in some classes such as Deglet and Sagay. This is since some classes contain fewer samples than others due to the unbalanced dataset. The Safavi and Rotana classes were classified with 100% accuracy. However, this may also be due to the small number of samples. Overall, the Linear SVM model subclassified fruit types quite successfully. However, it can be said that the model was affected by the imbalanced dataset due to low sensitivity rates in some classes.

Table 15. Sub-classification Results for the Best ML Model on the Unbalanced Dataset.

Class	**P**	**S**	**SP**	**F**	**Test Case**
Berhi	0,7500	0,9000	0,9950	0,8181	10
Deglet	0,7647	0,7222	0,9932	0,7428	18
Dokol	0,9215	0,9791	0,9928	0,9494	48
Iraqi	0,9285	0,8125	0,9983	0,8666	16
Rotana	1,0000	0,9655	1,0000	0,9824	29
Safavi	1,0000	1,0000	1,0000	1,0000	43
Sagay	0,7857	0,7333	0,9949	0,7586	15
K_PISTA	0,9465	0,9393	0,9595	0,9429	264
S_PISTA	0,9053	0,9161	0,9638	0,9107	167
The Average Result	***0,9265***	***0,9262***	***0,9715***	***0,9260***	***610***

Cross Validation is a method that tests how the model performs on different datasets by dividing the dataset into different subsets to evaluate the generalization ability of the models. In this study, 5 and 10-fold datasets were identified. Table 16 shows the cross-validation (CV) results of the machine learning models tested on the unbalanced

dataset. The Linear SVM model is the best performing model with both CV 5 (91,31%) and CV 10 (91,47%) accuracy rates. Since the standard deviation values are low, the model performs consistently across different data partitions. The RBF SVM model has an accuracy of 85,40% in CV 5, which increases to 88,03% in CV 10, but the standard deviation value (± 0,0274) is slightly high. This means that the model may be inconsistent across different datasets. The DT, RF, NB and MLP models show good generalization ability, although the accuracy is low. The most unsuccessful model was the Sigmoid SVM model.

Table 16. Cross Validation Scores of Models on Unbalanced Dataset.

ML Methods	Cross Validation Scores	
	Accuracy	
	CV 5	CV 10
RBF SVM	0,8540 Std: ± 0,0245	0,8803 Std: ± 0,0274
Linear SVM	0,9131 Std: ± 0,0152	0,9147 Std: ± 0,0262
POLY SVM	0,9049 Std: ± 0,0168	0,9131 Std: ± 0,0344
Sigmoid SVM	0,4327 Std: ± 0,0032	0,4327 Std: ± 0,0080
RF	0,8704 Std: ± 0,0158	0,8786 Std: ± 0,0304
DT	0,8114 Std: ± 0,0200	0,8049 Std: ± 0,0297
KNN	0,8622 Std: ± 0,0203	0,8622 Std: ± 0,0375
NB	0,8639 Std: ± 0,0133	0,8622 Std: ± 0,0329
MLP	0,8737 Std: ± 0,0152	0,8786 Std: ± 0,0245
LR	0,8967 Std: ± 0,0152	0,8950 Std: ± 0,0396

4.2. Results of Balanced Dataset

Table 17 shows the A, P, S, SP and F results of all models in the balanced data set after SMOTE is applied. BA, ROC AUC and K scores are given in Table 18. Among the models, RBF SVM emerged as the top performer. The confusion matrix of RBF SVM is presented in Figure 8. The ROC curve for best models is presented in Figure 9 and the classification report is presented in Table 19.

Table 17. The Performance Evaluations of the Balanced Dataset.

ML Methods	A	P	S	SP	F
RBF SVM	***0,9555***	***0,9558***	***0,9555***	***0,9947***	***0,9555***
Linear SVM	0,9333	0,9347	0,9333	0,9923	0,9330
POLY SVM	0,9511	0,9514	0,9511	0,9944	0,9510
Sigmoid SVM	0,0844	0,0071	0,0844	0,9155	0,0131
RF	0,9355	0,9366	0,9355	0,9924	0,9355
DT	0,8911	0,8960	0,8911	0,9873	0,8920
KNN	0,9377	0,9392	0,9377	0,9928	0,9380
NB	0,8844	0,8876	0,8844	0,9863	0,8852
MLP	0,9266	0,9285	0,9266	0,9903	0,9267
LR	0,9377	0,9389	0,9377	0,9929	0,9376

Table 18. Other Performance Evaluations of the Balanced Dataset.

ML Methods	BA	ROC AUC	Κ
RBF SVM	***0,9527***	***0,9736***	***0,9498***
Linear SVM	0,9267	0,9592	0,9247
POLY SVM	0,9479	0,9709	0,9448
Sigmoid SVM	0,1111	0,5000	0,0000
RF	0,9318	0,9619	0,9272
DT	0,8882	0,9373	0,8771
KNN	0,9329	0,9626	0,9297
NB	0,8787	0,9322	0,8695
MLP	0,9217	0,9504	0,9172
LR	0,9329	0,9626	0,9297

Looking at the results in Table 17, in general, the performance of all models has improved significantly. RBF-SVM achieved the highest accuracy rate (95,55%), so it can be considered the most successful model. POLY-SVM also has a very high accuracy rate (95,11%). Models such as Linear SVM, Random Forest and KNN also performed strongly (93%+ accuracy). The previously unsuccessful Sigmoid SVM again performed poorly (8,44% accuracy), even after SMOTE. As seen in Table 19, after SMOTE was applied, the models applied to the balanced dataset showed a significant improvement in all metrics. SMOTE generates synthetic examples for minority classes, allowing the model to generalize better to less common classes. A small performance degradation was observed for the Red Pistachio and Siirt Pistachio classes. This may be since these classes had quite a lot of examples before SMOTE and the synthetic examples after SMOTE did not fully preserve the class distinction. Sensitivity in the minority of classes increased significantly, making the model less biased.

Table 19. Sub-Classification Results for the Best ML Model After Applying SMOTE.

Class	P	S	SP	F	Test Case
Berhi	1,0000	0,9629	1,0000	0,9811	54
Deglet	0,9069	0,9285	0,9901	0,9176	42
Dokol	0,9642	0,9473	0,9949	0,9557	57
Iraqi	0,9705	0,9850	0,9947	0,9777	67
Rotana	1,0000	1,0000	1,0000	1,0000	54
Safavi	0,9800	0,9800	0,9975	0,9800	50
Sagay	0,9743	1,0000	0,9975	0,9870	38
K_PISTA	0,8750	0,8750	0,9878	0,8750	40
S_PISTA	0,8958	0,8958	0,9875	0,8958	48
The Average Result	***0,9558***	***0,9555***	***0,9947***	***0,9555***	***450***

According to Table 15 and Table 19, while the F1-Score of the Barhee class was 0,8181 before SMOTE was applied, it increased to 0,9811 after SMOTE. The main reason for this increase is that there were few examples of the Barhee class before SMOTE was applied and the model did not generalize this class well. When synthetic examples were added after SMOTE, the model was trained with more data and was able to classify

more successfully for this class. Precision increased by 33%, indicating that the false positive rate decreased and the model better identified the Barhee class. The same is true for Deglet Nour and Sagai. The increase in the F1-Score of the Barhee, Deglet Nour and Sagai classes can be attributed to the high similarity of the synthetic samples with the original data. According to the JSD analysis, there was no significant difference between synthetic and original examples for this class. Therefore, the model successfully learned the added synthetic examples and significantly improved its accuracy in this class. Looking at the cross-validation analysis to understand whether the high performance improvement, especially in these classes, is due to overlearning, it is seen that the model performs consistently across different data partitions. This suggests that the model does not overlearn and that the performance improvement is due to the improvement of the data distribution after SMOTE.

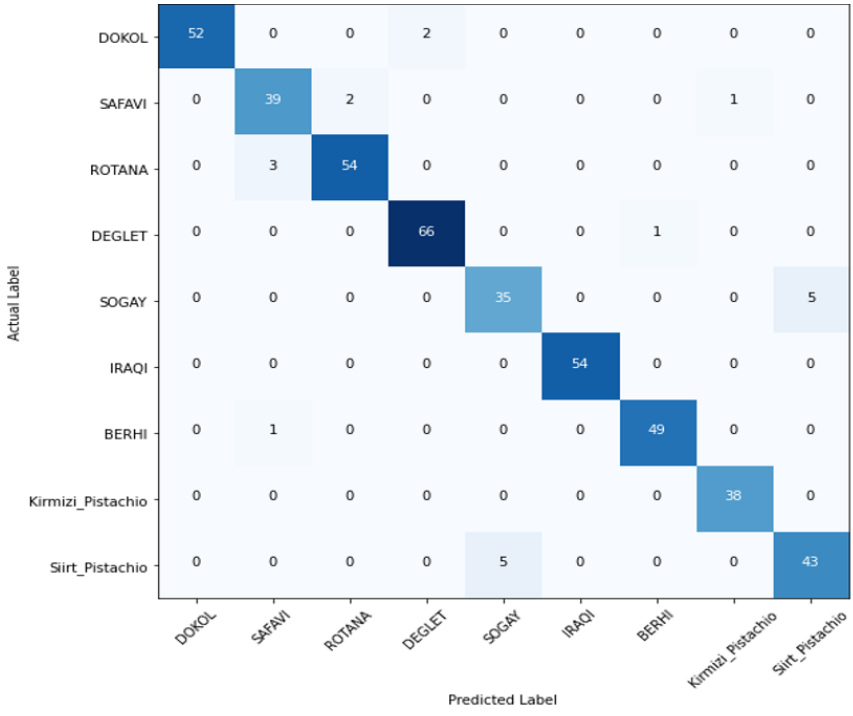

Figure 8. The Confusion Matrix of the Best ML Model After Applying SMOTE Technique.

The CV performance of the models on the balanced dataset was analyzed and the results are given in Table 18. Accuracy rates increased for Linear SVM, RBF SVM, POLY SVM and LR. This shows that the models can generalize better and give more consistent results in different data distributions. A significant decrease was observed in Sigmoid SVM. This indicates that the model is not suitable for the data. The CV 10 result of the MLP has decreased. This indicates that the model performs more variably against different data partitions and may have a higher variance.

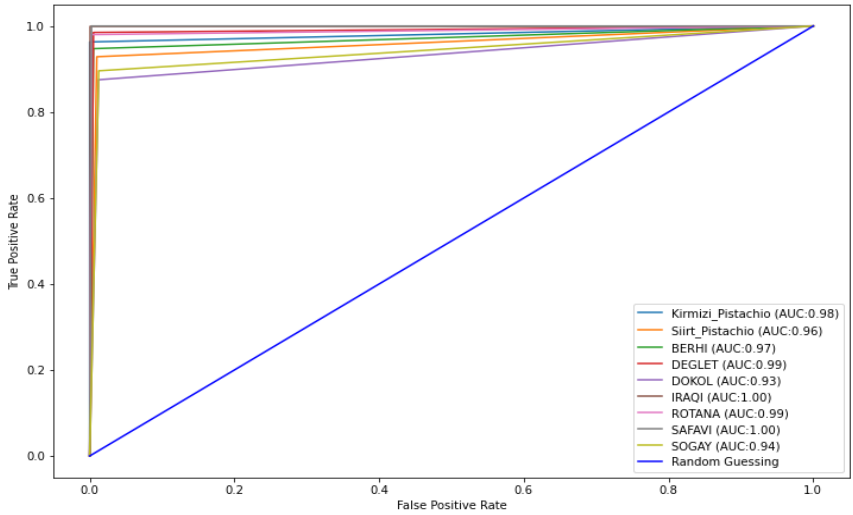

Figure 9. The ROC Curve of the Best ML Model After Applying SMOTE Technique.

Table 18. Cross Validation Scores of Models on Balanced Dataset.

ML Methods	Cross Validation Scores	
	Accuracy	
	CV 5	CV 10
RBF SVM	0,9022 Std: ± 0,0215	0,9066 Std: ± 0,0407
Linear SVM	0,9088 Std: ± 0,0163	0,9111 Std: ± 0,0371
POLY SVM	0,8933 Std: ± 0,0206	0,8955 Std: ± 0,0466
Sigmoid SVM	0,1488 Std: ± 0,0054	0,1488 Std: ± 0,0101
RF	0,8955 Std: ± 0,0166	0,8888 Std: ± 0,0281
DT	0,8155 Std: ± 0,0268	0,8200 Std: ± 0,0481
KNN	0,9000 Std: ± 0,0121	0,9111Std: ± 0,0314
NB	0,8755 Std: ± 0,0177	0,8866 Std: ± 0,0321
MLP	0,8777 Std: ± 0,0365	0,7644 Std: ± 0,0488
LR	0,9111 Std: ± 0,0272	0,9199 Std: ± 0,0361

5. CONCLUSION

This study investigated the performance of SMOTE-based machine learning models to address the imbalanced dataset problem. The experiments revealed that when imbalanced datasets are used, machine learning models show biases in certain classes and classification performance decreases. When the SMOTE technique is applied, however, the representation of minority classes increases, thereby improving the overall accuracy and sensitivity of the models to unbalance.

The results showed that the best performance was achieved by the RBF-SVM model with an accuracy of 95,55% on the dataset with SMOTE. The model that showed the

highest performance without SMOTE was Linear-SVM with an accuracy of 92,62%. However, the sigmoid SVM model performed poorly on both datasets. These results suggest that techniques to address data imbalance, such as SMOTE, can improve the performance of classification models on multiclass agricultural datasets.

The findings reveal that SMOTE-based data organization techniques offer a critical improvement for machine learning models working with imbalanced datasets. However, some drawbacks of SMOTE should also be considered; in particular, the new synthetic data may not be fully compatible with the original dataset, which may negatively affect the learning process in some models. Therefore, future work should complement the analysis with a comprehensive analysis of different variations of SMOTE, including hybrid oversampling methods or other balancing techniques.

Overall, this study has demonstrated that data imbalance removal is a critical element to improve the classification performance of machine learning models on agricultural products. In the future, a comparative study of different data imbalance removal approaches and their application on different types of agricultural datasets can increase the added value of the study.

Authors' Contribution
Corresponding author Fatih Bal contributes to the abstract, data collection and construction of the dataset, and designing the model for the study. Fatih Kayaalp contributes to the maturity of the study, introduction, literature review, and the preparation of the study's flow.

Available Materials
The date dataset is available in (Rıdvan Saraçoğlu, İlker Ali Özkan, 2021) and the pistachio dataset is available in (Koklu et al., 2021).

Conflict of Interest Statement
There is no conflict of interest between the authors.

Research and Publication Ethics Statement
The study complied with research and publication ethics.

REFERENCES

Aggarwal, S., & Kaur, D. (2013). Naive Bayes Classifier with Various Smoothing Techniques for Text Documents. *International Journal of Computer Trends and Technology (IJCTT*, *4*(4). http://ijcttjournal.org/Volume4/issue-4/IJCTT-V4I4P187.pdf

Anugerah Simanjuntak, Rosni Lumbantoruan, Kartika Sianipar, Rut Gultom, Mario Simaremare, Samuel Situmeang, & Erwin Panggabean. (2024). Research and Analysis of IndoBERT Hyperparameter Tuning in Fake News Detection. *Jurnal Nasional Teknik Elektro Dan Teknologi Informasi*, *13*(1), 60–67. https://doi.org/10.22146/jnteti.v13i1.8532

Aydın, B. S. Y. (2018). Siirt Yöresi Fıstık Yetiştiricilerinin Sulama Eğilimlerinin Belirlenmesi. *Süleyman Demirel Üniversitesi Ziraat Fakültesi Dergisi*, 119–127.

Bakan, Z., & Kanbay, F. (2024). Makine Öğrenmesi Yöntemleri ile Eğitim Başarısına Etki Eden Faktörlerin Modellenmesi. *İstanbul Ticaret Üniversitesi Fen Bilimleri Dergisi*, *23*(45), 27–41. https://doi.org/10.55071/ticaretfbd.1442084

Belete, D. M., & Huchaiah, M. D. (2022). Grid search in hyperparameter optimization of machine learning models for prediction of HIV/AIDS test results. *International Journal of Computers and Applications*, *44*(9), 875–886. https://doi.org/10.1080/1206212X.2021.1974663

Bhardwaj, P., Tiwari, P., Olejar, K., Parr, W., & Kulasiri, D. (2022). A machine learning application in wine quality prediction. *Machine Learning with Applications*, *8*, 100261. https://doi.org/10.1016/j.mlwa.2022.100261

Boyd, C., Brown, G. C., Kleinig, T. J., Mayer, W., Dawson, J., Jenkinson, M., & Bezak, E. (2024). Hyperparameter selection for dataset-constrained semantic segmentation: Practical machine learning optimization. *Journal of Applied Clinical Medical Physics*, *25*(12). https://doi.org/10.1002/acm2.14542

Breiman, L. (2001). Random Forests. *Machine Learning*, *45*(5–32). https://doi.org/https://doi.org/10.1023/A:1010933404324

Çağlar, A., Tomar, O., Vatansever, H., & Ekmekçi, E. (2017). Antepfıstığı (Pistacia vera L.) ve İnsan Sağlığı Üzerine Etkileri. *Akademik Gıda*, 436–447. https://doi.org/10.24323/akademik-gida.370408

Çelik, G. K. Ö. (2020). Yeniden Örnekleme Teknikleri Kullanarak SMS Verisi Üzerinde Metin Sınıflandırma Çalışması. *Erciyes Üniversitesi Fen Bilimleri Enstitüsü Dergisi*, *36*(3).

Chawla, N. V., Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P. (2002). SMOTE: Synthetic Minority Over-sampling Technique. *Journal of Artificial Intelligence Research*, *16*, 321–357. https://doi.org/10.1613/jair.953

Chemchem, A., Alin, F., & Krajecki, M. (2019). Combining SMOTE Sampling and Machine Learning for Forecasting Wheat Yields in France. *2019 IEEE Second International Conference on Artificial Intelligence and Knowledge Engineering (AIKE)*, 9–14. https://doi.org/10.1109/AIKE.2019.00010

Cortes, C., & Vapnik, V. (1995). Support-vector networks. *Machine Learning*, *20*(3), 273–297. https://doi.org/10.1007/BF00994018

Cover, T., & Hart, P. (1967). Nearest neighbor pattern classification. *IEEE Transactions on Information Theory*, *13*(1), 21–27. https://doi.org/10.1109/TIT.1967.1053964

Cox, D. R. (1958). The Regression Analysis of Binary Sequences. *Journal of the Royal Statistical Society. Series B (Methodological)*, *20*(2), 215–242.

Divakar, S., Bhattacharjee, A., & Priyadarshini, R. (2021). Smote-DL: A Deep Learning Based Plant Disease Detection Method. *2021 6th International Conference for Convergence in Technology (I2CT)*, 1–6. https://doi.org/10.1109/I2CT51068.2021.9417920

Echegaray, N., Gullón, B., Pateiro, M., Amarowicz, R., Misihairabgwi, J. M., & Lorenzo, J. M. (2023). Date Fruit and Its By-products as Promising Source of Bioactive Components: A Review. *Food Reviews International*, *39*(3), 1411–1432. https://doi.org/10.1080/87559129.2021.1934003

Ha, T. M., & Bunke, H. (1997). Off-line, handwritten numeral recognition by perturbation method. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *19*(5), 535–539. https://doi.org/10.1109/34.589216

Koklu, M., Kursun, R., Taspinar, Y. S., & Cinar, I. (2021). Classification of Date Fruits into Genetic Varieties Using Image Analysis. *Mathematical Problems in Engineering*, *2021*, 1–13. https://doi.org/10.1155/2021/4793293

Menéndez, M. L., Pardo, J. A., Pardo, L., & Pardo, M. C. (1997). The Jensen-Shannon divergence. *Journal of the Franklin Institute*, *334*(2), 307–318. https://doi.org/10.1016/S0016-0032(96)00063-4

Özdemir, A., Polat, K., & Alhudhaif, A. (2021). Classification of imbalanced hyperspectral images using SMOTE-based deep learning methods. *Expert Systems with Applications*, *178*, 114986. https://doi.org/10.1016/j.eswa.2021.114986

Prasetio, D., & Harlili, D. (2016). Predicting football match results with logistic regression. *2016 International Conference On Advanced Informatics: Concepts, Theory And Application (ICAICTA)*, 1–5. https://doi.org/10.1109/ICAICTA.2016.7803111

Quinlan, J. R. (1986). Induction of decision trees. *Machine Learning*, *1*(1), 81–106. https://doi.org/10.1007/BF00116251

Quinlan, J. R. (1996). Learning decision tree classifiers. *ACM Computing Surveys*, *28*(1), 71–72. https://doi.org/10.1145/234313.234346

Rıdvan Saraçoğlu İlker Ali Özkan, M. K. (2021). Classification of Pistachio Species Using Improved K-NN Classifier. *Journal of Nutrition and Internal Medicine*, *23*(1). https://doi.org/https://doi.org/10.23751/pn.v23i2.9686

Sağlam, F., & Cengiz, M. A. (2022). A novel SMOTE-based resampling technique trough noise detection and the boosting procedure. *Expert Systems with Applications*, *200*, 117023. https://doi.org/10.1016/j.eswa.2022.117023

Sher, M., Minallah, N., Ahmad, T., & Khan, W. (2023). Hyperparameters analysis of long short-term memory architecture for crop classification. *International Journal of Electrical and Computer Engineering (IJECE)*, *13*(4), 4661. https://doi.org/10.11591/ijece.v13i4.pp4661-4670

Shubhangi, D. C., & Pratibha, A. K. (2021). Asthma, Alzheimer's and Dementia Disease Detection based on Voice Recognition using Multi-Layer Perceptron Algorithm. *2021 International Conference on Innovative Computing, Intelligent Communication and Smart Electrical Systems (ICSES)*, 1–7. https://doi.org/10.1109/ICSES52305.2021.9633923

Townsend, J. T. (1971). Theoretical analysis of an alphabetic confusion matrix. *Perception & Psychophysics*, *9*(1), 40–50. https://doi.org/10.3758/BF03213026

Umer, M., Sadiq, S., Missen, M. M. S., Hameed, Z., Aslam, Z., Siddique, M. A., & NAPPI, M. (2021). Scientific papers citation analysis using textual features and SMOTE resampling techniques. *Pattern Recognition Letters*, *150*, 250–257. https://doi.org/10.1016/j.patrec.2021.07.009

Varshney, T., Chug, A., & Singh, A. P. (2021). Deep Learning Models for Prediction of Tomato Powdery Mildew Disease. *2021 8th International Conference on Signal Processing and Integrated Networks (SPIN)*, 1036–1041. https://doi.org/10.1109/SPIN52536.2021.9566132

Wang, X., Ren, H., Ren, J., Song, W., Qiao, Y., Ren, Z., Zhao, Y., Linghu, L., Cui, Y., Zhao, Z., Chen, L., & Qiu, L. (2023). Machine learning-enabled risk prediction of chronic obstructive pulmonary disease with unbalanced data. *Computer Methods and Programs in Biomedicine*, *230*, 107340. https://doi.org/10.1016/j.cmpb.2023.107340

Xiao, T., Liu, H., & Cheng, Y. (2019). Corn Disease Identification Based on improved GBDT Method. *2019 6th International Conference on Information Science and Control Engineering (ICISCE)*, 215–219. https://doi.org/10.1109/ICISCE48695.2019.00051

Yavaş, M., Güran, A., & Uysal, M. (2020). Covid-19 Veri Kümesinin SMOTE Tabanlı Örnekleme Yöntemi Uygulanarak Sınıflandırılması. *European Journal of Science and Technology*, 258–264. https://doi.org/10.31590/ejosat.779952

Yıldız, M. S. N. (2019). Hurma Ağacının (Phoenix dactylifera L.) İklim ve Toprak İstekleri. *International Journal of Engineering, Design and Technology*, *1*(2), 64–70.