

T.C.
VAN YÜZÜNCÜ YIL ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ
ZOOTEKNİ ANABİLİM DALI

**TARIMDA MAKİNE ÖĞRENMESİ ALGORİTMALARININ
UYGULAMALARI**

YÜKSEK LİSANS TEZİ

HAZIRLAYAN: SAMET ERİŞ
DANIŞMAN : Prof. Dr. ABDULLAH YEŞİLOVA

VAN-2021

T.C.
VAN YÜZÜNCÜ YIL ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ
ZOOTEKNİ ANABİLİM DALI

**TARIMDA MAKİNE ÖĞRENMESİ ALGORİTMALARININ
UYGULAMALARI**

YÜKSEK LİSANS TEZİ

HAZIRLAYAN: Samet ERİŞ

VAN-2021

KABUL VE ONAY SAYFASI

Zootekni Anabilim Dalı'nda Prof. Dr. Abdullah YEŞİLOVA danışmanlığında, Samet ERİŞ tarafından sunulan “Tarımda Makine Öğrenmesi Algoritmalarının Uygulamaları” isimli bu çalışma Lisansüstü Eğitim ve Öğretim Yönetmeliği'nin hükümleri gereğince 05/07/2021 tarihinde aşağıdaki jüri tarafından oy birliği ile başarılı bulunmuş ve yüksek lisans tezi olarak kabul edilmiştir.

Başkan: Prof. Dr. Abdullah YEŞİLOVA

İmza:

Üye: Doç. Dr. Gazel SER

İmza:

Üye: Doç. Dr. Yılmaz KAYA

İmza:

Fen Bilimleri Enstitüsü Yönetim Kurulu'nun 27/07/2021 tarih ve 2021/36-1 sayılı kararı ile onaylanmıştır.

İmza

Prof. Dr. Harun AKKUŞ
Enstitü Müdürü

TEZ BİLDİRİMİ

Tez içindeki bütün bilgilerin etik davranış ve akademik kurallar çerçevesinde elde edilerek sunulduğunu, ayrıca tez yazım kurallarına uygun olarak hazırlanan bu çalışmada bana ait olmayan her türlü ifade ve bilginin kaynağına eksiksiz atıf yapıldığını bildiririm.

(İmza)

Samet ERİŞ

ÖZET

TARIMDA MAKİNE ÖĞRENMESİ ALGORİTMALARININ UYGULAMALARI

ERİŞ, Samet

Yüksek Lisans Tezi, Zootekni Anabilim Dalı
Tez Danışmanı: Prof. Dr. Abdullah YEŞİLOVA
Temmuz 2021, 81 sayfa

Bu çalışmada bir tarım veri seti üzerinde yaygın olarak kullanılan makine öğrenmesi algoritmalarından en doğru tahmin üreten model tespiti yapılmıştır. Tespit edilen model ile tarımsal ürünlerin toprak çeşitleri için daha verimli ürün seçimi veya alternatif yetiştirilebilecek tarım ürünleri konusunda kullanıcıya tavsiye veren bir tahminleme yazılımı yapılmıştır. Topraktaki azot, potasyum, fosfor, nem, ph, sıcaklık ortalaması, yağış miktarı ortalaması gibi özellikleri içeren 22 farklı tarım ürününün bulunduğu açık kaynak lisanslı bir veri seti kullanılmıştır. Sınıflama işlemini Orange yazılımı kullanarak destek vektör makineleri, naive bayes sınıflandırıcı, karar ağaçları, adaboost, olasılıksal dereceli azalma, lojistik regresyon, çok katmanlı algılayıcı, k-en yakın komşu ve rastgele orman algoritmaları modellenmiştir. Bu dokuz farklı modelin Orange yazılımı kullanılarak eğitim ve test süresi, AUC değeri, doğruluk skoru, F1 skoru, kesinlik değerleri, hassasiyet değerleri, kayıp oranları ve özgüllük değerleri hesaplanmıştır. Bu sonuçlara göre rastgele orman en iyi model olarak tespit edilmiştir. Rastgele orman modelinde %99.5 doğruluk oranı, F1 skoru 0.995, AUC değeri 1.000, kesinlik değeri 0.995, hassasiyet değeri 0.995, özgüllük değeri olarak rastgele orman 1.000 en başarılı algoritma olarak tespit edilmiştir. Buna karşın Naive bayes modeli en kısa sürede sınıflama işlemini yapan model olarak saptanmıştır. Kayıp oranı olarak en az skor 0.048 ile lojistik regresyon modelinde tespit edilmiştir. Tarımsal faaliyetlerle ilgilenenlerin veya araştırmacıların kullanabileceği python dili ve kütüphaneleri kullanılarak rastgele orman modeli ile basit bir ara yüze sahip ürün tahminleme uygulaması yazılmıştır.

Anahtar kelimeler: Algoritma, Makine öğrenmesi, Orange, Sınıflama

ABSTRACT

APPLICATIONS OF MACHINE LEARNING ALGORITHMS IN AGRICULTURE

ERİŞ, Samet

M. Sc. Thesis, Department of Animal Science

Supervisor: Prof. Dr. Abdullah YEŞİLOVA

July 2021, 81 pages

In this study, the model that produces the most accurate prediction from the widely used machine learning algorithms on an agricultural data set was determined. With the determined model, a prediction software has been developed that advises the user on the selection of agricultural products more efficient for the soil types or on alternative agricultural products that can be grown. An open source licensed data set containing 22 different agricultural products, including properties such as nitrogen, potassium, phosphorus, moisture, pH, average temperature, average precipitation in the soil, was used. Support vector machines, naive bayes classifier, logistic regression, decision trees, adaboost, stochastic gradient descent, multilayer perceptron, k-nearest neighbor and random forest algorithms were modeled using Orange software. Training and testing time, AUC value, accuracy score, F1 score, precision values, sensitivity values, loss rates and specificity values were calculated using Orange software of these nine different models. According to these results, random forest was determined as the best model. In the random forest model, 99.5% accuracy rate, F1 score 0.995, AUC value 1.000, precision value 0.995, sensitivity value 0.995, random forest 1.000 as specificity value were determined as the most successful algorithm. On the other hand, the naive bayes model was determined as the model that performs the classification process in the shortest time. The least score as the loss rate was determined in the logistic regression model with 0.048. A product prediction application with a simple interface was written with the random forest model using the python language and libraries that can be used by those interested in agricultural activities or researchers.

Keywords: Algorithm, Classification, Machine learning, Orange

ÖN SÖZ

Tez konusunun belirlenmesinde, planlanmasında ve yürütülmesinde her türlü katkı ve desteğini gördüğüm danışman hocam sayın Prof. Dr. Abdullah YEŞİLOVA 'ya teşekkürlerimi sunarım. Benim için her zaman en iyisini isteyen annem Güler ERİŞ ve babam Rıfat ERİŞ'e bugüne kadar verdikleri emeklerden dolayı ve yüksek lisans eğitimim boyunca desteklerini esirgemeyen eşim Kübra ERİŞ'e teşekkürlerimi sunarım. Oğlum Selim ERİŞ'e ileride ilham olması dileğiyle...

2021

Samet ERİŞ

İÇİNDEKİLER

	Sayfa
ÖZET.....	i
ABSTRACT.....	iii
ÖNSÖZ.....	v
İÇİNDEKİLER.....	vii
ÇİZELGELER LİSTESİ.....	ix
ŞEKİLLER LİSTESİ.....	xi
SİMGELER VE KISALTMALAR.....	xv
EKLER DİZİNİ.....	xvii
1. GİRİŞ.....	1
2. KAYNAK BİLDİRİŞLERİ.....	3
3. MATERYAL VE YÖNTEM.....	13
3.1. Materyal.....	13
3.2. Yöntem.....	13
3.2.1. Makine öğrenmesi algoritmalarının kullanımı.....	15
3.2.2. Kullanılan bilgisayar konfigürasyonu.....	16
3.2.3. Adaboost algoritması.....	16
3.2.4. Çok katmanlı algılayıcı algoritması.....	18
3.2.5. Destek vektör makineleri algoritması.....	20
3.2.6. Karar ağaçları algoritması.....	21
3.2.7. K-en yakın komşu algoritması.....	23
3.2.8. Lojistik regresyon algoritması.....	24
3.2.9. Naive Bayes sınıflandırıcı algoritması.....	25
3.2.10. Olasılıksal dereceli azalma algoritması.....	26
3.2.11. Rastgele orman algoritması.....	29
3.2.12. Veri seti ön hazırlık işlemleri.....	30
3.2.13. Hedef veri seçimi.....	32
3.2.14. Veri setinin bölünmesi.....	33
3.2.15. Çapraz doğruluk.....	34

	Sayfa
3.3. İncelenen Özellikler.....	34
3.4. Seçilen Modelin Programlanması.....	37
3.5. Modelin Ara Yüze Uyarlanması.....	37
4. BULGULAR.....	39
4.1. Tanıtıcı İstatistikler.....	39
4.2. Algoritmaların Test Sonuçları.....	48
4.2.1. Adaboost modeli ve sonuçları.....	48
4.2.2. Çok katmanlı algılayıcı modeli ve sonuçları.....	50
4.2.3. Destek vektör makineleri modeli ve sonuçları.....	52
4.2.4. Karar ağaçları modeli ve sonuçları.....	54
4.2.5. K-en yakın komşu modeli ve sonuçları.....	56
4.2.6. Lojistik regresyon modeli ve sonuçları.....	57
4.2.7. Naive bayes sınıflandırıcı modeli ve sonuçları.....	59
4.2.8. Olasılıksal dereceli azalma modeli ve sonuçları.....	61
4.2.9. Rastgele orman modeli ve sonuçları.....	64
4.3. Tüm Modellerin Toplu Test Sonuçları.....	65
4.4. Model Dosyası Oluşturulması ve Tahmin Ara Yüzü.....	67
5. TARTIŞMA ve SONUÇLAR.....	71
KAYNAKLAR.....	75
EKLER.....	79
ÖZGEÇMİŞ.....	81

ÇİZELGELER LİSTESİ

Çizelge	Sayfa
Çizelge 3.1. Çalışmanın yapıldığı bilgisayarın konfigürasyonu.....	16
Çizelge 4.1. Elma için tanıttıcı istatistikler.....	39
Çizelge 4.2. Fasulye için tanıttıcı istatistikler.....	39
Çizelge 4.3. Güve fasulyesi için tanıttıcı istatistikler.....	40
Çizelge 4.4. Güvercin bezelyesi için tanıttıcı istatistikler.....	40
Çizelge 4.5. Hindistan cevizi için tanıttıcı istatistikler.....	40
Çizelge 4.6. Jüt için tanıttıcı istatistikler.....	40
Çizelge 4.7. Kahve için tanıttıcı istatistikler.....	41
Çizelge 4.8. Karpuz için tanıttıcı istatistikler.....	41
Çizelge 4.9. Kavun için tanıttıcı istatistikler.....	41
Çizelge 4.10. Mango için tanıttıcı istatistikler.....	41
Çizelge 4.11. Maş fasulyesi için tanıttıcı istatistikler.....	42
Çizelge 4.12. Mercimek için tanıttıcı istatistikler.....	42
Çizelge 4.13. Mısır için tanıttıcı istatistikler.....	42
Çizelge 4.14. Muz için tanıttıcı istatistikler.....	42
Çizelge 4.15. Nar için tanıttıcı istatistikler.....	43
Çizelge 4.16. Nohut için tanıttıcı istatistikler.....	43
Çizelge 4.17. Pamuk için tanıttıcı istatistikler.....	43
Çizelge 4.18. Papaya için tanıttıcı istatistikler.....	43
Çizelge 4.19. Pirinç için tanıttıcı istatistikler.....	44
Çizelge 4.20. Portakal için tanıttıcı istatistikler.....	44
Çizelge 4.21. Siyah mercimek için tanıttıcı istatistikler.....	44
Çizelge	Sayfa

Çizelge 4.22. Üzüm için tanıtıcı istatistikler.....	44
Çizelge 4.23. Adaboost modelinin test ve skor sonuçları.....	49
Çizelge 4.24. Adaboost modeli hata matrisi.....	50
Çizelge 4.25. Çok katmanlı algılayıcı modelinin test ve skor sonuçları.....	51
Çizelge 4.26. Çok katmanlı algılayıcı modeli hata matrisi.....	51
Çizelge 4.27. Destek vektör makineleri modelinin test ve skor sonuçları.....	53
Çizelge 4.28. Destek vektör makineleri modelinin hata matrisi.....	53
Çizelge 4.29. Karar ağaçları modelinin test ve skor sonuçları.....	54
Çizelge 4.30. Karar ağaçları modeli hata matrisi.....	55
Çizelge 4.31. K-en yakın komşu modeli test ve skor verileri.....	56
Çizelge 4.32. K-en yakın komşu modeli hata matrisi.....	57
Çizelge 4.33. Lojistik Regresyon modeli test ve skor verileri.....	58
Çizelge 4.34. Lojistik regresyon modeli hata matrisi.....	59
Çizelge 4.35. Naive bayes sınıflandırıcı modeli test ve skor verileri.....	60
Çizelge 4.36. Naive bayes modeli hata matrisi.....	61
Çizelge 4.37. Olasılıksal dereceli azalma modeli test ve skor verileri.....	62
Çizelge 4.38. Olasılıksal dereceli azalma modeli hata matrisi.....	63
Çizelge 4.39. Rastgele orman modeli test ve skor verileri.....	64
Çizelge 4.40. Rastgele orman modeli hata matrisi.....	65
Çizelge 4.41. Kullanılan modellerin doğruluk sırasına göre test ve skor tablosu.....	66
Çizelge 4.42. Bütün modellerin doğruluk skoruna doğruluk oranı.....	66
Çizelge 4.43. Modellerin eğitim ve test sürelerinin toplamı.....	67

ŞEKİLLER LİSTESİ

Şekil	Sayfa
Şekil 2.1. İki bulanık model ve USLE modelinin araştırma yapılan havzada tahmin ettikleri toprak erozyonu miktarlarının karşılaştırılması (Yakupoglu ve ark., 2008).....	3
Şekil 2.2. Yabancı ot tespидinde görüntü işleme basamakları (Burgos-Artizzu ve ark., 2010).....	4
Şekil 2.3. Yapay sinir ağlarının mistleme sisteminde kullanılan yapısı (Sungur ve Altun, 2010).....	5
Şekil 2.4. Uzman sistemin çalışma yapısı (Robindro ve Sarma, 2013).....	6
Şekil 2.5. Limon yaprağı sınıflandırılma işlemi (Singh ve Misra, 2017).....	8
Şekil 2.6. Çekim Sistemi a) Kontrollü Mod b) Kontrolsüz Mod (Sabzi ve Abbaspour-Gilandeh, 2018).....	9
Şekil 2.7. Birinci ve ikinci tip segmentasyon (a): Orijinal resim, (b): İlk tür bölümlleme, (c): Orijinal görüntü üzerinde ikinci tür bölümlendirme, (d): Birinci tür segmentasyon resmi üzerinde ikinci tür bölümlleme resmi, (e) Gri tonlamalı resmin üzerinde ikinci bölümlleme türü ve (f): İkili görüntü üzerinde ikinci tür bölümlleme (Sabzi ve Abbaspour-Gilandeh, 2018).....	9
Şekil 2.8. İlaçlama yapacak araç, araca bağlı olan püskürtme başlığı kısmı ve sistemin hedef ve hedef olmayan ayırımı yaptığı görüntü (Partel ve ark., 2019).....	11
Şekil 2.9. Akıllı teknoloji kullanılarak geliştirilen yabancı ot haritası (Partel ve ark., 2019).....	11
Şekil 3.1. Orange yazılımının ara yüzü.....	17
Şekil 3.2. Adaboost algoritmasının parametreleri.....	17
Şekil 3.3. Çok katmanlı algılayıcı modeli.....	18
Şekil 3.4. Çok katmanlı algılayıcı aracının özellikleri.....	19
Şekil 3.5. Destek vektör makineleri aracının özellikleri.....	20
Şekil 3.6. Karar ağacı örneği.....	22

Şekil	Sayfa
Şekil 3.7. Karar ağaçları aracının özellikleri.....	22
Şekil 3.8. K-en yakın komşu aracının özellikleri.....	24
Şekil 3.9. Lojistik regresyon aracının özellikleri.....	25
Şekil 3.10. Naive Bayes sınıflandırma aracının özellikleri.....	26
Şekil 3.11. Olasılıksal dereceli azalma aracının özellikleri.....	28
Şekil 3.12. Rastgele orman aracının özellikleri.....	30
Şekil 3.13. Veri setinin ön hazırlık aşamaları.....	31
Şekil 3.14. Veri setinin Orange yazılımı tarafından okunması.....	31
Şekil 3.15. Eksik verilerin ne yapılacağının seçilmesi.....	32
Şekil 3.16. Orange ile hedef veri seçimi.....	33
Şekil 3.17. Veri setinin eğitim ve test verisi olarak ayarlanması.....	33
Şekil 3.18. Hata matrisi tablosu.....	34
Şekil 4.1. Ürünlerin yetiştirildiği azot değeri ortalamasına göre karşılaştırılması.....	45
Şekil 4.2. Ürünlerin yetiştirildiği potasyum değeri ortalamasına göre karşılaştırılması.....	45
Şekil 4.3. Ürünlerin yetiştirildiği fosfor değeri ortalamasına göre karşılaştırılması.....	46
Şekil 4.4. Ürünlerin yetiştirildiği nemlilik (%) değeri ortalamasına göre karşılaştırılması.....	46
Şekil 4.5. Ürünlerin yetiştirildiği ph değeri ortalamasına göre karşılaştırılması.....	47
Şekil 4.6. Ürünlerin yetiştirildiği sıcaklık (°C) değeri ortalamasına göre karşılaştırılması.....	47
Şekil 4.7. Ürünlerin yetiştirildiği yağış miktarı(mm) değeri ortalamasına göre karşılaştırılması.....	48
Şekil 4.8. Orange yazılımı ile tasarlanan adaboost modeli.....	49
Şekil 4.9. Orange yazılımı ile tasarlanan çok katmanlı algılayıcı modeli.....	50
Şekil 4.10. Orange yazılımı ile tasarlanan destek vektör makineleri modeli.....	52

Şekil	Sayfa
Şekil 4.11. Orange yazılımı ile tasarlanan karar ağaçları modeli.....	54
Şekil 4.12. Orange yazılımı ile yapılan k-en yakın komşu modeli.....	56
Şekil 4.13. Orange yazılımı ile yapılan k-en yakın komşu modeli.....	58
Şekil 4.14. Orange yazılımı ile tasarlanan naive bayes sınıflandırıcı modeli.....	60
Şekil 4.15. Orange yazılımı ile tasarlanan olasılıksal dereceli azalma modeli.....	62
Şekil 4.16. Orange yazılımı ile tasarlanan rastgele orman modeli.....	64
Şekil 4.17. Kullanılan makine öğrenmesi modellerinin test edilmesi.....	65
Şekil 4.18. Ürün tavsiye sistemi ara yüzü.....	69



SİMGELER VE KISALTMALAR

Bu çalışmada kullanılmış bazı simgeler ve kısaltmalar, açıklamaları ile birlikte aşağıda sunulmuştur.

Simgeler	Açıklama
----------	----------

%	Yüzde
°C	Santigrat derece
CO ₂	Karbon dioksit
cm	Santimetre
sn.	Saniye

Kısaltmalar	Açıklama
-------------	----------

AUC	Eğirinin altında kalan alan
CART	Sınıflandırma ve regresyon ağacı
ÇKA	Çok katmanlı algılayıcı
DVM	Destek vektör makineleri
GPL	Genel kamu lisansı
İDM	İki değişkenli model
JESS	Java uzman sistem kabuğu
LED	Işık veren diyot
OECD	Ekonomik İşbirliği ve Kalkınma Örgütü
RELU	Doğrultulmuş doğrusal birim
USLE	Evrensel toprak kaybı denklemi
ÜDM	Üç değişkenli model



EKLER DİZİNİ

	Sayfa
Ek 1. Kullanıcı ara yüzü tasarım kodları App.py dosyası.....	
....79	
Ek 2. Model dosyası oluşturulması için kullanılan model.py dosyası kodları.....	80





1. GİRİŞ

Makine öğrenmesi algoritmaları, insanların karmaşık veri setlerini keşfetmesine, çözümüleme yapmasına ve bunlardan mantıklı kurallar çıkarımları yapmasına olanak sağlayan kod parçacıklardır. Bu algoritmalar, bir uygulamanın belirli bir hedefi gerçekleştirmesi için izlediği sınırlı ve belirli adım adım ilerleyen yönergelerin bütünüdür. Makine öğrenmesi modellerinin ana hedefi, insanların tahmin yapmak veya verileri kategorilere ayırmak için kullanabileceği desenler oluşturmak veya keşfetmektir. Makine öğrenmesi algoritmaları, büyük kümeyi temsil eden eğitim verilerini temel alan değişkenleri kullanır. Eğitim verilerinin sayısı arttıkça dünyayı daha gerçekçi bir şekilde temsil etmesine olanak sağlar ve bu sayede algoritma daha doğru sonuçlar hesaplar. Farklı algoritmalar istenilen hedefi gerçekleştirmek için verileri farklı şekilde işleyerek analiz etmektedir. Makine öğrenmesi algoritmaları tıpkı diğer bilgisayar algoritmaları gibi yapılacak bir işin veya çözülecek bir problemi belli bir sıra ve mantık çerçevesinde değerlendirip istenilen sonucu üretmek için araçlardır. Bazı algoritmalar problem çözümünde hız konusunda fayda sağlarken bazıları da problemlerin çözümünde doğruluk skorunu arttırmada ve kayıp oranı azaltmada fayda sağlamaktadır.

Yaygın olarak kullanılan bazı makine öğrenme algoritmaları bulunmaktadır. Naive bayes algoritmaları, ilgili bir olayın gerçekleşmesini temel alarak bir olayın gerçekleşme olasılığını hesaplamaktadır. Destek vektör makineleri birbirine en yakın olan veri noktalarını kullanarak aralarına bir hiper düzlem çizer. Bu yapılan işlem, sınıfları ayırır ve bunların daha net şekilde birbirinden ayırt edilmesini sağlamak için aralarındaki mesafeyi maksimuma çıkarır. Karar ağacı algoritmaları verileri iki veya daha fazla homojen kümeye ayırmak için kullanılmaktadır. Verileri, veri noktaları arasındaki en önemli fark yaratan öğeyi temel alarak bu ayırma işlemini gerçekleştirir. K-En yakın komşu algoritmaları da kullanılabilen bütün veri noktalarını kayıt ederek bir mesafe fonksiyonu ile ölçüldüğü gibi her yeni veri noktasını buna en yakın konumlandırılan veri noktalarını temel alarak sınıflandırma yapmaktadır. Dereceli artırma algoritmaları ise modelin performansını arttıran birleştirme işlemi ile zayıf tahmin modellerini (çoğunlukla karar ağaçlarını) bir araya getirilmesi sonucunda bir

tahmin modeli oluşturur. K-ortalamlar algoritmaları, verileri kümeleme yaparak sınıflandırma işlemi yapar ve K değeri kümelerin sayısı ile aynı olmaktadır. Her kümenin içinde bulunan veri noktaları homojen olarak bulunmaktadır. Bunlar, kendileri haricindeki diğer kümelerde bulunan veri noktalarına karşı heterojen özellik göstermektedir (Anonim, 2021).

Çalışmada makine öğrenmesi algoritmalarını kullanarak tarımsal ürünlerin toprak çeşitlerine göre daha verimli ürün seçimi veya alternatif yetiştirilebilecek tarım ürünleri konusunda kullanıcıya tavsiye veren bir tahminleme uygulaması yapılmıştır. Bir tarım veri setinde tahminleme işlemini makine öğrenmesi algoritmalarından adaboost, destek vektör makineleri, karar ağaçları, olasılıksal dereceli azalma, çok katmanlı algılayıcı, naive bayes sınıflandırıcı, k-en yakın komşu ve rastgele orman algoritmaları kullanılarak sekiz farklı model test Orange yazılımı ile test edilmiştir. Yiğit ve ark. (2019) yaptıkları çalışmada sağlıklı yaprak görselleri ile kusurlu yaprak görsellerini bu algoritmalarından yapay sinir ağı, naive bayes algoritması, rastgele orman algoritması, en yakın komşu ve destek vektör makinesi kullanarak %92,53 kesinlikle tespit yapmıştır. Tarımsal veriler üzerinde yapılan bu çalışma ile bu veri seti için en doğru tahminler sunan makine öğrenmesi modeli için Python dili ve kütüphaneleri ile basit düzeyde çiftçilerin kullanabileceği tahminleme ara yüzü tasarlanarak toprak analizi değerlerine göre en verimli yetişecek ürün tespitini yapmasına olanak sağlamaktadır. Robindro ve Sarma (2013) yaptıkları çalışmada java uzman sistem kabuğu ile tasarlanan uzman sistemle bitkide gözlenen hastalık belirtilerini kullanarak, hastalıkların tespit edilmesi ve hastalıkların şiddetine göre alınacak tedbirler konusunda öneriler vermektedir. Üreticinin konu hakkında uzman yönlendirmesine ihtiyaç duyduğunda, geliştirilen bu uzman sistemin çiftçiye karar desteği sağlayacağını belirtilmiştir.

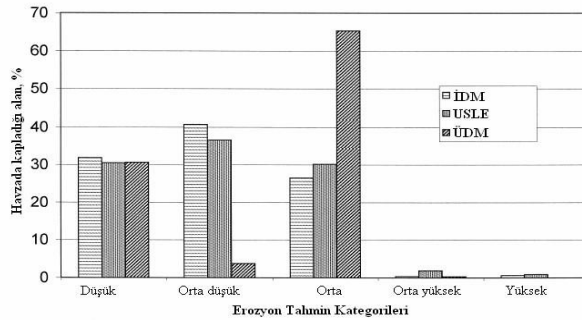
Bu çalışmada makine öğrenmesi algoritmalarının kullanılan veri setlerinde uygulanabilirlikleri kontrol edilmiştir. Eğer kullanılacak olan algoritmanın yeterliliklerini ve eksikliklerini hata matrisleri ile doğru analiz edilirse daha sonra yapılacak olan çalışmalarda hangi algoritmanın hangi konuda daha etkili çalıştığını anlamlandırarak sonuca ulaşmak daha kolay olacaktır. Algoritma seçimi açık kaynak kodlu Orange yazılımı ile yapılmıştır. Çalışmada kullanılacak veri setlerinin ön işleme tabi tutulup analizleri yapılacaktır. Çalışma sonucunda doğruluk oranı ne kadar yüksek olursa tahminleme modeli o kadar başarılı olacağı tespit edilmiştir. Ülkemizde ve

dünyada ziraat alanında birçok yapay zekâ çalışmaları yapılmaktadır. Yapılan bu tahminleme sisteminin tarım alanındaki bu çalışmalara katkı sağlayacağı düşünülmektedir.



2. KAYNAK BİLDİRİŞLERİ

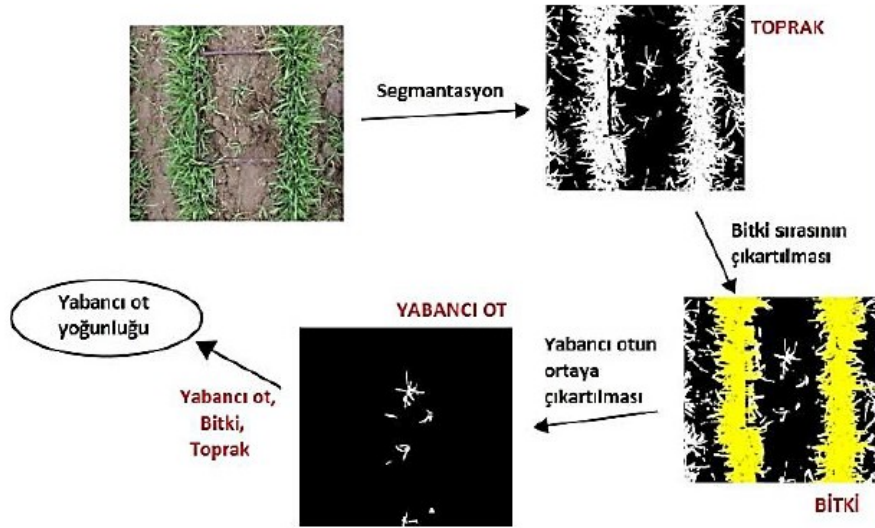
Toprak erozyonunu tahminlemede makine öğrenmesi modellerinden bulanık mantık temelli modellerin kullanımı incelenmiştir. İki değerli mantık oluşturulmuş ve iki değerli klasik mantıktan bulanık mantığa evrilme süreçleri hakkında tespitler yapılmıştır. Yapılan çalışmada geniş bir havzadaki toprak erozyonunu tahminlemek için bulanık kural tabanlı sistemi konu alan iki farklı model iki değişkenli model (İDM) ve üç değişkenli model (ÜDM) oluşturulması konusunda çalışılmıştır. Oluşturulan modellerden elde edilen veriler USLE (evrensel toprak kayıp denklemi)'den elde edilen sonuçlarla kıyaslanarak yorumlamalar yapılmıştır. Sonuç olarak bulunan İDM ve ÜDM verileri ile USLE denklemi Şekil 2.1'de belirtilen grafikteki gibi beş kategoride düşük, orta düşük, orta, orta yüksek ve yüksek olarak sınıflandırılmıştır. Toprak erozyonunu tahmin etmek için bulanık mantık ile oluşturulan modellerin klasik mantık ile oluşturulan modellerden daha başarılı olduğunu ve daha az parametreye ihtiyaç duyduğu belirtilmiştir (Yakupoğlu ve ark., 2008).



Şekil 2.1. İki bulanık model ve USLE modelinin araştırma yapılan havzada tahmin ettikleri toprak erozyonu miktarlarının karşılaştırılması (Yakupoğlu ve ark., 2008).

Yabancı ot tespiti için bir görüntüdeki yabancı ot, bitki ve toprak yüzdelерinin tahmini için görüntülemeye dayalı birkaç gelişmiş yöntem üzerine bir çalışma yapılmıştır. Farklı görüntü verisi setlerinde kullanmak üzere en iyi parametre değerini ve yöntemi birlikte bulmak için genetik bir algoritma kullanmışlardır. Tavsiye edilen yöntemler hem yabancı otların doğru tespit edilmesinde hem de düşük hesaplama

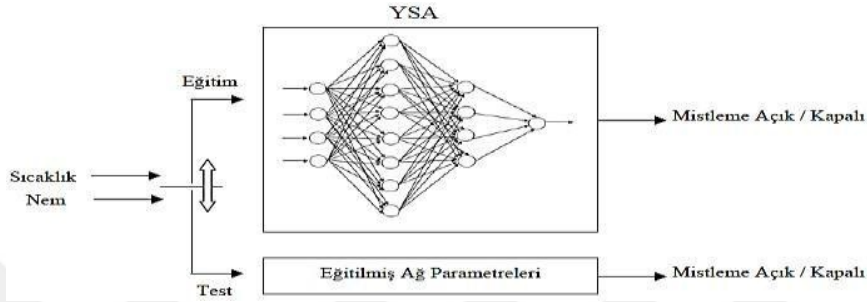
karmaşıklığında, çok iyi sonuçlar ürettiği belirtilmiştir. Çalışma görüntü işleme kullanılarak her bir farklı tarımsal elementin çıkarıldığı üç basamağa ayrılarak yapılmıştır (Şekil 2.2). Bu aşamalar birinci basamakta bitki örtüsü olan ve bitki örtüsü olmayan bölümlerin yani toprak kısımlarının çıkarılması, daha sonra bitki sırasının görüntüden çıkartılması ve yabancı otun ortaya çıkartılması şeklindedir. Çalışmada bütün aşmalarda, her bir analizi hassaslaştırmak için değerin değiştirilebileceği birkaç girdi değişkeni kullanan farklı ve değiştirilebilir metotlar önermektedirler. Çalışmanın gerçek zamanlı görüntü işleme sisteminin oluşturulmasında bir basamak olacağı belirtilmiştir (Burgos-Artizzu ve ark., 2010).



Şekil 2.2. Yabancı ot tespiti için görüntü işleme basamakları (Burgos-Artizzu ve ark., 2010).

Konya havzasında don olaylarına karşı yapay sinir ağları kullanılarak mistleme sistemi geliştirilmiştir. Çalışmada meyve ağaçlarını soğuktan ve ani sıcaklık değişimlerinden en az zararla etkilenmesi amacıyla otomatik olarak sıcaklık ve nemi ayarlayan bir yapay zekâ sistemi tasarlanmıştır. Tasarlanan sistem sensörler aracılığıyla otomatik olarak devreye girmesi için ortam sıcaklığı ve bağıl nem değerleri kaydedilerek bu değerler daha sonra yapay sinir ağları ile eğitilip testleri gerçekleştirilmiştir. Çalışmada 2008 yılında elde edilen 8724 adet sıcaklık ve nem değeri verileri ile yapay zekâ eğitilmiştir. Gerçek değerlerin performanslarının ölçülmesi için 2009 yılında toplanılan 8760 adet nem ve sıcaklık verisi toplanarak kullanılmıştır. Yapay sinir ağları ile tasarlanan bu sistem Şekil 2.3'de sıcaklık ve nem değerlerinin

kurulan otomasyon sistemini devreye alıp almaması gerektiğine dair çıkış değerlerini göstermektedir Çalışmada üretilen yapay zeka mistleme sistemi otomatik olarak beklenen değerlerde çalıştığı gözlemlenmiştir. Böylelikle yapay zeka sinir ağlarının tarımda don olaylarını engellemede kullanılabileceği tespit edilmiştir (Sungur ve Altun, 2010).



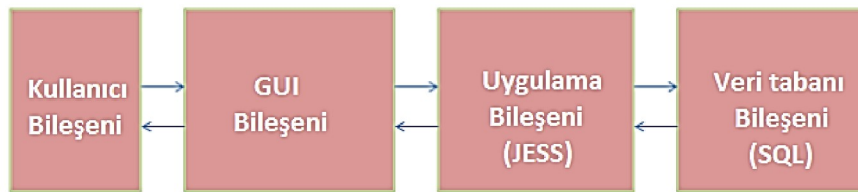
Şekil 2.3. Yapay sinir ağlarının mistleme sisteminde kullanılan yapısı (Sungur ve Altun, 2010).

Ataş (2011), çalışmasında aflatoksin ile kirlenmiş biberleri kirlenmemiş olanlardan hızlı ve zararsız bir şekilde ayırabilecek bir bilgisayarla görü sistemi geliştirmiştir. Türkiye'nin farklı bölgelerinden toplanan acı biber örneklerinin hiperspektral görüntü serileri halojen ve ultraviyole aydınlatmaları altında elde edilmiştir. Öznitelik seçimi için fisher ayırım gücü, teorik bilgiler minimum yedeklilik maksimum uygunluk yöntemi ve önerilen çok katmanlı algılayıcı tabanlı öznitelik seçim şemaları kullanılmıştır. Son olarak lineer diskriminant sınıflandırıcı, destek vektör makineleri ve çok katmanlı algılayıcı sınıflandırıcılar olarak kullanılmıştır. Bu yöntemleri kullanarak doğrusal ayrımsallık sınıflandırıcısı, destek vektör makineleri ve yapay sinir ağları modellerini sınıflandırıcı olarak kullanılmıştır. Yüksek sınıflandırma doğruluğu ve öngörücü sağlamlığı elde etmek için, halojen uyarılı ve ardışık spektral bant özelliklerinin nicelleştirilmiş mutlak farkına sahip bir yapay görme sisteminin kullanılması önerilmiştir.

Arazi toplulaştırma çalışmalarında bulanık mantık kullanılarak dağıtım aşamasında sabit tesisin açısı, en büyük parselin semti, sabit tesis genişliği ve en büyük parsel genişliği olarak bir bulanık mantık sistemi üretilmiştir. Tarım Reformu Genel Müdürlüğünden elde edilen mülakat esaslı dağıtım sonucuyla karşılaştırılmış ve sonuçlar incelendiğinde yeni dağıtım için harcanan iş gücü, ortalama parsel genişliği,

parsel adeti, dağıtımda harcanılan zamanda yapay zekâ modeli daha iyi sonuçlar verdiği belirtilmiştir. Arazi toplulaştırma sonucunda parsel adeti %40 olarak azalmıştır. Mülakat esaslı dağıtım modelinde %35,9 azalma göstermiştir. Bulanık mantık yapay zekâ modelinde ise %35,9 azalma olduğu tespit edilmiştir. Ortalama parsel büyüklükleri tekrar kıyas edildiğinde mülakat esaslı dağıtımda ortalama parsel genişliği %13,4 artmış olup, bulanık mantık yapay zekâ modelinde %24 arttığı ortaya konulmuştur (Kara, 2012).

Bitkilerdeki hastalıklarının tespit edilmesi için Java dilinde yazılmış uzman sistem mimarisinin tasarım ve prototipi üretilmiştir. Uzman sistem, veri tabanı, çıkarım motoru ve kullanıcı ara yüzünden oluşan bir bilgisayar programıdır. Çalışmada tasarlanan uzman sistem, bitkide gözlenen hastalık belirtilerini kullanarak hastalıkların nasıl tespit edileceği ve hastalıkların şiddetine göre alınacak tedbirler konusunda öneriler sunmaktadır. Çalışma ayrıca pirinç bitkisinde hastalıkların tespiti için bir uzman sistem kullanılmasının önemini ortaya koymaktadır. Ayrıca böyle bir uzman sistem çiftçiler için çok önemlidir. Çünkü uzmanlar her zaman çiftçilerin yanında bulunması mümkün değildir ve hastalıklar verimsizliğin önemli faktörlerinden biridir. Tarımsal faaliyetlerde bu alanda çalışan insanların bir uzman tavsiyesine ihtiyaç duyduğunda, geliştirilen bu uzman sistemin çiftçiye karar desteği sağlayacağını vurgulamışlardır. Şekil 2.4’de yapılan çalışmanın çalışma sistemi gösterilmiştir (Robindro ve Sarma, 2013).



Şekil 2.4. Uzman sistemin çalışma yapısı (Robindro ve Sarma, 2013).

Arpa ve mısır tarımı yapılan tarlalarda kullanılan yabancı ot tespiti yapabilen otomatik bir uzman sistem tasarlanmıştır. Sistem birinci basamakta görüntü işleme ile histogram oluşturarak analiz yaparak karar vermektedir. İkinci basamakta ise iki farklı yöntem önerilerek yeşillik tanımlaması yapılmaktadır. Klasik yeşillik tanımlaması kullanılan karar verme ve bulanık mantık kullanılan yöntem aracılığı ile karar verme

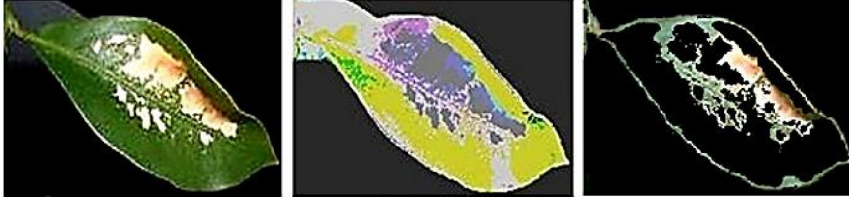
mekanizması olarak iki farklı yaklaşım kurulmuştur. Tasarlanan uzman sistem sayesinde yeşil bitkilerin tanımlama işleminden sonra toprak ve geri kalan bölümlerin ilgili kategorilerde tanımlamaları yapılması için görüntü işleme ile analizlerin yapılabileceği vurgulanmıştır (Romeo ve ark., 2013).

CO₂ emisyonunun takip edilmesi gelecek yılların CO₂ emisyonlarının planlanması çevreye vereceği zararın özellikle iklim değişikliğinin önüne geçilmesi için önemli bir görevdir. Yapılan bu çalışmada Ekonomik İşbirliği ve Kalkınma Örgütü ülkelerinden yüksek CO₂ salınımına sahip ülkelerden Almanya, Amerika Birleşik Devletleri, Avustralya, Japonya, Kanada, Güney Kore, İtalya, Fransa, Türkiye ve Meksika için makine öğrenmesi yöntemleri ile CO₂ emisyonları için tahminler gerçekleştirilmiştir. Çalışmada makine öğrenmesi algoritmalarından M5P, destek vektör makineleri ve çok katmanlı algılayıcı modelleri ile ülkelerin CO₂ emisyonları tahmin edilmeye çalışılmıştır. Kullanılan makine öğrenmesi algoritmaları birbiri ile kıyas edildiğinde en başarılı algoritma destek vektör makineleri algoritmasının daha başarılı sonuçlar verdiği tespit edilmiştir (Garip, 2017).

Gıda güvenliğini temin etmek amacıyla gıdalar üzerinde bulunan patojenik mikroorganizmalar veya kanserojen toksinler içeren gıdaların belirli ışıklar altında görüntü işleme metotlarına tabi tutulup içerisindeki zararlıların tespiti sağlanmıştır. Çalışmada incirin hiperspektral görüntüleme teknolojisi kullanılarak üzerindeki aflatoksin bileşeni ve yüzey küf bileşeni arasındaki ilişki tespit edilmesine yönelik özgün yöntemler geliştirilmiştir. Gıda güvenliğinin sağlanması aynı zamanda ürünlere zarar vermeden analizlerini gerçekleştirilecek sistematik görüntü işleme metotları ile güvenli hale gelmektedir. Aflatoksin ve yüzey küf yaklaşımı için görüntü sınıflandırmaya yönelik yeni algoritmalar tasarlanarak eğitim kümesi için örnek sayısı azaltılarak etiketleme maliyetleri düşürülmüştür (Güneş, 2017).

Bitki hastalıklarını erken teşhis amacıyla yapılan çalışmada limon ağacı yapraklarını renklerine göre sınıflandırarak bitki hastalıklarını otomatik olarak tespit edilmesi için genetik algoritmalar kullanmıştır. Başarı oranını arttırmaya yönelik makine öğrenmesi algoritmalarından yapay sinir ağları, hibrit algoritmalar, bulanık mantık ve naive bayes sınıflayıcı algoritmalarını kullanarak hastalıklı yaprakların sınıflandırma işlemi yapılmıştır. Şekil 2.5'te görüntü işleme yöntemi ile limon

yaprağının renklerine göre sınıflandırma işlemi gösterilmektedir (Singh ve Misra, 2017).



Şekil 2.5. Limon yaprağı sınıflandırılma işlemi (Singh ve Misra, 2017).

Bilgili (2018), çalışmasında gemilerin kullandığı fosil yakıtların yanma sürecinde oluşan gaz emisyonunun çevreye ve insan sağlığına verdiği zararlarından bahsetmiştir. Gemi emisyonlarının azaltılmasına, kontrol altına alınmasına ve tahminlemesine yönelik, dokuz adet kuru yük gemisinin günlük bazda alınan raporlar ile elde edilen verilerden emisyon miktarları hesaplanıp regresyon analizi yapılmıştır. Daha sonra bulunan sonuçlar makine öğrenmesi algoritmalarından yapay sinir ağları ile modellenmiş ve regresyon analizinden daha başarılı sonuçlar elde edilmiştir. Atlantik ve Pasifik Okyanus’larında iki farklı rotada altı aylık uygulama sonucunda iki okyanusun ve zaman dilimi için farklı deniz ve hava koşulları gemi rotalarındaki emisyon miktarları tahmin edilmiştir.

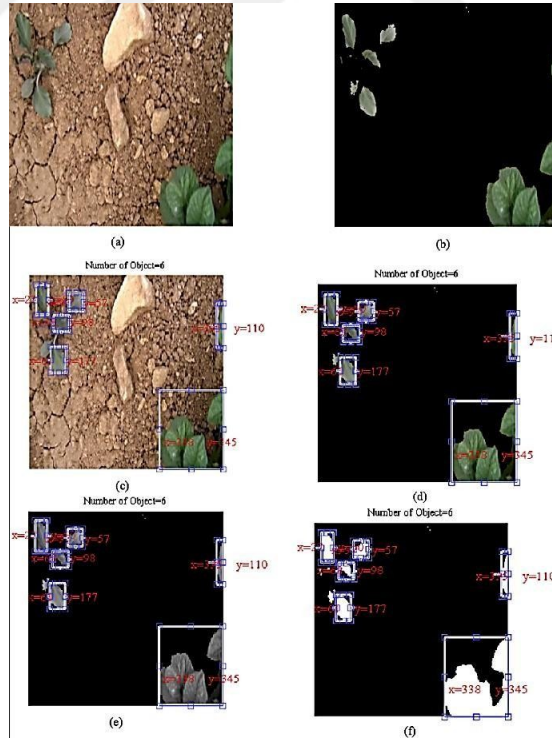
Makine öğrenmesinde yaygın kullanılan yapay sinir ağları, k-en yakın komşu, k-means, naive bayes, ve karar ağacı sınıflandırma yöntemlerinin, ekolojik verilerde ne şekilde kullanılabileceğinin anlaşılması amacıyla iki adet örnek veri kümesi, bu yöntemler kullanılarak değerlendirilmiştir. Bu yöntemleri sınamak için oluşturulan modeller, geliştirilen test ortamında farklı parametrik değerler ve “one hot encoding” doğrulama yöntemi ile test edildi. Modellenen bütün makine öğrenmesi yöntemleri %90 üzerinde bir başarı göstermiştir. (Balta, 2018).

Patates yetiştirirken sık rastlanan 3 farklı yabancı otun tespiti için yapay zeka destekli görüntü işleme sistemi yapılmıştır. Kullanılan sistem, yabancı otları ve patates bitkisini sınıflandıracak gerçek zamanlı video işleme alt sistemi ile birbirinden ayırmaktadır. Görüntüleme işlemi, beyaz LED lambalar kullanılarak kontrollü aydınlatma koşullarında yapılmıştır. Çalışmada çekimler tamamlandıktan sonra sınıflandırma işleminde 30 renk doku ve şekil özellikleri çıkartılmış ve karar ağaçları algoritmaları kullanılarak en önemli altı özellik tespit edilmiştir. Bitkilerin yabancı ot

veya patates olduğunun tespiti için yapay sinir ağları ve bayes sınıflandırıcı kullanılmış olup bu iki model birbiri ile karşılaştırılmıştır. Karşılaştırma sonucunda yapay sinir ağları %99.0 test grubunda %98.1 kesinlik sonucu vermiştir. Bayes sınıflandırıcı ise %71.7 test grubunda %73.3 kesinlik sonucu vermiştir. Yapay sinir ağları daha başarılı sonuçlar verdiği rapor edilmiştir. Şekil 2.7’de kurulan görüntü alma sistemi 2 ayrı mod olarak gösterilmiştir. Şekil 2.8’de ise iki tip sınıflanmadan sonucu elde edilen görüntülerin çıktısı verilmiştir (Sabzi ve Abbaspour-Gilandeh, 2018).



Şekil 2.6. Çekim Sistemi a) Kontrollü Mod b) Kontrolsüz Mod (Sabzi ve Abbaspour-Gilandeh, 2018).



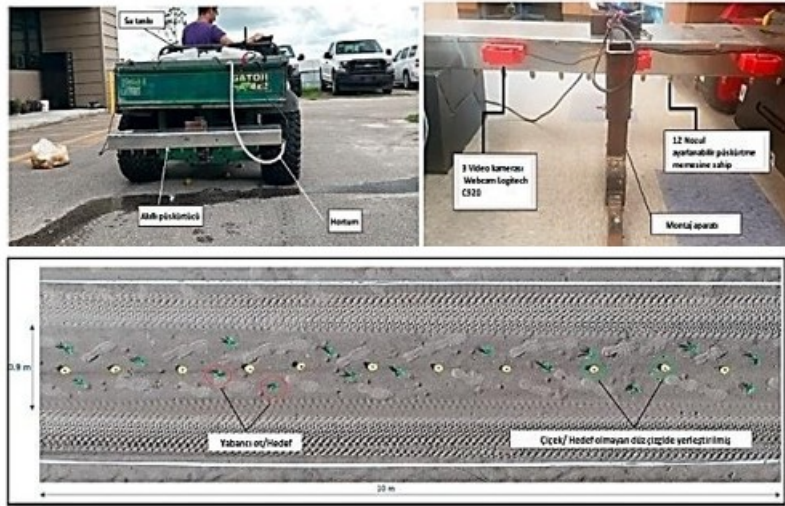
Şekil 2.7. Birinci ve ikinci tip segmentasyon (a): Orijinal resim, (b): İlk tür bölümleme, (c): Orijinal görüntü üzerinde ikinci tür bölümlendirme, (d): Birinci tür segmentasyon resmi üzerinde ikinci tür bölümleme resmi, (e) Gri tonlamalı

resmin üzerinde ikinci bölümlene türü ve (f): İkili görüntü üzerinde ikinci tür bölümlene (Sabzi ve Abbaspour-Gilandeh, 2018).

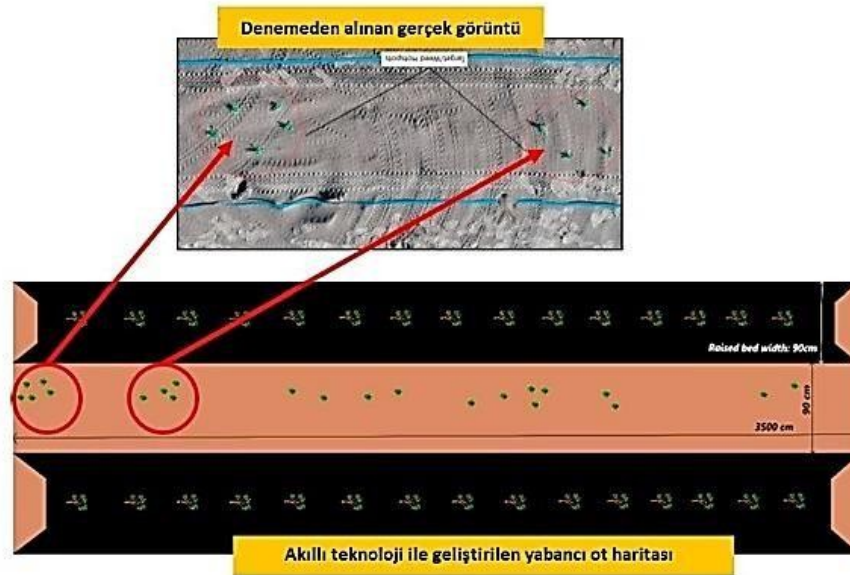
Öztürk (2018) çalışmasında, insansız hava araçları ile otomatik olarak görüntü işleme yaparak yaprakların şekillerini baz alarak farklı bitkilerin sınıflandırması sayesinde tarımda kimyasal ve biyolojik iyileşmeye katkı sağlayacak bir konu üzerinde çalışmıştır. Ayrıca, insansız hava aracı kullanılarak yapılan görüntüleme ile bitki hastalıkları ve böceklenme gibi vakaların tespiti gerçekleştirilerek önlem alınmasını kolaylaştırmaktadır. Çalışmada sınıflandırma işlemi watershed algoritması ile yapılmıştır. Görüntü işleme ve özellik çıkarımı bu özelliklerin öğretilmesi konusunda destek vektör makineleri algoritmasını kullanmıştır.

Yiğit ve ark. (2019), yapay sinir ağı, naive bayes algoritması, rastgele orman algoritması, en yakın komşu ve destek vektör makinesi gibi yapay zekâ tekniklerini kullanarak bitki yapraklarının görsel olarak otomatik tanımlanması üzerine bir çalışma yapılmıştır. Bitki yapraklarının otomatik olarak tanınması için yapay sinir ağı, rastgele orman algoritması, en yakın komşu, naive bayes sınıflayıcı ve destek vektör makineleri kullanılmıştır. Çalışmada 32 farklı bitki kullanılmıştır ve görüntü işleme ile renk, doku, desen ve boyut olarak özellik seçimi yapılmış dört farklı gruplandırma yapılmıştır. Sonuç olarak %92.91 kesinliğe sahip destek vektör makinaları en başarılı algoritma olarak tespit edilmiştir. Daha sonra araştırmacılar hastalıklı ve kusurlu yaprakların tespiti için destek vektör makinaları algoritmasını kullanmış ve test sonucunda %92.53 kesinlik sonucu ile en etkili faktörün doku olduğu bildirilmiştir.

Yabancı otları tespit edip istenilen hedefe püskürtme yapılmasını sağlayan yapay zekâ destekli akıllı bir pülverizatör tasarlanıp geliştirilmiştir. Pülverizatör, yabancı otları derin öğrenme metotları ile tespit edip püskürtme amacıyla üzerinde bulunan 12 ayrı kısa tepki püskürtme başlığına sahip bir yapay zekâ yazılımına sahiptir. Akıllı pülverizatör, yabancı ot (hedef) ile biber bitkilerini (hedef olmayan) ayırımını yaparak yalnızca yabancı ota ilaç püskürtme işlemini gerçekleştirmektedir. Şekil 2.8’de ilaçlama yapacak araç, araca bağlı olan püskürtme başlığı kısmı ve sistemin hedef ve hedef olmayan ayırımı yaptığı görüntü belirtilmiştir. Şekil 2.9’da ise bu sınıflandırma yapıldıktan sonra yapay zekanın oluşturduğu yabancı ot haritasına gösterilmektedir (Partel ve ark., 2019).



Şekil 2.8. İlaçlama yapacak araç, araca bağlı olan püskürtme başlığı kısmı ve sistemin hedef ve hedef olmayan ayırımı yaptığı görüntü (Partel ve ark., 2019).



Şekil 2.9. Akıllı teknoloji kullanılarak geliştirilen yabancı ot haritası (Partel ve ark., 2019).

3. MATERYAL VE YÖNTEM

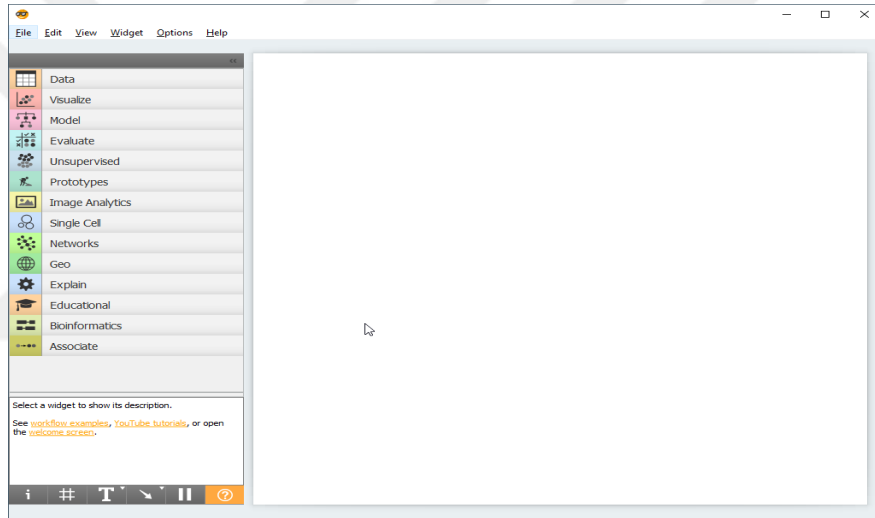
3.1. Materyal

Çalışmada kullanılan veri setinde elma, fasulye, güve fasulyesi, güvercin bezelyesi, hindistan cevizi, jüt, kahve, karpuz, kavun, mango, maş fasulyesi, mercimek, mısır, muz, nar, nohut, pamuk, papaya, pirinç, portakal, siyah mercimek ve üzüm olarak 22 farklı ürünün yetiştirilme koşulları bulunmaktadır. Her bitkinin yetiştirme şartını sağlayan yüzer adet topraktaki azot (%), fosfor (%), potasyum (%), ortalama sıcaklık(°C), nemlilik (%), pH ve yağış miktarı(mm) bilgilerini içeren veri seti kullanılmıştır. Kullanılan veri setinde eksik gözlem bulunmamaktadır. Veri seti Vishwakarma Teknoloji Enstitüsü’de eğitim alan Atharva Ingle tarafından açık kaynak lisanslı olarak kullanımına izin verilmiştir (Anonim, 2020b).

3.2. Yöntem

Çalışmada en iyi makine öğrenmesi modelinin seçimi için Lübllyana Üniversitesi (Slovenya) ve Jožef Stefan Enstitüsü tarafından geliştirilen Orange isimli açık kaynak kodlu yazılım kullanılmıştır. Orange, veri görselleştirme, makine öğrenimi, veri madenciliği ve veri analizi için bileşen tabanlı bir görsel programlama yazılım paketidir. Bileşenler araç (widget) olarak adlandırılır ve basit veri görselleştirme, alt küme seçimi ve ön işlemeden öğrenme algoritmalarının ampirik değerlendirmesine ve öngörücü modellemeye kadar çeşitlilik gösterir. Görsel programlama, iş akışlarının önceden tanımlanmış veya kullanıcı tarafından tasarlanmış araçlar bağlanarak oluşturulduğu bir ara yüz aracılığıyla gerçekleştirilirken, ileri düzey kullanıcılar Orange’ı veri işleme ve araç ögesi değişikliği için bir Python kitaplığı olarak da kullanabilir. Orange, GPL (Genel Kamu Lisansı) altında yayınlanan açık kaynaklı bir yazılım paketidir. Varsayılan kurulum, 6 araç setinde (veri, görselleştirme, sınıflandırma, regresyon, değerlendirme ve denetimsiz) bir dizi makine öğrenimi, ön işleme ve veri görselleştirme algoritmalarını içerir. Ek işlevler eklenti olarak mevcuttur (biyoinformatik, veri füzyonu ve metin madenciliği vb.). Orange, kullanıcının üzerine

araç yerleştirdiği ve bir veri analizi iş akışı oluşturduğu bir tuval ara yüzünden oluşur. Araçlar, verileri okuma, bir veri tablosu gösterme, özellikleri seçme, tahmin edicileri eğitme, öğrenme algoritmalarını karşılaştırma, veri öğelerini görselleştirme vb. gibi temel işlevleri sunar. Kullanıcı, görselleştirmeleri etkileşimli olarak keşfedebilir veya seçilen alt kümeyi diğer araçlarla besleyebilir. Program, deney seçimi, öneri sistemleri ve tahmine dayalı modelleme için bir platform sağlar ve biyotıp, biyoenformatik, genomik araştırma ve öğretimde kullanılır. Bilimde, yeni makine öğrenimi algoritmalarını test etmek ve genetik ve biyoinformatikte yeni teknikleri uygulamak için bir platform olarak kullanılır. Eğitimde biyoloji, biyotıp ve bilişim öğrencilerine makine öğrenimi ve veri madenciliği yöntemlerini öğretmek için kullanılmıştır.



Şekil 3.10. Orange yazılımının ara yüzü.

Çalışmada kullanılan veri seti üzerinde Orange yazılımı vasıtasıyla makine öğrenmesi algoritmalarından;

- Destek vektör makineleri (support vector classifier),
- Naive bayes sınıflandırıcı (naive bayes classifier),
- Karar ağaçları (decision tree),
- Lojistik regresyon (logistic regression)
- Adaboost (adaptif boosting),
- Olasılıksal dereceli azalma (stochastic gradient descent),
- Çok katmanlı algılayıcı (multi-layer perceptron classifier),

- K-en yakın komşu (k-nearest neighbors),
- Rastgele orman (random forest)

olmak üzere dokuz farklı makine öğrenmesi algoritması modellenmiştir. Bu yöntemler bir fonksiyon vasıtasıyla bir kısmı eğitim verisi bir kısmı da test verisi olacak şekilde veri tabanı ikiye ayrılıp algoritmadan sınıflandırma tahminleri yapılmaları istenmiştir.

3.2.1. Makine öğrenmesi algoritmalarının kullanımı

Teknolojinin gelişmesi ile beraber birçok alanda çevrimiçi ve çevrimdışı sistemlerde bilgiler toplanmaktadır. Toplanan bu bilgiler o kadar fazlaşmıştır ki veri yığınları haline gelmişlerdir. Bu veri yığınlarından anlamlı bilgiler çıkartma işine veri madenciliği denilmektedir. Veri madenciliği birçok disiplinin beraber kullanılması sonucu anlamlı bilgiler çıkartmayı sağlamaktadır. İstatistik, makine öğrenmesi veri tabanı teknolojileri ve veri görselleştirme bunlardan bazılarıdır. Makine öğrenmesi yapılan uygulamada yönergeler doğrudan verilmez, matematiksel modeller ile yapay zekanın öğrenmesi sağlanır. Makine öğrenmesi yapay zekanın bir alt kümesi olarak kabul edilmektedir. Algoritmaların kolay uygulanabilir olması sebebiyle farklı senaryolarda kolaylıkla entegre olarak çalışabilmektedirler. Makine öğrenmesinde genellikle denetimli öğrenme, denetimsiz öğrenme ve pekiştirmeli öğrenme olarak üç farklı yöntem kullanılmaktadır. Çalışmada veri etiketlerini önceden algoritmaya öğreterek sınıflandırma yaptırdığımız denetimli öğrenme yöntemi kullanılmıştır. Makine öğrenmesi algoritmalarının genel olarak çalışması ve kullanım mekanizması dört basamakta çalışmaktadır.

1. Veri toplama ve hazırlanma: Veri kaynaklarından elde edilen veriler düzenli bir depolama biçiminde arşivlenir. Veriler incelenir ve anormallikler belirlenir. Hangi makine öğrenmesi algoritmasının uygulanabilir olduğuna karar verilmektedir.

2. Modelin Eğitimi: Modelin en yüksek doğruluğa ulaşması için hazırlanan veriler eğitim ve test verisi olarak bölünür. Kurulan modelde parametreler değiştirilerek doğruluk oranı artırılır.

3. Modelin Doğrulanması: Nihai olarak kullanılacak makine öğrenmesi algoritmalarının performansı ve doğruluğu test edildiği aşamadır.

4. Sonuçların Yorumlanması: Sonuçları tahmin etmek için sonuç çıkarımlarının yapıldığı aşamadır.

3.2.2. Kullanılan bilgisayar konfigürasyonu

Çalışmada kullanılan bilgisayarın konfigürasyonu eğitim ve test sürelerinde farklılık göstermektedir. Daha güçlü işlemcilerde süreler kısalmaktadır.

Çizelge 3.1. Çalışmanın yapıldığı bilgisayarın konfigürasyonu

Bileşen	Konfigürasyon
İşlemci	AMD Ryzen 5 3600
RAM	16GB DDR4 3200 MHZ
İşletim Sistemi	Windows 10 64 Bit
Orange Versiyonu	3.28.0

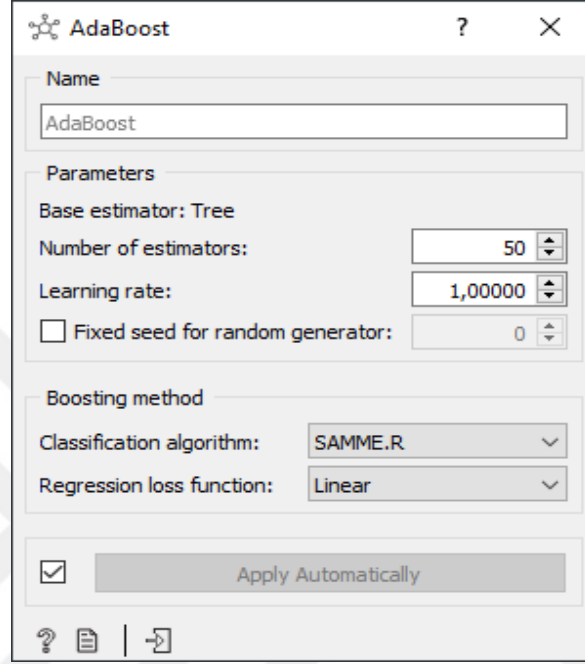
3.2.3. Adaboost algoritması

Adaptif boosting veya adaboost, en basit yükseltme algoritmalarından biridir. Genellikle, karar ağaçları modelleme için kullanılır. Her biri son modeldeki hataları düzelten çoklu sıralı modeller oluşturulur. Adaboost, yanlış tahmin edilen gözlemlere ağırlık atar ve sonraki model bu değerleri doğru şekilde tahmin etmek için çalışır (Ay, 2019).

Adaboost algoritmasını gerçekleştirme adımları aşağıda verilmiştir:

1. Başlangıçta, veri kümesindeki tüm gözlemlere eşit ağırlık verilir.
2. Bir model, veri alt kümesine dayanır.
3. Bu model kullanılarak, tüm veri seti için tahminler yapılmıştır.
4. Hatalar tahminler ve gerçek değerler karşılaştırılarak hesaplanır.
5. Bir sonraki modeli oluştururken, yanlış tahmin edilen veri noktalarına daha fazla ağırlık verilir.

6. Ağırlıklar hata değeri kullanılarak belirlenebilir. Örneğin, hata ne kadar yüksekse, gözlem için tayin edilen ağırlıktır.
7. Bu işlem, hata fonksiyonu değişmeden veya tahmin edicilerin sayısının maksimum limitine ulaşılan kadar tekrarlanır.

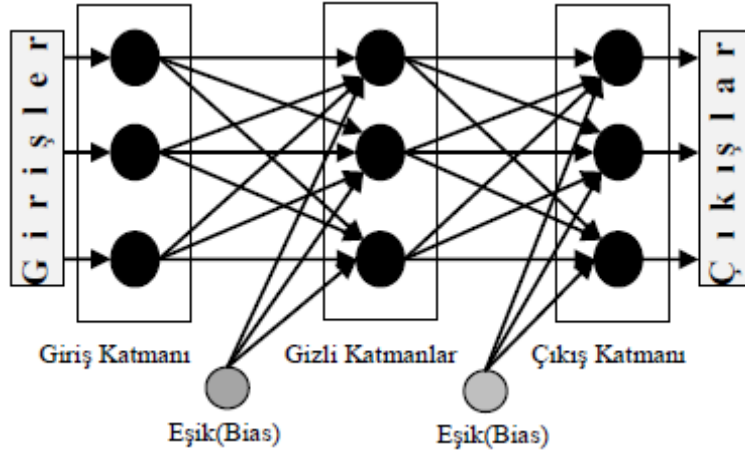


Şekil 3.11. Adaboost algoritmasının parametreleri.

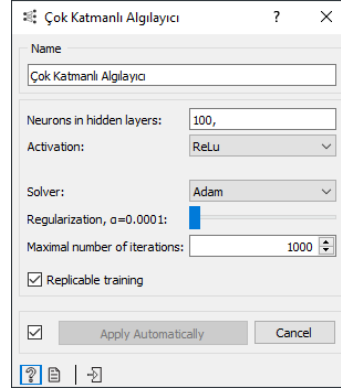
Algoritma modellemek için parametre seçimi kullanılan veri setinin boyutuna ve içerdiği değerlere göre tasarlayan kişi tarafından seçilmektedir. Çalışmada Orange yazılımının önerdiği varsayılan parametreler değiştirilmeden kullanılmıştır. Çalışmada tahminleyici sayısı 50, öğrenme oranı 1.0 ve sınıflandırma algoritması SAMME.R olarak hesaplamalar yapılmıştır. Öğrenme oranı yeni edinilen bilginin eski bilgileri ne ölçüde geçersiz kılacağını belirler (0 = temsilci hiçbir şey öğrenmez, 1 = temsilci yalnızca en son bilgileri dikkate alır). Sınıflandırma algoritması (sınıflandırma girdide ise) SAMME (temel tahmin edicinin ağırlıklarını sınıflandırma sonuçlarıyla günceller) veya SAMME.R (temel tahmin edicinin ağırlığını olasılık tahminleriyle günceller). Regresyon kaybı işlevi (girişte regresyon varsa) Lineer olarak belirlenmiştir.

3.2.4. Çok katmanlı algılayıcı algoritması

Yapay sinir ağları genel olarak tek katmanlı algılayıcılar ve çok katmanlı algılayıcılar olarak ikiye ayrılmaktadır. Tek katmanlı algılayıcıların doğrusal olmayan problemlerin çözmede yeterlilik sağlayamaması sebebiyle geliştirilen çok katmanlı algılayıcılar, veri girişinin yapıldığı bir girdi katmanı, bir veya daha fazla gizli (ara) katman ve bir çıktı katmanından meydana gelmektedir. Çok katmanlı algılayıcıda katmanlar arası ileri ve geri yayılım şeklinde isimlendirilen aşamalar vardır. İleri yayılım aşamasında, ağın çıktısı ve hata değeri hesaplanıp kaydedilir. Geri yayılım aşamasında ise hesaplanan hata değerinin küçültülmesi için katmanlar arası bağlantı ağırlık değerleri güncellenir. Şekil 3.3'te çok katmanlı algılayıcı örnek bir yapısı gösterilmiştir. Çok katmanlı algılayıcı modeli doğrusal algılayıcıdaki en küçük kareler algoritmasının genelleştirilmesi olan geri yayılım öğrenme algoritmasını kullanır. Geri yayılım algoritması, ağın çıktısının belirlendiği ileri besleme ve oluşan hatanın derecesini minimize edecek şekilde geri yayılarak ağırlıkların güncellendiği geri besleme aşamalarından oluşmaktadır.



Şekil 3.12. Çok katmanlı algılayıcı modeli.



Şekil 3.13. Çok katmanlı algılayıcı aracının özellikleri.

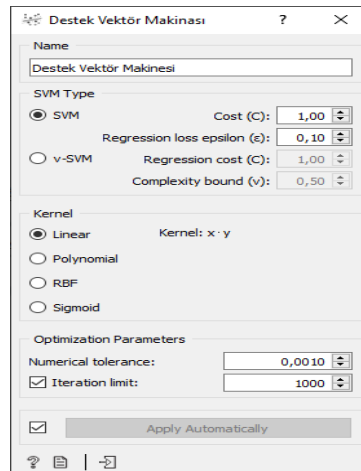
Algoritma modellemek için parametre seçimi kullanılan veri setinin boyutuna ve içerdiği değerlere göre tasarlayan kişi tarafından seçilmektedir. Çalışmada Orange yazılımının önerdiği varsayılan parametreler değiştirilmeden kullanılmıştır. Çalışmada çok katmanlı algılayıcı aracı için gizli katmanlardaki nöron sayısı değeri 100, aktivasyon algoritması ReLu (Rectified Linear Unit), çözücü olarak Adam, normalleştirme basamağı $\alpha=0.0001$, maksimum iterasyon sayısı 1000 olarak kullanılmıştır. Bir katman içindeki nöronların birbirleriyle ilişkileri yoktur ve sistemde akılda kalan bilgiyi sonraki katmana ya da çıkışa aktarma işini gerçekleştirirler (her bir nöronun vazifesidir). Arka arkaya iki katmandaki nöronlar birbirlerini çeşitli aktivasyon değerleriyle etkilemekte ve modelin öğrenme seviyesini belirleyen bir aktarım gerçekleştirmektedir. Eğer aktivasyon fonksiyonu uygulanmazsa çıkış sinyali basit bir doğrusal fonksiyon olur. Doğrusal fonksiyonlar yalnızca tek dereceli polinomlardır. Aktivasyon fonksiyonu kullanılmayan bir sinir ağı sınırlı öğrenme gücüne sahip bir doğrusal regresyon gibi davranacaktır fakat çalışmada çok katmanlı algılayıcımızın doğrusal olmayan durumları da öğrenmesi istenmektedir. ReLU fonksiyonu hesaplama yükünün sigmoid ve hiperbolik tanjant fonksiyonlarına göre az olması çok katmanlı ağlarda daha çok tercih edilmesine sebep olmuştur. Çözücü algoritması olarak kullanılan Adam, eğitim verilerine dayalı yinelemeli ağırlıklarını güncellemek için klasik olasılıksal dereceli azalma prosedürü yerine kullanılabilen bir optimizasyon algoritmasıdır. Adam çözücü hesaplama açısından verimli ve az hafıza gereksinimleri sebebiyle tercih edilmiştir. Normalleştirme basamağı ne kadar küçük olursa öğrenme o kadar verimli gerçekleşmesinden dolayı 0.0001 gibi çok küçük bir basamak alınmıştır.

Tekrarlama (iterasyon) sayısı algoritmanın en doğru tahmini bulmak için kaç kez çalışacağını göstermektedir.

3.2.5. Destek vektör makineleri algoritması

Destek vektör makineleri (DVM) yüksek boyutlu verilerde sıkça kullanılan bir ayırım metodudur. Veri üzerinde eğitilerek uygulanır. Eğitim için konveks optimizasyonunda kullanılan herhangi bir metot seçilebilir. Temelde düşük boyutta lineer olarak ayrılamayacak bir veri kümesini, daha yüksek boyuta taşıyarak bir düzlem yardımıyla ayırmayı sağlar.

Destek vektör makineleri istatistiksel öğrenme teorisine dayalı parametrik olmayan bir sınıflandırma yöntemidir (Vapnik, 1995). DVM ikili sınıflandırmalar için geliştirilmiştir ve bu yöntem ile az sayıda örnekleme verisi ile doğru sınıflandırma sonuçları elde etmek mümkündür (Foody ve Mathur, 2004). Özellik uzayında sınıflar arasında ki sınırı belirlemek için optimum algoritmanın kullanıldığı bir yöntemdir. Başlangıçta iki sınıflı doğrusal verilerin sınıflandırılması için tasarlanmış olan yöntem daha sonra çok sınıflı ve doğrusal olmayan verilerin sınıflandırılması için geliştirilmiştir. Temel olarak iki sınıfı birbirinden ayırabilen hiper düzlemin belirlenmesi prensibine dayanmaktadır (Vapnik, 1995). Bu yöntem kernel fonksiyonları kullanılarak çok boyutlu özellik uzayında çalışma özelliğine sahiptir ve yöntemden elde edilen sonuçlar seçilen kernel ve parametrelerin özelliklerine bağlıdır (Kavzoglu ve Çölkesen, 2009).



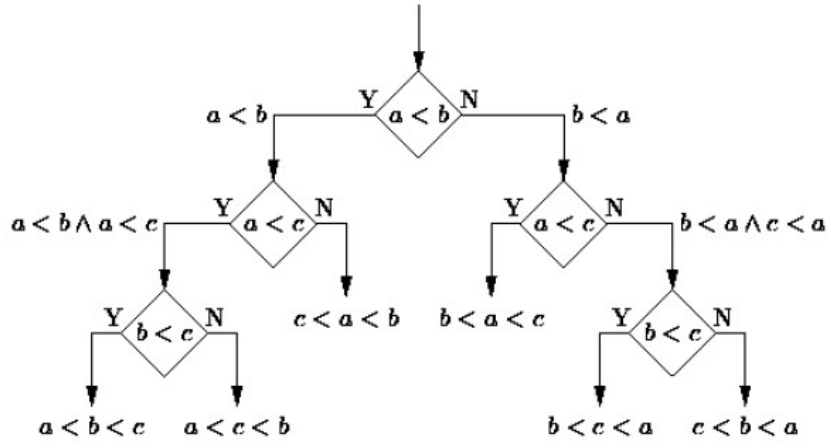
Şekil 3.14. Destek vektör makineleri aracının özellikleri.

Algoritma modellemek için parametre seçimi kullanılan veri setinin boyutuna ve içerdiği değerlere göre tasarlayan kişi tarafından seçilmektedir. Çalışmada Orange yazılımının önerdiği varsayılan parametreler değiştirilmeden kullanılmıştır. Çalışmada destek vektör makineleri aracı cost (C) değeri 1.00, Regression loss epsilon(ϵ) 0.10, çekirdek fonksiyonu olarak linear x.y, optimizasyon parametresi ayarları kısmında sayısal tolerans değeri 0.001 ve tekrarlama limit değeri 1000 olarak kullanılmıştır. Cost, sınıflandırma ve regresyon işlemleri için kayıp için ceza terimi olarak kullanılır. Regression loss epsilon, tahmin edilen değerlerle hiçbir cezanın ilişkilendirilmediği gerçek değerlere olan mesafeyi tanımlar. Çalışmamız için değiştirilmesi bir şey değiştirmemektedir. Çekirdek, nitelik uzayını maksimum marj hiper düzlemine uyacak şekilde yeni bir özellik uzayına dönüştüren ve böylece algoritmanın doğrusal, polinom, RBF ve sigmoid çekirdekleriyle modeli oluşturmaya izin veren bir işlemdir. Çekirdeği belirleyen işlevler seçildikten sonra sunulur. Çalışmada doğrusal çekirdek fonksiyonu kullanılmıştır. Tolerans değeri beklenen değerden izin verilen sapmayı ayarlamak için kullanılır. Tekrarlama sayısı algoritmanın en doğru tahmini bulmak için kaç kez çalışacağını göstermektedir.

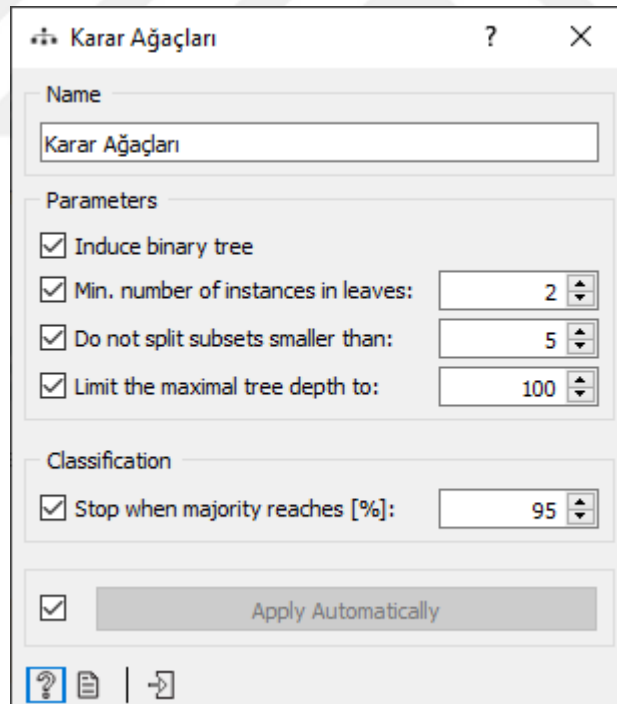
3.2.6. Karar ağaçları algoritması

Karar ağaçları algoritması düğümlerin sınanması yöntemiyle öğrenir ve ağaç şekillenir. Elde bulunan verilerin hangi düğümler kullanılarak en yüksek verimi verdiğini tespit etmek ve oluşturulan ağacı budama veya büyütme işlemi gerçekleştirmek üzerine kurulmuştur. Karar ağaçları tek bir noktadan başlayıp dallanarak genişler. Başlangıç noktasına kök denilmektedir. Kararın her bir seçeneğine dal denilmektedir. Dalın sonuna gelinmesi ile dallanma yapılmayacak bir duruma varılmış ise buna yaprağa gelindiği ifade edilir. Her dalda bir sınıflandırma sorusu bulunmaktadır. Her yaprakta ise dallanma sonucundaki sınıflardan birisi bulunur. Aralarındaki bağımlılığı yüksek olan değişkenler veri kümesi bu değişkenin değerine göre ikiye ayrılır ve ayrılmalar bir dal saflaşımına kadar bu şekilde devam etmektedir. Bu değişkenlerin birbiri arasındaki bağımlılığı yüksek ilişki veri setini ikiye ayırarak dallanmalara sebep olur. Bu dallanmalar ne kadar keskin olarak birbirinden ayrılıyorsa

sınıflama o kadar düzgün çalışıyor demektir. En iyi tahminin elde edilmesi toplanmış olan bağımlı ve bağımsız değişkenler arasındaki ilişkinin keşfedilmesi ile gerçekleşmektedir.



Şekil 3.15. Karar ağacı örneği.



Şekil 3.16. Karar ağaçları aracının özellikleri.

Algoritma modellemek için parametre seçimi kullanılan veri setinin boyutuna ve içerdiği değerlere göre tasarlayan kişi tarafından seçilmektedir. Çalışmada Orange yazılımının önerdiği varsayılan parametreler değiştirilmeden kullanılmıştır. Çalışmada

karar ağaçları algoritması aracında Induce binary tree özelliği ikili dallanmaların olması için seçilmiştir. Minimum yaprak sayısı 2 olarak belirlenmiştir. Bu sayede algoritma hiçbir zaman dalların herhangi birine belirtilen eğitim örneğinden daha azını koyacak bir ayırım oluşturmaz. “Do not split subsets smaller than” işlevi ile algoritmanın düğümleri verilen örnek sayısından yani beşten daha azıyla bölmesini yasaklar. “Limit the maximal tree depth” işlevi sınıflandırma ağacının derinliğini belirtilen düğüm seviyesi sayısı ile sınırlar. Çalışmada maksimum ağaç derinliği 100 olarak alınmıştır. “Stop when majority reaches [%]” işlevi modelin belirli bir çoğunluk eşliğine ulaşıldıktan sonra düğümleri bölmeyi durdurur ve %95 olarak belirlenmiştir.

3.2.7. K-en yakın komşu algoritması

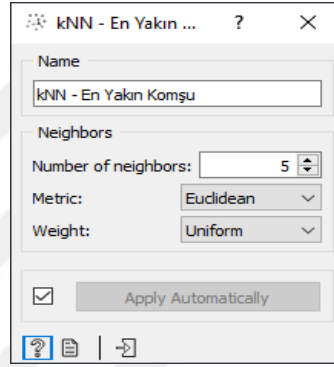
K-en yakın komşu algoritması, 19. yüzyıldan beri denetimli öğrenme ortamlarındaki verileri analiz etmek için yaygın olarak kullanılan bir makine öğrenme algoritmasıdır. Popülaritesi, kullanım kolaylığı ve kolayca anlaşılabilir sonuçlarından kaynaklanmaktadır. K-en yakın komşu algoritmasındaki “k”, bir veri kümesindeki sonuçları sınıflandırmak veya tahmin etmek için kullanılan en yakın komşuların sayısını ifade eder. Her yeni gözlemin sınıflandırması veya tahmini, ağırlıklı ortalamalara göre belirlenen bir mesafeye (yani en yakın komşu) göre hesaplanır. K-en yakın komşu analizi, verilerdeki gözlemlerin dağılımı hakkında önceden çok az bilgi olduğunda mantıklı bir seçimdir.

Her gözlem arasındaki ilişkiyi yakalamak amacıyla, etiketli bir veri seti verildiğinde K-en yakın komşu algoritması kullanılabilir. Bu benzerlik ilişkisi, veri noktaları arasında bir uzaklık metriği olarak ifade edilir. Gözlemler arasındaki aralıklar Öklid, Manhattan, Chebyshev, Hamming mesafesi ve Kosinüs benzerliği olarak ölçülebilir. K-en yakın komşu algoritması parametrik olmayan bir algoritmadır, çünkü ilişkinin fonksiyonel formu hakkında açık bir varsayım yoktur. Ayrıca, belirli bir veri modelini uygulamaktan ziyade doğrudan eğitim örneklerine odaklandığı anlamda örnek tabanlı öğrenmenin bir örneğidir.

K-en yakın komşu algoritması, verilerin bir test kümesinde eğitim yapmak yerine tüm veri kümesini işler. En yakın komşuların sayısı (k), veri seti için mümkün olan en iyi uyumu elde etmek için Veri bilimcisinin seçtiği hiper parametredir. K sayısı

küçük olduğunda, tahmin bölgesini sınırlandırırız ve sınıflandırıcıyı genel dağılımı daha az hassas hale getiririz. Sınıflandırma problemleri için bu en yaygın sınıf değeri olabilir. (k=3) Regresyon problemleri için, bu ortalama çıkış değişkeni (küme sayısı) olabilir.

Algoritma, veri araştırmacısının benzer belgeler için semantik olarak aradığı konsept araştırmasında kullanılır. Keşfetmek, algoritmanın verilen soruyla ilgili tüm e-postaları, kısa özetleri, sözleşmeleri, vb. bulmak için kullanılabileceği başka bir kullanım senaryosudur. K-en yakın komşu algoritması, yapay sinir ağları ve destek vektör makineleri gibi daha karmaşık sınıflandırıcılar için bir ölçüt olarak sıklıkla kullanılır.



Şekil 3.17. K-en yakın komşu aracının özellikleri.

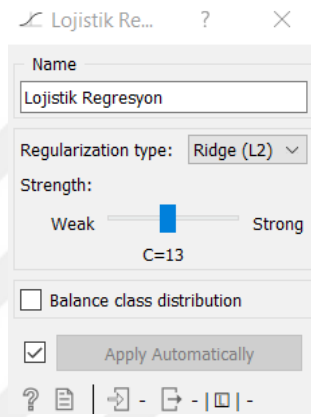
Algoritma modellemek için parametre seçimi kullanılan veri setinin boyutuna ve içerdiği değerlere göre tasarlayan kişi tarafından seçilmektedir. Çalışmada Orange yazılımının önerdiği varsayılan parametreler değiştirilmeden kullanılmıştır. En yakın komşuların sayısı model kriterleri olarak 5 ayarlanmıştır. Mesafe parametresini (metric) Euclidean olarak yani öklid iki nokta arasındaki mesafe olarak seçilmiştir. Ağırlık olarak da Uniform seçilerek her bölmedeki tüm noktalar eşit olarak ağırlıklandırılarak çalışmada kullanılmıştır.

3.2.8. Lojistik regresyon algoritması

Çok değişkenli istatistiksel verilerin sınıflandırılması, bu verilere uygulanabilecek çeşitli istatistiksel yöntemler için gerekli bir ön analiz olmasının yanı sıra pratikte başlı başına bir analiz olarak da sıkça kullanılmaktadır. Lojistik regresyon

analizi de sınıflandırma ve atama işlemlerini yapmak için kullanılan yöntemlerden biridir. Bu yöntem normal dağılım varsayımı veya süreklilik varsayımı gibi ön koşul gerektiren yöntemlere bir alternatif olarak geliştirilmiştir.

Lojistik model, bağımlı değişkenin 0, 1 gibi ikili ya da ikiden çok düzey içeren kesikli değişken olması durumunda normallik varsayımı sınırlaması olmaması nedeniyle kullanım rahatlığı sağlamaktadır (Tatlıdil 1996; Özdamar 1997). Lojistik regresyon, bağımlı değişkenin tahminini olasılık olarak hesaplayarak olasılık kurallarına uygun sınıflama işlemi yapma imkânı vermektedir.



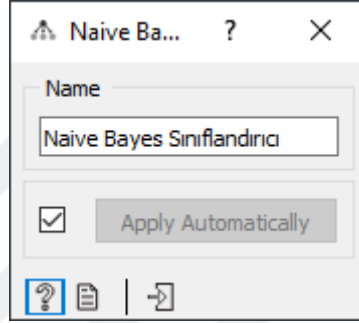
Şekil 3.18. Lojistik regresyon aracının özellikleri.

Algoritma modellemek için parametre seçimi kullanılan veri setinin boyutuna ve içerdiği değerlere göre tasarlayan kişi tarafından seçilmektedir. Çalışmada Orange yazılımının önerdiği varsayılan parametreler değiştirilmeden kullanılmıştır. Normalleştirme tipi Ridge (L2) olarak ve güç değeri $C=13$ olarak kullanılmıştır.

3.2.9. Naive Bayes sınıflandırıcı algoritması

Naive bayes algoritmasının temeli bayesli sınıflandırmalar yani koşullu olasılık teoremine dayanmaktadır. Sık kullanılan istatistiksel sınıflandırma yöntemlerden birisidir. Veri setinde yer alan bütün sınıflar için olasılık değerleri girdilerin özelliklerine göre koşullu olasılık kullanılarak hesaplanır. Hesaplanan olasılıklar ile her bir örneğe ait sınıf etiketi olasılıklar ile tahmin edilir. Sonradan gerçekleşecek belirsiz bir olayın geçmişte gerçekleşmiş durumlardan elde edilen olasılıklardan faydalanılır.

tahmin edilebiliyorsa bayesli sınıflandırıcı algoritmalar da veri setindeki örneklem kümesini geçmiş durum olarak olasılıklarını saptayıp gelecek durumlar için tahmin üretmektedir. Sınıflandırma tahmini için örneklerin bütün ilişkilerini olasılıksal olarak hesaplayıp, bundan yapılacak tahminin beklenen riskleri en düşük seviyede tutacak şekilde gerçekleştirme kaidesi üzerinden sınıflanama yapılmaktadır. (Alpaydın, 2014). Basit bayes sınıflandırıcılarda özelliklerin hepsinin önem düzeyi eşittir ve birbirinden bağımsızdır. Yani bir özelliğin değerinin diğerlerine bağlı olmadığı kabul edilir. Yine de çok fazla sayıda olasılığın hesap edilmesi koşulu vardır, ancak uygulaması kolaydır ve çoğunlukla başarılı sonuçlar üretmektedir. (Heckerman, 1999).



Şekil 3.19. Naive Bayes sınıflandırma aracının özellikleri.

Algoritma modellemek için parametre seçimi kullanılan veri setinin boyutuna ve içerdiği değerlere göre tasarlayan kişi tarafından seçilmektedir. Çalışmada Orange yazılımının önerdiği varsayılan parametreler değiştirilmeden kullanılmıştır. Naive bayes algoritmasının aracında özelleştirilebilir parametre bulunmamaktadır.

3.2.10. Olasılıksal dereceli azalma algoritması

Olasılıksal dereceli azalma, konveks kayıp fonksiyonları altında doğrusal sınıflandırıcıların ayırt edici öğreniminde basit ama çok etkili bir yaklaşımdır. Makine öğrenmesi konusunda yıllardır var olsa da çok büyük veri setleri kullanımlarının çoğalması ile beraber kullanım oranı gün geçtikçe artmaktadır (Bottou, 2010).

Olasılıksal dereceli azalma yönteminde hata fonksiyonun türevi, dereceli azalmada kullanılan yöntemin aksine, bir seferde tüm veri üzerinden hesaplanmak yerine her adımda sadece tek bir veri kullanılarak hesaplanır. Sonrasında W ve b

değerleri bu fonksiyonun türevine göre güncellenir ve bu işleme veri kümesindeki tüm veriler tamamlanana kadar devam edilir. Olasılıksal dereceli azalma yönteminde W ve b değerleri zincir kuralına göre hesaplanır. Zincir kuralına göre eğer y fonksiyonu x değişkenine göre türevi alınabilen bir fonksiyon ise türev alınabilir. Olasılıksal dereceli azalma algoritması rastgele olması nedeniyle, aynı başlangıç ağırlıkları ile her denemede farklı değerlerden geçerek sonuca ulaşmasıdır. Dereceli azalma yöntemi başlangıç ağırlıkları değişmediği sürece aynı yolu takip ederek aynı sonucu verecektir. Olasılıksal dereceli azalma ile bir yapay sinir ağı eğitimi yedi adımda özetlenebilir (Bağ, 2019).

1. Sıfıra yakın (fakat sıfır olmayacak şekilde) ilk ağırlık değerlerinin verilmesi,
2. Veri kümesindeki ilk satırının girilmesi ve sonuçların alınması.
3. Ağdaki her nöronda bir önceki katmandan gelen verinin işlenerek çıktının oluşturulması.
4. Çıktı ile gerçek değerin karşılaştırılması ve hatanın hesaplanması.
5. Geri yayılım algoritması kullanılarak ağırlıkların hataya göre güncellenmesi.
6. 1. maddeden 5. maddeye kadar olan sürecin tüm veri kümesi tamamlanana kadar devam ettirilmesi.
7. Tüm veri kümesi için çıktıların alınma ve ağırlıkların güncellenmesi işlemi bittikten sonra yeni bir döngüye başlanması.

Şekil 3.20. Olasılıksal dereceli azaltma aracının özellikleri.

Algoritma modellemek için parametre seçimi kullanılan veri setinin boyutuna ve içerdiği değerlere göre tasarlayan kişi tarafından seçilmektedir. Çalışmada Orange yazılımının önerdiği varsayılan parametreler değiştirilmeden kullanılmıştır. Çalışmada olasılıksal dereceli azaltma algoritmasını test etmek için Algoritma bölümünde Sınıflandırma kayıp fonksiyonu olarak hinge maksimum marj sınıflandırması için seçilmiştir. Regresyon kaybı fonksiyonu squared loss kare hata kaybının beklenen değerine karşılık gelen bir risk fonksiyonudur. Normalleştirme kısmında aşırı öğrenmeyi engellemek için normalleştirme metodu olarak standart normalleştirme fonksiyonu olan Ridge (L2) seçilmiştir, Düzeltme gücü ne kadar düzeltme uygulanacağını (ne kadar az düzenlersek, modelin verilere uymasına o kadar izin veririz) ifade etmektedir. Düzeltme gücünün değeri 0.00001 olarak sıfıra yakın bir değer seçilmiştir. Öğrenme parametleri bölümünde Öğrenme oranı constant yani sabit olarak bütün öğrenme adımlarında sabit olarak tutulmuştur. İlk öğrenme oranı varsayılan 0.01 olarak belirlenmiştir. Tolerans değeri beklenen değerden izin verilen sapmayı ayarlamak için kullanılır ve 0.001 olarak belirlenmiştir. İterasyon sayısı algoritmanın en doğru

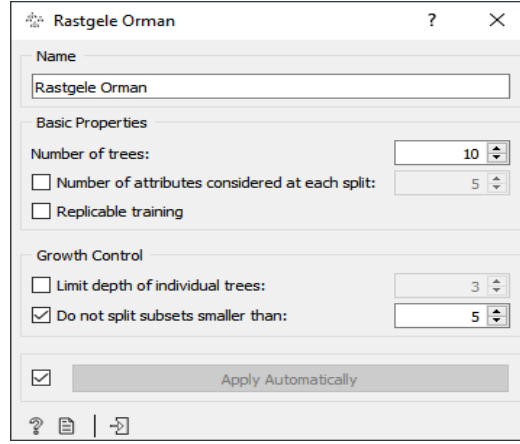
tahmini bulmak için kaç kez çalışacağını göstermektedir ve 1000 olarak belirlenmiştir. Ayrıca her iterasyondan sonra verilerin karışması için “shuffle data after each iteration” seçeneği aktif edilmiştir.

3.2.11. Rastgele orman algoritması

Toplu sınıflandırma yöntemleri, bir sınıflandırıcı yerine birden çok sınıflandırıcı üreten ve sonrasında onların tahminlerinden alınan oylar ile yeni veriyi sınıflandıran öğrenme algoritmalarıdır. En yaygın olarak kullanılan toplu sınıflandırıcılar, torbalama, hızlandırma ve rastgele ormandır. Rastgele orman, ağaç tipi sınıflandırıcılar topluluğu olarak tanımlanabilir. Rastgelelik özelliği eklenerek torbalama yönteminin geliştirilmiş bir versiyonudur (Breiman, 2001). Rastgele orman, tüm değişkenler arasından en iyi dalı kullanarak her bir düğümü dallara ayırmak yerine, her bir düğümde rastgele olarak seçilen değişkenler arasından en iyisini kullanarak her bir düğümü dallara ayırır. Her bir veri seti orijinal veri setinden yer değiştirmeli olarak üretilir. Sonra rastgele özellik seçimi kullanılarak ağaçlar geliştirilir. Geliştirilen ağaçlar budanmaz (Archer 2008; Breiman 2001). Bu strateji Rastgele ormanın doğruluğunu eşsiz yapar. Rastgele orman aynı zamanda çok hızlıdır, aşırı uyuma karşı dayanıklıdır ve ne kadar istenirse o kadar ağaçla çalışılır (Breiman ve Cutler, 2005).

Rastgele orman algoritmasını başlatmak için kullanıcı tarafından iki parametre tanımlanmalıdır. Bu parametreler, en iyi bölünmeyi belirlemek için her bir düğümde kullanılan değişkenlerin sayısı (m) ve geliştirilecek ağaçların sayısı N 'dir. İlk olarak eğitim veri setinin $2/3$ 'ünden önyükleme örnekleri oluşturulur. Eğitim veri setinin $1/3$ 'lük geri kalan kısmı hataları test etmek için kullanılır. Sonra her bir önyüklemeli örnekten budama olmadan ağaç geliştirilir. Her bir düğümde m değişkenleri tüm değişkenler arasından rastgele olarak seçilir ve bu değişkenler arasından en iyi dal belirlenir. Yeterli öngörü gücü ile yeterli miktarda düşük korelasyon sağlayan değişken sayısının seçimi son derece önemlidir (Horning, 2010). Breiman (2001)'e göre toplam m adet değişken sayısının kare köküne eşit alınan m değişken sayısı genel olarak optimum sonuca en yakın sonucu verir. Rastgele orman ağaç üretmek için CART (Classification and Regression Tree) algoritmasını kullanır (Breiman, 2001). Her bir

düğüm de dallar CART algoritmasının kriterine (örneğin GINI indeksi) göre oluşturulur.

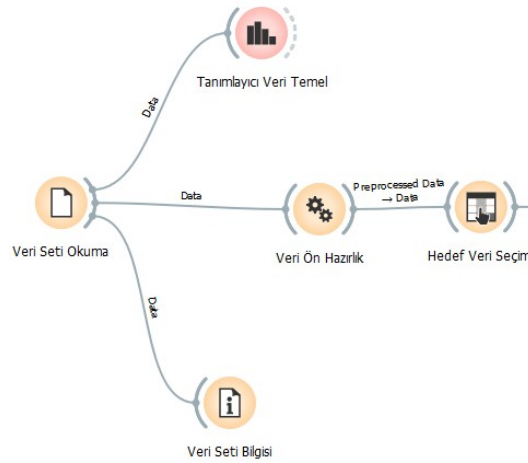


Şekil 3.21. Rastgele orman aracının özellikleri.

Algoritma modellemek için parametre seçimi kullanılan veri setinin boyutuna ve içerdiği değerlere göre tasarlayan kişi tarafından seçilmektedir. Çalışmada Orange yazılımının önerdiği varsayılan parametreler değiştirilmeden kullanılmıştır. Çalışmadan kullanılan rastgele orman algoritması test edilirken ağaç sayısı 10 olarak belirlenmiş ve alt grupların beşten daha küçük parçaya bölünmemesi istenilmiştir.

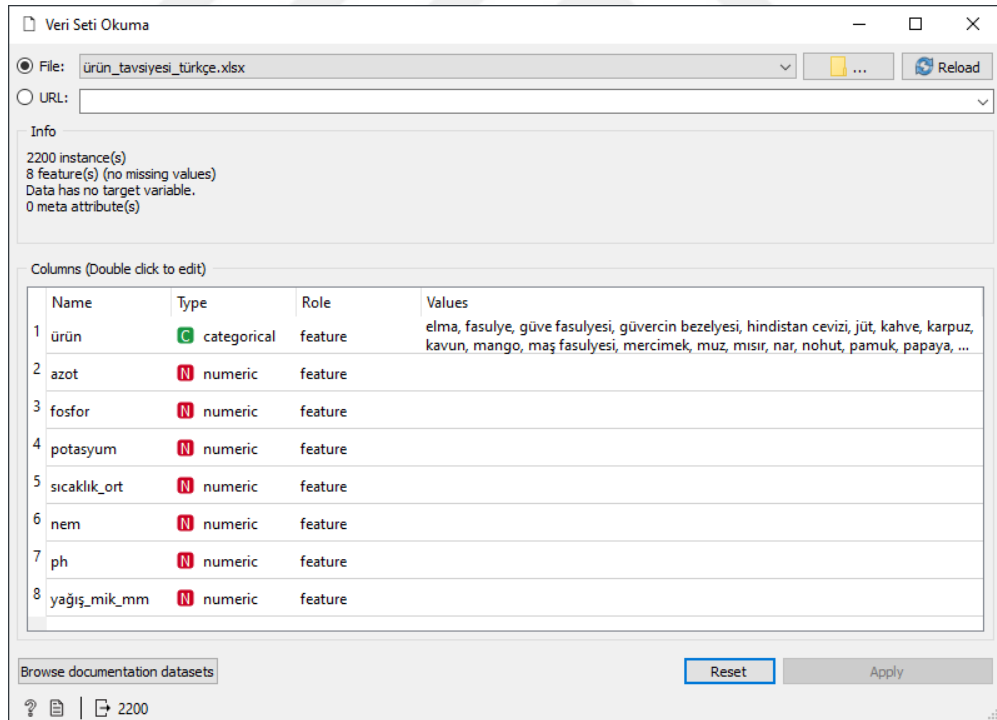
3.2.12. Veri seti ön hazırlık işlemleri

Çalışmalarda kullanılacak olan veri setleri her zaman hemen kullanıma hazır olarak bulunmaz. Bu sebeple bir ön hazırlık aşamasına tabi tutulur. Çalışmamızda öncelikle veri setini kullanacağımız programa veya yazılım diline okutmamız gerekmektedir. Orange yazılımı ile veri okuma işlemleri de araçlar aracılığıyla yapılmaktadır (Şekil 3.13).



Şekil 3.22. Veri setinin ön hazırlık aşamaları.

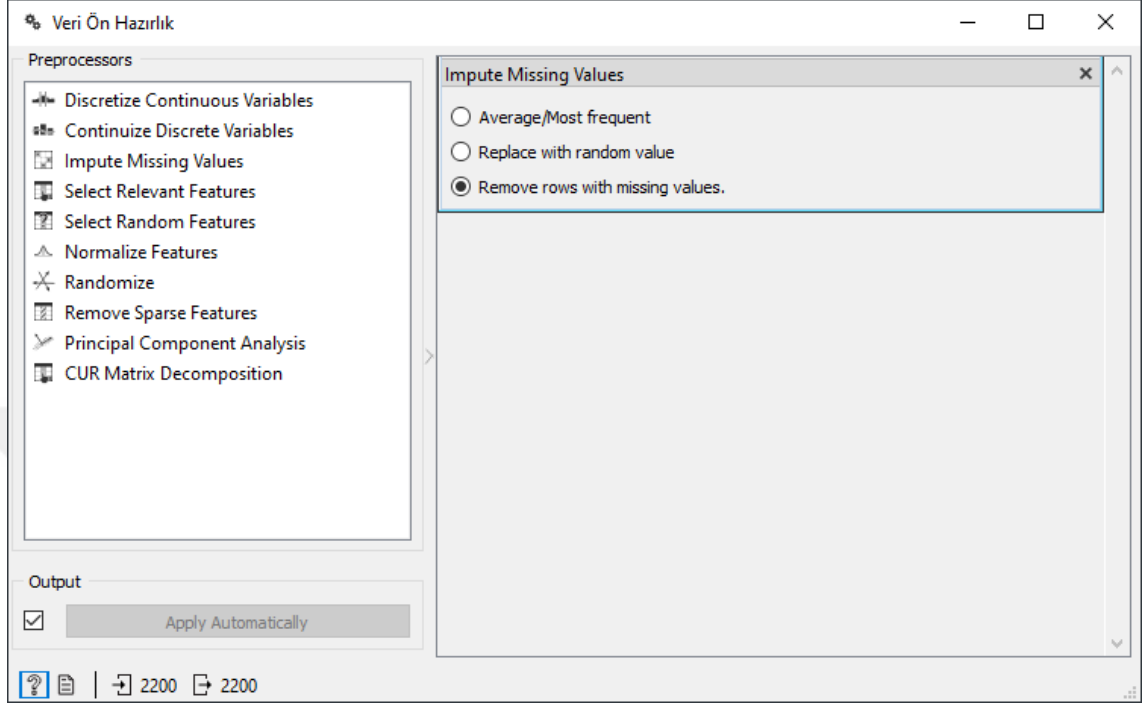
Veri seti okutulan bölümde özellikler otomatik olarak algılanmaktadır. Eğer yanlış kategoride işlem görmüş veri türü varsa bu aşamada düzeltme uygulanabilir.



Şekil 3.23. Veri setinin Orange yazılımı tarafından okunması.

Ardından çalışmada kullandığımız veri setinde eksik değer (missing value) var ise buna nasıl bir düzeltme uygulayacağımız önışleme (preprocess) aracı aracılığıyla

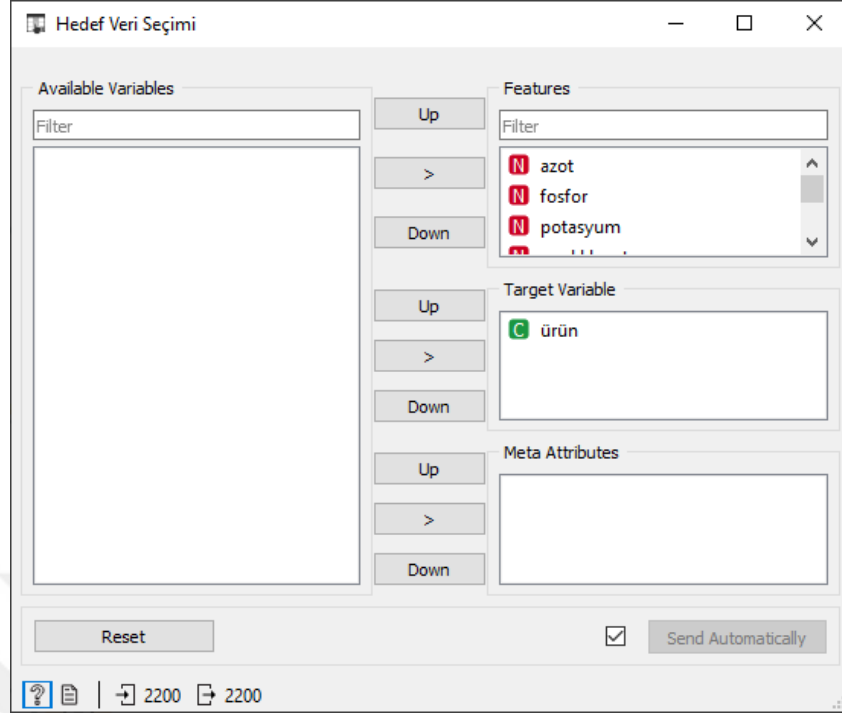
yapılmaktadır. Çalışmada eksik değer varsa o satırı tamamen silme komutu verilmiştir (Şekil 3.15).



Şekil 3.24. Eksik verilerin ne yapılacağını seçilmesi.

3.2.13. Hedef veri seçimi

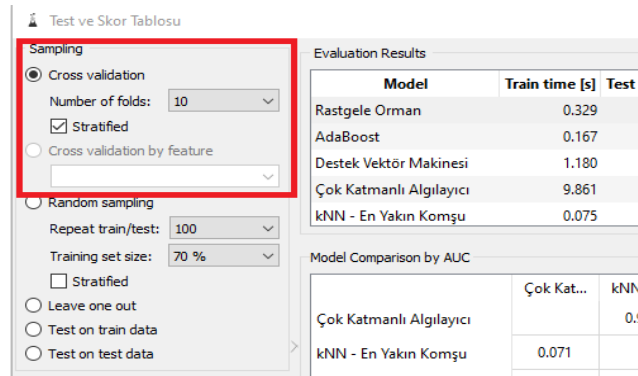
Çalışmada kullanılacak olan veriler arasında tahmin edilmesini veya sınıflandırılmasının yapılmasını istenilen sütun seçimi yapılmalıdır. Hedef sütun seçimi için “Select Column” aracı kullanılır. Bu aşamada birden çok sütun seçilebilmektedir. Fakat çalışmada tek bir sütun (ürün) üzerinden sınıflandırma işlemi yapılmıştır (Şekil 3.16).



Şekil 3.25. Orange ile hedef veri seçimi.

3.2.14. Veri setinin bölünmesi

Makine öğrenmesi algoritmalarının sınıflandırma işlemi yapabilmesi için çalışmada kullanılacak olan veri setini eğitim veri seti ve test veri seti olarak ikiye ayrılması gerekmektedir. Çalışmada veri seti “Test & Score” aracı kullanılarak çapraz doğruluk yöntemi ile bölünmüştür. Katlanma sayısı (number of folds) 10 olarak seçilmiştir (Şekil 3.17).



Şekil 3.26. Veri setinin eğitim ve test verisi olarak ayarlanması.

3.2.15. Çapraz doğruluk

Çapraz doğrulama (cross validation), yapılan bir istatistiksel analizin bağımsız bir veri setinde nasıl bir sonuç elde edeceğini sınavan bir model doğrulama tekniğidir. Başlıca kullanım alanı bir öngörü sisteminin pratikte hangi doğrulukla çalışacağını kestirmektir. Bir öngörü probleminde, model genellikle bir "bilinen veri" kümesiyle eğitilir ("eğitim kümesi") ve bir "bilinmeyen veri" kümesiyle ("doğrulama kümesi" ya da "test kümesi") sınanır. Bu sınavanın amacı, eğitilen modelin yeni verilere genelleşme kabiliyetini ölçmek ve aşırı uyma ya da seçim yanlılığı problemlerini tespit etmektir.

Çapraz doğrulamanın ilk adımı farklı miktarlarda örnekler seçilerek veri kümeleri oluşturulmasıdır. Tipik olarak verilen eğitim kümesi ve test kümesi olarak birbirini tamamlayan iki kümeye ayrılır. Birinci küme üzerinde analiz yapılarak model oluşturulur ve ikinci küme üzerinde model sınanır. Varyansı azaltmak için bu doğrulama adımı birkaç defa tekrarlanır. Çapraz doğrulama ismini aynı veri örneklerinin farklı şekillerde gruplanmasından (çaprazlama) alır. Çaprazlanan gruplar üzerinde yapılan sınav sonuçları birleştirilerek (örneğin ortalama) modelin genel tahmin başarısı ölçülür (Anonim, 2020a).

3.3. İncelenen Özellikler

Hata Matrisi: Karışıklık matrisi olarak da bilinmektedir. Gerçek değerleri önceden bilinen bir dizi test verisi üzerinde, bir sınıflandırma modelinin performans ölçütlerini ifade etmek için kullanılan bir tablodur (Şekil 3.18).

		Tahmin Edilen		
		Pozitif	Negatif	
Gerçekte Olan	Pozitif	Doğru Pozitif (TP)	Yanlış Negatif (FN)	Gerçek Pozitifler Toplamı
	Negatif	Yanlış Pozitif (FP)	Doğru Negatif (TN)	Gerçek Negatifler Toplamı
TOPLAM				

Şekil 3.27. Hata matrisi tablosu.

Doğru Pozitifler (TP): Gerçek değeri 1 ve tahmin edilen değerin de 1 olduğu örneklerdir.

Doğru Negatifler (TN): Gerçek değeri 0 ve tahmin edilen değerin de 0 olduğu örneklerdir.

Yanlış Pozitifler (FP): Gerçek değeri 0 ancak tahmin edilen değerin 1 olduğu örneklerdir.

Yanlış Negatifler (FN): Gerçek değeri 1 ancak tahmin edilen değerin 0 olduğu örneklerdir.

Karışıklık matrisinden çıkarılan sonuçlardan hesaplanan bazı değerler bulunmaktadır. TP, TN, FP ve FN'nin birbirleriyle ilişkisini gösteren bu kavramlar aşağıda verilmiştir.

$$Toplam = TP + TN + FP + FN \quad (3.1)$$

$$Gerçek Pozitifler Toplamı = TP + FN \quad (3.2)$$

$$Gerçek Negatifler Toplamı = TN + FP \quad (3.3)$$

Doğruluk Oranı: Genel olarak, sınıflayıcının ne sıklıkta doğru tahmin ettiğinin bir ölçüsüdür.

$$Doğruluk Oranı = (TP + TN) / Toplam \quad (3.4)$$

Yanlış Sınıflandırma Oranı: Sınıflayıcının ne sıklıkta yanlış tahmin ettiğinin bir ölçüsüdür. Hata Oranı (error rate) olarak da bilinir.

$$Hata Oranı = (FP + FN) / Toplam \quad (3.5)$$

Doğru Pozitif Değerlerin Oranı: Sınıflayıcının ne kadar doğru pozitif değerini doğru tahmin ettiğinin bir ölçüsüdür. Hassasiyet (sensitivity, hit rate ya da recall), isabet oranı veya hatırlama olarak da bilinir. Mümkün olduğu kadar yüksek olmalıdır.

$$\text{Hassasiyet} = TP / \text{Gerçek Pozitifler Toplamı} \quad (3.6)$$

Doğru Negatif Değerlerin Oranı: Sınıflayıcının ne kadar doğru negatif değerini doğru tahmin ettiğinin bir ölçüsüdür. Özgüllük veya Seçicilik olarak da bilinir. Mümkün olduğu kadar yüksek olması istenmektedir.

$$\text{Özgüllük} = TN / \text{Gerçek Negatifler Toplamı} \quad (3.7)$$

Yanlış Pozitif Değerlerin Oranı: Gerçek değeri 0 olmasına karşın 1 olarak tahmin edilenlerin oranıdır. Yan ürün (Fall-out) olarak da bilinir.

$$\text{Yan Ürün} = FP / \text{Gerçek Negatifler Toplamı} \quad (3.8)$$

Yanlış Negatif Değerlerin Oranı: Gerçek değeri 1 olmasına karşın 0 olarak tahmin edilenlerin oranıdır. Kayıp oranı olarak da bilinir. (LogLoss / Miss Rate) mümkün olduğu kadar küçük olması istenilir.

$$\text{Kayıp Oranı} = FN / \text{Gerçek Pozitifler Toplamı} \quad (3.9)$$

Kesinlik: Tüm sınıflardan, doğru olarak ne kadar tahmin edildiğinin bir ölçüsüdür. Mümkün olduğu kadar yüksek olmalıdır. Pozitif tahmin edici değer olarak da bilinir.

$$\text{Kesinlik} = TP / TP + FP \quad (3.10)$$

F1 Skoru: Bu değer, hassasiyet ve kesinlik değerlerinin harmonik ortalamasıdır. Sınıflandırıcının ne kadar iyi performans gösterdiğinin bir ölçüsüdür ve sınıflandırıcıları karşılaştırmakta sıklıkla kullanılır.

$$F1 \text{ Skoru} = \frac{2 * \text{Kesinlik} * \text{Hassasiyet}}{\text{Kesinlik} + \text{Hassasiyet}} \quad (3.11)$$

ROC Eğrisi (ROC Curve): Sınıflandırıcının bütün olası değerler üzerinde performansını özetlemek için kullanılan bir grafikdir. Belirli bir sınıfa gözlem atanması eşiğini değiştirdiğinizde hassasiyet (x eksen) karşı özgüllük (y eksen) çizerek oluşturulur. ROC Eğrisi, hassasiyet / özgüllük raporu oluşturmaya yarar. ROC eğrisinin altındaki alan (Area Under Curve (AUC)), bir parametrenin iki sınıf arasında ne kadar iyi ayırt edilebileceğinin bir ölçüsüdür (Anonim, 2017).

Eğitim Süresi: Veri setini eğitmek için geçen süredir. Bu sürenin mümkün olduğunda kısa olması istenilmektedir.

Test Süresi: Veri setinin eğitimi bittikten sonra doğru veriler ile test edilme aşamasıdır. Bu sürenin mümkün olduğunda kısa olması istenilmektedir.

3.4. Seçilen Modelin Programlanması

Orange yazılımı ile incelenecek sekiz tane makine öğrenmesi modeli arasından çapraz doğruluk metodu ile en yüksek F1 skoru bulunan model için python dili ve kütüphaneleri ile tahminleme modeli oluşturulacaktır. Orange yazılımı olmadan da python dili ve kütüphaneleri ile en iyi algoritmanın tespiti yapılabilmektedir fakat Orange yazılımı ile çok kısa bir süre birçok model test edilip kodlama yapmadan model seçimi yapmak mümkündür.

3.5. Modelin Ara Yüze Uyarlanması

Kullanıcı ara yüzü olmayan bir uygulama pratik fayda sağlaması mümkün değildir. Bu tasarlanan uygulamayı herhangi bir çiftçi web üzerinden bağlanıp kullanabilecek bir seviyede basitlikte tasarlanması sayesinde amaca hizmet etmiş olacaktır. Çalışmada python dili ve kütüphaneleri ile modellenen makine öğrenmesi algoritmasının kullanıcıdan alınan azot, fosfor, potasyum, sıcaklık ortalaması, nem, pH, yağış miktarı değerleri girilerek modelden en iyi üretilcek ürünün tahmin edilmesi istenilecektir. Bu tahminleme ara yüzü python dili ile yazılan streamlit kütüphanesi ile yapılacaktır.



4. BULGULAR

4.1. Tanıtıcı İstatistikler

Çalışmada kullanılan veri setinde yer alan elma, fasulye, güve fasulyesi, güvercin bezelyesi, hindistan cevizi, jüt, kahve, karpuz, kavun, mango, maş fasulyesi, mercimek, mısır, muz, nar, nohut, pamuk, papaya, pirinç, portakal, siyah mercimek ve üzüm olarak 22 farklı ürünün azot (%), fosfor (%), potasyum (%), sıcaklık ortalamaları (°C), nemlilik (%), pH ve yıllık yağış miktarı (mm) ortalamaları için gözlem sayısı (n), ortalama, standart sapma, minimum ve maksimum değerlerinin gösterildiği tanıtıcı istatistikler aşağıdaki çizelgelerde yer almaktadır.

Çizelge 4.2. Elma için tanıtıcı istatistikler

	n	Ortalama	Standart Sapma	Minimum	Maksimum
Azot	100	20.800	11.864	0.000	40.000
Fosfor	100	134.220	8.140	120.000	145.000
Potasyum	100	199.890	3.321	195.000	205.000
Sıcaklık Ort.	100	22.631	0.827	21.037	23.997
Nemlilik	100	92.333	1.459	90.026	94.920
pH	100	5.930	0.269	5.514	6.499
Yağış Mik. Ort	100	112.655	7.103	100.117	124.983

Çizelge 4.3. Fasulye için tanıtıcı istatistikler

	n	Ortalama	Standart Sapma	Minimum	Maksimum
Azot	100	20.750	10.834	0.000	40.000
Fosfor	100	67.540	7.571	55.000	80.000
Potasyum	100	20.050	3.102	15.000	25.000
Sıcaklık Ort.	100	20.115	2.600	15.330	24.924
Nemlilik	100	21.605	2.167	18.092	24.970
pH	100	5.749	0.145	5.503	5.998
Yağış Mik. Ort	100	105.920	26.111	60.276	149.744

Çizelge 4.4. Güve fasulyesi için tanıtıcı istatistikler

	n	Ortalama	Standart Sapma	Minimum	Maksimum
Azot	100	21.440	11.343	0.000	40.000
Fosfor	100	48.010	7.547	35.000	60.000
Potasyum	100	20.230	3.048	15.000	25.000
Sıcaklık Ort.	100	28.195	2.210	24.018	31.999
Nemlilik	100	53.160	6.995	40.009	64.956
pH	100	6.831	1.862	3.505	9.935
Yağış Mik. Ort	100	51.198	13.751	30.920	74.443

Çizelge 4.5. Güvercin bezelyesi için tanıtıcı istatistikler

	n	Ortalama	Standart Sapma	Minimum	Maksimum
Azot	100	20.730	11.850	0.000	40.000
Fosfor	100	67.730	7.294	55.000	80.000
Potasyum	100	20.290	2.815	15.000	25.000
Sıcaklık Ort.	100	27.742	5.716	18.319	36.978
Nemlilik	100	48.062	10.964	30.400	69.691
pH	100	5.794	0.830	4.548	7.445
Yağış Mik. Ort	100	149.458	32.986	90.054	198.830

Çizelge 4.6. Hindistan cevizi için tanıtıcı istatistikler

	n	Ortalama	Standart Sapma	Minimum	Maksimum
Azot	100	21.980	11.762	0.000	40.000
Fosfor	100	16.930	8.357	5.000	30.000
Potasyum	100	30.590	2.999	25.000	35.000
Sıcaklık Ort.	100	27.410	1.386	25.009	29.869
Nemlilik	100	94.844	2.686	90.017	99.982
pH	100	5.977	0.288	5.502	6.470
Yağış Mik. Ort	100	175.687	29.472	131.090	225.632

Çizelge 4.7. Jüt için tanıtıcı istatistikler

	n	Ortalama	Standart Sapma	Minimum	Maksimum
Azot	100	78.400	10.968	60.000	100.000
Fosfor	100	46.860	7.196	35.000	60.000
Potasyum	100	39.990	3.314	35.000	45.000
Sıcaklık Ort.	100	24.958	1.185	23.094	26.986
Nemlilik	100	79.640	5.508	70.883	89.891
pH	100	6.733	0.446	6.003	7.488
Yağış Mik. Ort	100	174.793	15.078	150.236	199.836

Çizelge 4.8. Kahve için tanıttıcı istatistikler

	n	Ortalama	Standart Sapma	Minimum	Maksimum
Azot	100	101.200	12.345	80.000	120.000
Fosfor	100	28.740	7.276	15.000	40.000
Potasyum	100	29.940	3.247	25.000	35.000
Sıcaklık Ort.	100	25.540	1.503	23.060	27.924
Nemlilik	100	58.870	5.839	50.046	69.948
pH	100	6.790	0.417	6.021	7.493
Yağış Mik. Ort	100	158.066	25.695	115.156	199.474

Çizelge 4.9. Karpuz için tanıttıcı istatistikler

	n	Ortalama	Standart Sapma	Minimum	Maksimum
Azot	100	99.420	12.565	80.000	120.000
Fosfor	100	17.000	7.536	5.000	30.000
Potasyum	100	50.220	3.265	45.000	55.000
Sıcaklık Ort.	100	25.592	0.850	24.044	26.986
Nemlilik	100	85.160	2.957	80.026	89.984
pH	100	6.496	0.281	6.001	6.957
Yağış Mik. Ort	100	50.786	5.867	40.127	59.760

Çizelge 4.10. Kavun için tanıttıcı istatistikler

	n	Ortalama	Standart Sapma	Minimu m	Maksimum
Azot	100	100.320	12.176	80.000	120.000
Fosfor	100	17.720	7.187	5.000	30.000
Potasyum	100	50.080	3.218	45.000	55.000
Sıcaklık Ort.	100	28.663	0.861	27.024	29.943
Nemlilik	100	92.343	1.505	90.015	94.962
pH	100	6.359	0.232	6.003	6.781
Yağış Mik. Ort	100	24.690	2.774	20.211	29.867

Çizelge 4.11. Mango için tanıttıcı istatistikler

	n	Ortalama	Standart Sapma	Minimu m	Maksimum
Azot	100	20.070	12.329	0.000	40.000
Fosfor	100	27.180	7.664	15.000	40.000
Potasyum	100	29.920	3.097	25.000	35.000
Sıcaklık Ort.	100	31.209	2.654	27.003	35.990
Nemlilik	100	50.157	2.756	45.022	54.964
pH	100	5.766	0.704	4.508	6.967
Yağış Mik. Ort	100	94.705	3.339	89.291	100.812

Çizelge 4.12. Maş fasulyesi için tanıttıcı istatistikler

	n	Ortalama	Standart Sapma	Minimum	Maksimum
Azot	100	20.990	11.511	0.000	40.000
Fosfor	100	47.280	7.870	35.000	60.000
Potasyum	100	19.870	3.148	15.000	25.000
Sıcaklık Ort.	100	28.526	0.842	27.015	29.915
Nemlilik	100	85.500	2.851	80.035	89.996
pH	100	6.724	0.287	6.219	7.199
Yağış Mik. Ort	100	48.404	7.117	36.120	59.872

Çizelge 4.13. Mercimek için tanıttıcı istatistikler

	n	Ortalama	Standart Sapma	Minimum	Maksimum
Azot	100	18.770	12.197	0.000	40.000
Fosfor	100	68.360	7.335	55.000	80.000
Potasyum	100	19.410	2.968	15.000	25.000
Sıcaklık Ort.	100	24.509	3.321	18.065	29.944
Nemlilik	100	64.805	2.935	60.091	69.924
pH	100	6.928	0.549	5.916	7.841
Yağış Mik. Ort	100	45.680	5.633	35.035	54.939

Çizelge 4.14. Mısır için tanıtıcı istatistikler

	n	Ortalama	Standart Sapma	Minimum	Maksimum
Azot	100	77.760	11.949	60.000	100.000
Fosfor	100	48.440	8.010	35.000	60.000
Potasyum	100	19.790	2.942	15.000	25.000
Sıcaklık Ort.	100	22.389	2.679	18.042	26.550
Nemlilik	100	65.092	5.449	55.282	74.829
pH	100	6.245	0.413	5.514	6.996
Yağış Mik. Ort	100	84.767	15.547	60.652	109.752

Çizelge 4.15. Muz için tanıtıcı istatistikler

	n	Ortalama	Standart Sapma	Minimum	Maksimum
Azot	100	100.230	11.107	80.000	120.000
Fosfor	100	82.010	7.690	70.000	95.000
Potasyum	100	50.050	3.383	45.000	55.000
Sıcaklık Ort.	100	27.377	1.428	25.010	29.909
Nemlilik	100	80.358	2.805	75.032	84.978
pH	100	5.984	0.271	5.505	6.490
Yağış Mik. Ort	100	104.627	9.398	90.110	119.848

Çizelge 4.16. Nar için tanıtıcı istatistikler

	n	Ortalama	Standart Sapma	Minimum	Maksimum
Azot	100	18.870	12.618	0.000	40.000
Fosfor	100	18.750	7.387	5.000	30.000
Potasyum	100	40.210	3.033	35.000	45.000
Sıcaklık Ort.	100	21.838	2.206	18.071	24.963
Nemlilik	100	90.126	2.826	85.129	94.999
pH	100	6.429	0.492	5.562	7.200
Yağış Mik. Ort	100	107.528	2.895	102.518	112.475

Çizelge 4.17. Nohut için tanıtıcı istatistikler

	n	Ortalama	Standart Sapma	Minimum	Maksimum
Azot	100	40.090	12.151	20.000	60.000
Fosfor	100	67.790	7.499	55.000	80.000
Potasyum	100	79.920	3.262	75.000	85.000
Sıcaklık Ort.	100	18.873	1.170	17.025	20.995
Nemlilik	100	16.860	1.712	14.258	19.970
pH	100	7.337	0.801	5.989	8.869
Yağış Mik. Ort	100	80.059	7.944	65.114	94.782

Çizelge 4.18. Pamuk için tanıtıcı istatistikler

	n	Ortalama	Standart Sapma	Minimum	Maksimum
Azot	100	117.770	11.629	100.000	140.000
Fosfor	100	46.240	7.349	35.000	60.000
Potasyum	100	19.560	3.170	15.000	25.000
Sıcaklık Ort.	100	23.989	1.136	22.001	25.992
Nemlilik	100	79.843	3.051	75.005	84.877
pH	100	6.913	0.627	5.801	7.995
Yağış Mik. Ort	100	80.398	11.232	60.654	99.931

Çizelge 4.19. Papaya için tanıtıcı istatistikler

	n	Ortalama	Standart Sapma	Minimum	Maksimum
Azot	100	49.880	12.220	31.000	70.000
Fosfor	100	59.050	7.057	46.000	70.000
Potasyum	100	50.040	3.097	45.000	55.000
Sıcaklık Ort.	100	33.724	6.263	23.012	43.675
Nemlilik	100	92.403	1.420	90.039	94.945
pH	100	6.741	0.147	6.502	6.993
Yağış Mik. Ort	100	142.628	64.397	40.352	248.859

Çizelge 4.20. Pirinç için tanıtıcı istatistikler

	n	Ortalama	Standart Sapma	Minimum	Maksimum
Azot	100	79.890	11.918	60.000	99.000
Fosfor	100	47.580	7.905	35.000	60.000
Potasyum	100	39.870	2.946	35.000	45.000
Sıcaklık Ort.	100	23.689	2.031	20.045	26.930
Nemlilik	100	82.273	1.418	80.123	84.969
pH	100	6.425	0.769	5.005	7.868
Yağış Mik. Ort	100	236.181	34.292	182.562	298.560

Çizelge 4.21. Portakal için tanıtıcı istatistikler

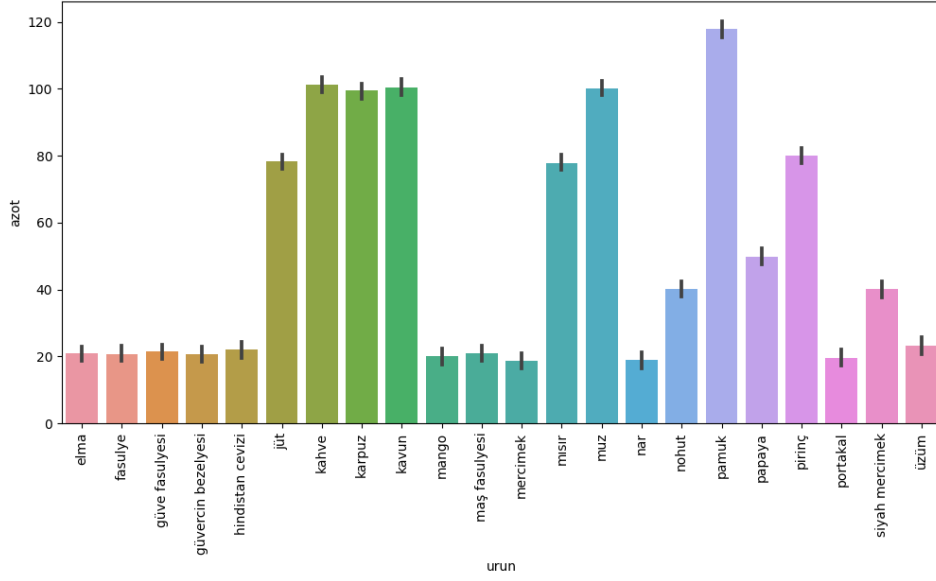
	n	Ortalama	Standart Sapma	Minimum	Maksimum
Azot	100	19.580	11.942	0.000	40.000
Fosfor	100	16.550	7.691	5.000	30.000
Potasyum	100	10.010	3.057	5.000	15.000
Sıcaklık Ort.	100	22.766	7.343	10.011	34.907
Nemlilik	100	92.170	1.430	90.006	94.964
pH	100	7.017	0.576	6.010	7.996
Yağış Mik. Ort	100	110.475	5.717	100.174	119.695

Çizelge 4.22. Siyah mercimek için tanıttıcı istatistikler

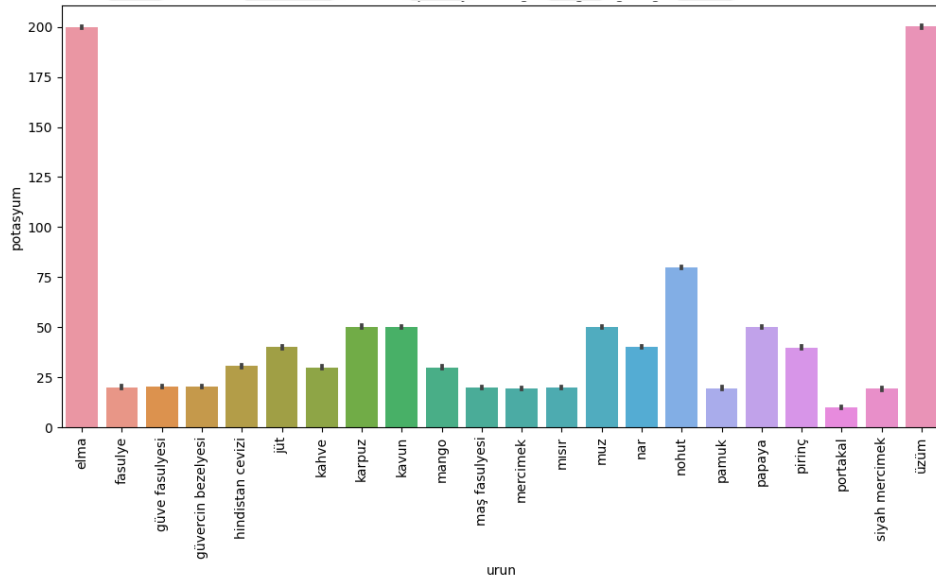
	n	Ortalama	Standart Sapma	Minimu m	Maksimum
Azot	100	40.020	12.664	20.000	60.000
Fosfor	100	67.470	7.151	55.000	80.000
Potasyum	100	19.240	3.188	15.000	25.000
Sıcaklık Ort.	100	29.973	2.686	25.097	34.947
Nemlilik	100	65.118	2.798	60.065	69.961
pH	100	7.134	0.373	6.500	7.775
Yağış Mik. Ort	100	67.884	4.184	60.418	74.916

Çizelge 4.23. Üzüm için tanıttıcı istatistikler

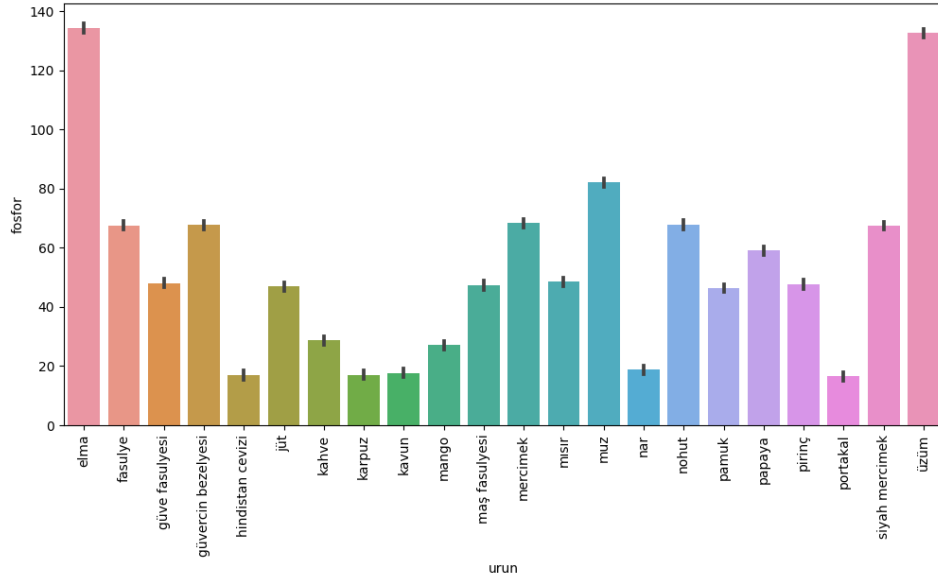
	n	Ortalama	Standart Sapma	Minimu m	Maksimum
Azot	100	23.180	12.467	0.000	40.000
Fosfor	100	132.530	7.619	120.000	145.000
Potasyum	100	200.110	3.266	195.000	205.000
Sıcaklık Ort.	100	23.850	9.739	8.826	41.949
Nemlilik	100	81.875	1.177	80.016	83.984
pH	100	6.026	0.298	5.511	6.500
Yağış Mik. Ort	100	69.612	2.952	65.011	74.915



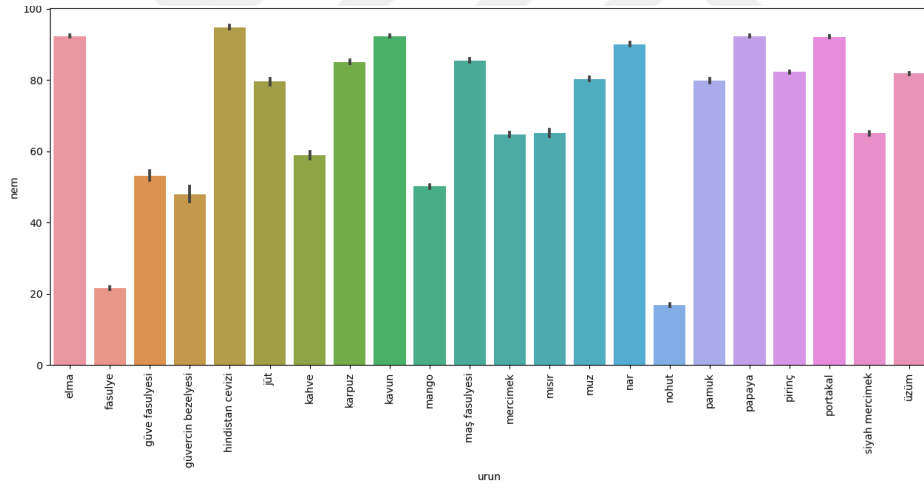
Şekil 4.28. Ürünlerin yetiştirildiği azot değeri ortalamasına göre karşılaştırılması.



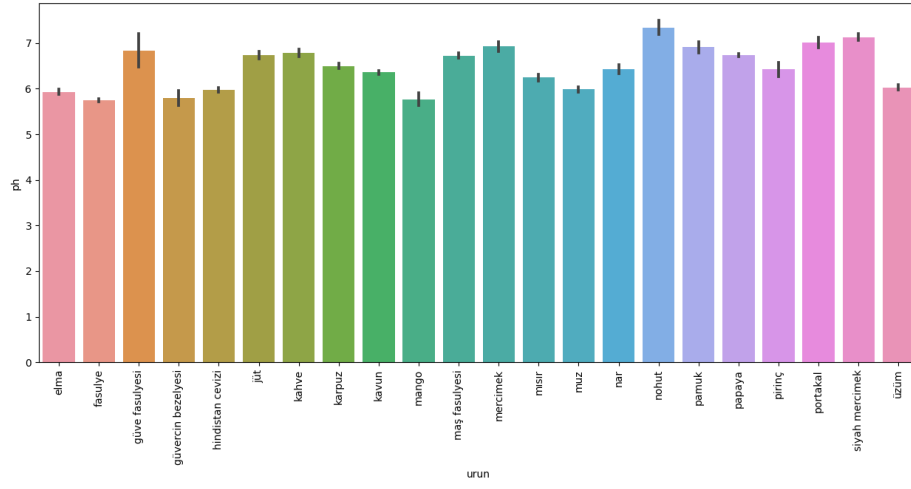
Şekil 4.29. Ürünlerin yetiştirildiği potasyum değeri ortalamasına göre karşılaştırılması.



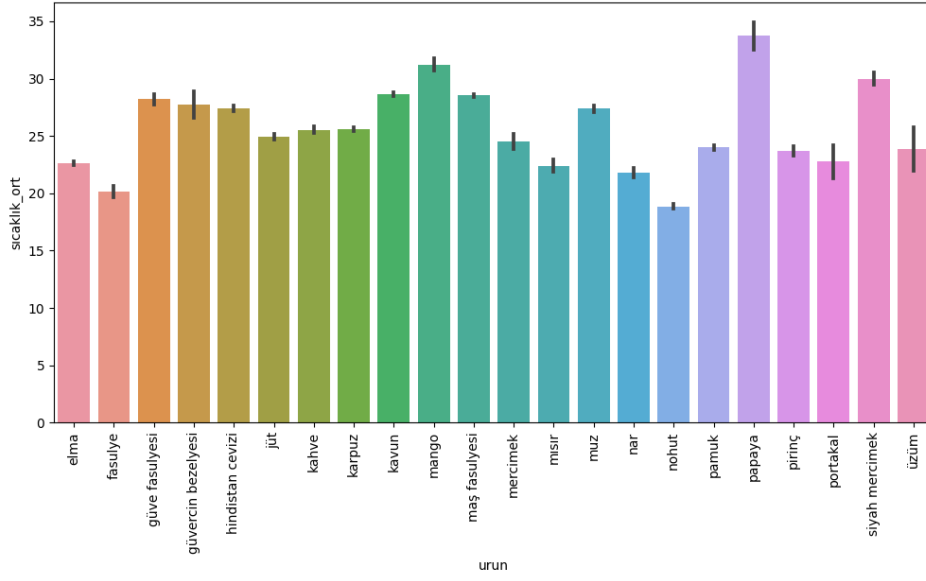
Şekil 4.30. Ürünlerin yetiştirildiği fosfor değeri ortalamasına göre karşılaştırılması.



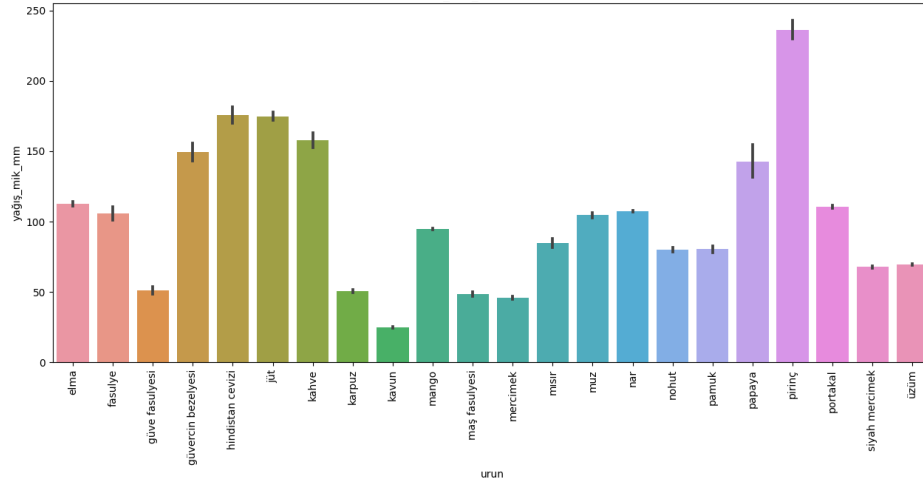
Şekil 4.31. Ürünlerin yetiştirildiği nemlilik (%) değeri ortalamasına göre karşılaştırılması.



Şekil 4.32. Ürünlerin yetiştirildiği ph değeri ortalamasına göre karşılaştırılması.



Şekil 4.33. Ürünlerin yetiştirildiği sıcaklık (°C) değeri ortalamasına göre karşılaştırılması.



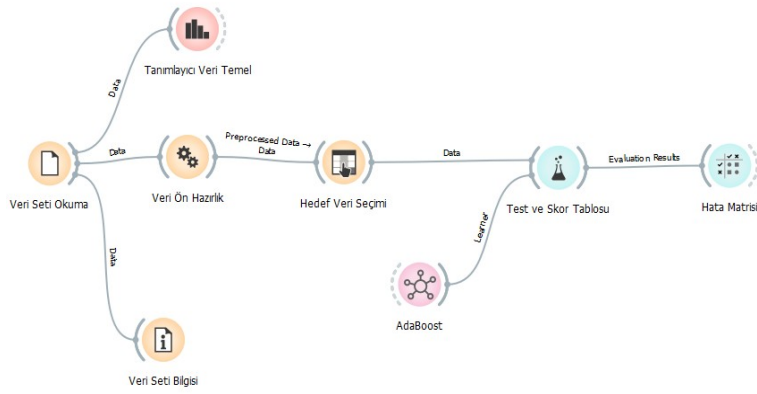
Şekil 4.34. Ürünlerin yetiştirildiği yağış miktarı(mm) değeri ortalamasına göre karşılaştırılması.

4.2. Algoritmaların Test Sonuçları

Çalışmada kullanılan algoritmalar Orange yazılımı kullanılarak eğitim ve test süresi, AUC değeri, doğruluk, F1, kesinlik, hassasiyet, kayıp oranı ve özgüllük değerleri hesaplanmıştır.

4.2.1. Adaboost modeli ve sonuçları

Çalışmada adaboost algoritması Şekil 4.8’de görüldüğü gibi Orange yazılımı ile modellenerek test ve skor verileri alınmıştır.



Şekil 4.35. Orange yazılımı ile tasarlanan adaboost modeli.

Orange yazılımı ile tasarlanan adaboost modeli (Şekil 4.8) çalıştığında eğitim süresi 0.159 sn., test süresi 0.012 sn., AUC değeri 0.995, doğruluk skoru 0.990, F1 skoru 0.990, kesinlik değeri 0.990, hassasiyet değeri 0.990, kayıp oranı 0.345 ve özgüllük 1.000 olarak bulunmuştur. Test ve skor tablosu 10 katlı çapraz doğruluk örneklemesini kullanarak üretilen sonuçlar Çizelge 4.23'te verilmiştir.

Çizelge 4.24. Adaboost modelinin test ve skor sonuçları

Kullanılan Model	Adaboost
Eğitim Süresi (sn)	0.159
Test Süresi (sn)	0.012
AUC	0.995
Doğruluk	0.990
F1 Skoru	0.990
Kesinlik	0.990
Hassasiyet	0.990
Kayıp Oranı	0.345
Özgüllük	1.000

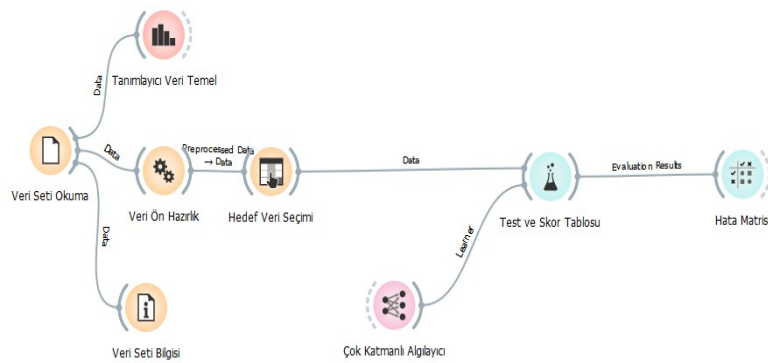
Çizelge 4.25. Adaboost modeli hata matrisi

	Predicted																						Σ
	elma	fasulye	güve fasulyesi	güvercin bezelyesi	hindistan cevizi	jüt	kahve	karpuz	kavun	mango	maz fasulyesi	merc...	muz	mısır	nar	nohut	pamuk	papaya	pirinç	portakal	siyah merc...	üzüm	
elma	100	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100
fasulye	0	100	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100
güve fasulyesi	0	0	96	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	3	0	100
güvercin bezelyesi	0	0	0	100	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100
hindistan cevizi	0	0	0	0	99	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	100
jüt	0	0	0	0	0	91	0	0	0	0	0	0	0	0	0	2	0	7	0	0	0	0	100
kahve	0	0	0	0	0	0	100	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100
karpuz	0	0	0	0	0	0	0	100	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100
kavun	0	0	0	0	0	0	0	0	100	0	0	0	0	0	0	0	0	0	0	0	0	0	100
mango	0	0	0	0	0	0	0	0	0	100	0	0	0	0	0	0	0	0	0	0	0	0	100
maz fasulyesi	0	0	0	0	0	0	0	0	0	0	100	0	0	0	0	0	0	0	0	0	0	0	100
merc...	0	0	1	0	0	0	0	0	0	0	0	99	0	0	0	0	0	0	0	0	0	0	100
muz	0	0	0	0	0	0	0	0	0	0	0	0	100	0	0	0	0	0	0	0	0	0	100
mısır	0	0	0	0	0	0	0	0	0	0	0	0	0	100	0	0	0	0	0	0	0	0	100
nar	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100	0	0	0	0	0	0	0	100
nohut	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100	0	0	0	0	0	0	100
pamuk	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100	0	0	0	0	0	100
papaya	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100	0	0	0	0	100
pirinç	0	0	0	0	0	6	0	0	0	0	0	0	0	0	0	0	0	0	94	0	0	0	100
portakal	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100	0	0	100
siyah merc...	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	99	0	100
üzüm	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100	100
Σ	100	100	97	100	99	97	100	100	100	100	100	100	100	101	100	100	102	100	101	101	102	100	2200

Adaboost modeli hata matrisi incelendiğinde (Çizelge 4.24) veri setinde bulunan güve fasulyesinin bir tanesi mercimek, üç tanesi siyah mercimek olarak, jütün iki tanesi pamuk yedi tanesi pirinç olarak, pirincin altı tanesi jüt olarak, siyah mercimeğin bir tanesi mısır olarak yanlış tahmin edilmiştir. Genel olarak bakıldığında adaboost modelinde pirinç ve jüt arasında yanlış tahmin yoğunluğu tespit edilmiştir.

4.2.2. Çok katmanlı algılayıcı modeli ve sonuçları

Çalışmada çok katmanlı algılayıcı algoritması Şekil 4.9'da görüldüğü gibi Orange yazılımında modellenerek test ve skor verileri alınmıştır.



Şekil 4.36. Orange yazılımı ile tasarlanan çok katmanlı algılayıcı modeli.

Orange yazılımı ile tasarlanan çok katmanlı algılayıcı modeli (Şekil 4.9) çalıştığında eğitim süresi 9.615 sn., test süresi 0.030 sn., AUC değeri 1.000, doğruluk skoru 0.985, F1 skoru 0.985, kesinlik değeri 0.985, hassasiyet değeri 0.985, kayıp oranı 0.169 ve özgüllük 0.999 olarak bulunmuştur. Test ve skor tablosu 10 katlı çapraz doğruluk örnekleme kullanarak üretilen sonuçlar Çizelge 4.25'te verilmiştir.

Çizelge 4.26. Çok katmanlı algılayıcı modelinin test ve skor sonuçları

Kullanılan Model	Çok Katmanlı Algılayıcı
Eğitim Süresi (sn)	9.615
Test Süresi (sn)	0.030
AUC	1.000
Doğruluk	0.985
F1 Skoru	0.985
Kesinlik	0.985
Hassasiyet	0.985
Kayıp Oranı	0.169
Özgüllük	0.999

Çizelge 4.27. Çok katmanlı algılayıcı modeli hata matrisi

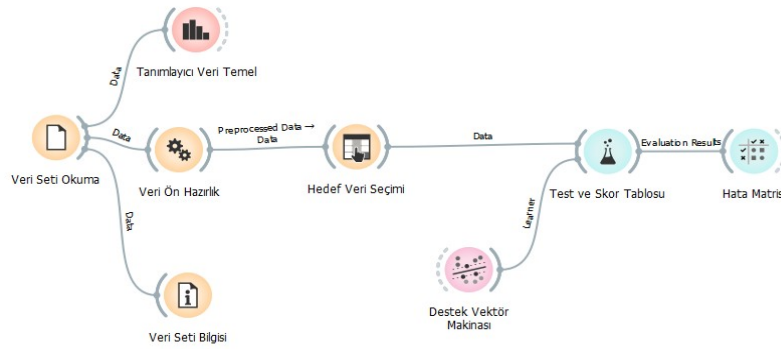
		Predicted																						Σ
		elma	fasulye	güve fasulyesi	güvercin bezelyesi	hindistan cevizi	jüt	kahve	karpuz	kavun	mango	maş fasulyesi	merc...	muz	msır	nar	nohut	pamuk	papaya	pirinç	portakal	siyah merc...	üzüm	Σ
Actual	elma	100	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100
	fasulye	0	96	1	1	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	1	0	100
	güve fasulyesi	0	1	92	0	0	0	0	0	0	0	0	3	0	0	0	0	0	0	0	0	4	0	100
	güvercin bezelyesi	0	0	0	100	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100
	hindistan cevizi	0	0	0	0	100	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100
	jüt	0	0	0	0	0	89	1	0	0	0	0	0	0	0	0	0	0	1	9	0	0	0	100
	kahve	0	0	0	0	0	1	99	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100
	karpuz	0	0	0	0	0	0	0	99	1	0	0	0	0	0	0	0	0	0	0	0	0	0	100
	kavun	0	0	0	0	0	0	0	0	100	0	0	0	0	0	0	0	0	0	0	0	0	0	100
	mango	0	0	0	3	0	0	0	0	0	97	0	0	0	0	0	0	0	0	0	0	0	0	100
	maş fasulyesi	0	0	0	0	0	0	0	0	0	0	100	0	0	0	0	0	0	0	0	0	0	0	100
	merc...	0	0	1	0	0	0	0	0	0	0	0	99	0	0	0	0	0	0	0	0	0	0	100
	muz	0	0	0	0	0	0	0	0	0	0	0	0	100	0	0	0	0	0	0	0	0	0	100
	msır	0	0	1	0	0	0	1	0	0	0	0	0	0	97	0	0	1	0	0	0	0	0	100
	nar	0	0	0	0	4	0	0	0	0	0	0	0	0	0	96	0	0	0	0	0	0	0	100
	nohut	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100	0	0	0	0	0	0	100
	pamuk	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100	0	0	0	0	0	100
	papaya	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	99	0	0	0	0	100
	pirinç	0	0	0	0	0	6	0	0	0	0	0	0	0	0	0	0	0	0	94	0	0	0	100
	portakal	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100	0	0	0	100
	siyah merc...	0	1	1	1	0	0	0	0	0	0	0	0	0	2	0	0	0	0	0	95	0	0	100
	üzüm	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	99	0	100
	Σ	101	98	96	105	104	96	101	99	101	97	100	102	101	100	96	100	101	100	103	100	100	99	2200

Çok katmanlı algılayıcı hata matrisi incelendiğinde (Çizelge 4.26) veri setinde bulunan fasulyenin bir tane güve fasulyesi, bir tane güvercin bezelyesi bir tane mısır, bir

tane siyah mercimek olarak, güve fasulyesinin bir tanesi fasulye, üç tanesi mercimek dört tanesi siyah mercimek olarak, jütün bir tanesi kahve, bir tanesi papaya, dokuz tanesi pirinç olarak, kahvenin bir tanesi jüt olarak, karpuzun bir tanesi kavun olarak, mangonun üç tanesi güvercin bezelyesi olarak, mercimeğin bir tanesi güve fasulyesi olarak, mısırın bir tanesi güve fasulyesi, bir tanesi kahve, bir tanesi pamuk olarak, narın dört tanesi hindistan cevizi olarak, papayanın bir tanesi muz olarak, pirincin altı tanesi jüt olarak, siyah mercimeğin bir tanesi fasulye, bir tanesi güve fasulyesi, bir tanesi güvercin bezelyesi, iki tanesi mısır olarak, üzümün bir tanesi elma olarak yanlış tahmin ettiği tespit edilmiştir. Çok katmanlı algılayıcı modeli birçok farklı üründe yanlış tahminler üretmiştir. En çok yanlış tahmin sayısı güve fasulyesi, jüt, pirinç ve siyah mercimek olmuştur.

4.2.3. Destek vektör makineleri modeli ve sonuçları

Çalışmada destek vektör makineleri algoritması Şekil 4.10'da görüldüğü gibi Orange yazılımında modellenerek test ve skor verileri alınmıştır.



Şekil 4.37. Orange yazılımı ile tasarlanan destek vektör makineleri modeli.

Orange yazılımı ile tasarlanan destek vektör makineleri modeli (Şekil 4.3) çalıştığında eğitim süresi 1.206 sn., test süresi 0.089 sn., AUC değeri 1.000, doğruluk skoru 0.986, F1 skoru 0.986, kesinlik değeri 0.987, hassasiyet değeri 0.986, kayıp oranı 0.136 ve özgülük 0.999 olarak bulunmuştur. Test ve skor tablosu 10 katlı çapraz doğruluk örneklemesini kullanarak üretilen sonuçlar Çizelge 4.27'de verilmiştir.

Çizelge 4.28. Destek vektör makineleri modelinin test ve skor sonuçları

Kullanılan Model	Destek Vektör Algoritmaları
Eğitim Süresi (sn)	1.206
Test Süresi (sn)	0.089
AUC	1.000
Doğruluk	0.986
F1 Skoru	0.986
Kesinlik	0.987
Hassasiyet	0.986
Kayıp Oranı	0.136
Özgüllük	0.999

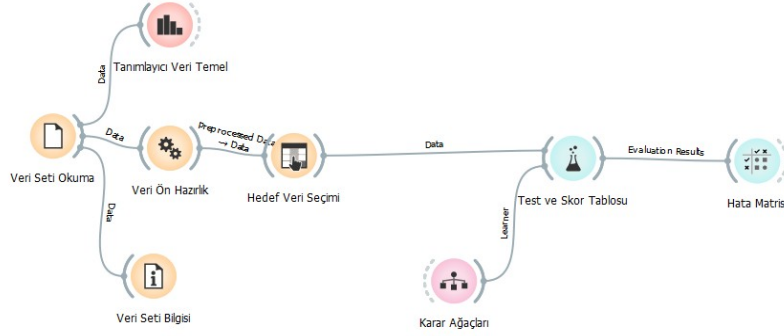
Çizelge 4.29. Destek vektör makineleri modelinin hata matrisi

		Predicted																						Σ
		elma	fasulye	güve fasulyesi	güvercin bezelyesi	hindistan cevizi	jüt	kahve	karpuz	kavun	mango	maş fasulyesi	merc...	muz	msır	nar	nohut	pamuk	papaya	pirinç	portakal	siyah merc...	üzüm	
Actual	elma	100	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100
	fasulye	0	100	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100
	güve fasulyesi	0	0	98	0	0	0	0	0	0	0	2	0	0	0	0	0	0	0	0	0	0	0	100
	güvercin bezelyesi	0	1	0	99	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100
	hindistan cevizi	0	0	0	0	100	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100
	jüt	0	0	0	0	0	96	0	0	0	0	0	0	0	0	0	0	0	0	4	0	0	0	100
	kahve	0	0	0	0	0	2	98	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100
	karpuz	0	0	0	0	0	0	0	100	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100
	kavun	0	0	0	0	0	0	0	0	100	0	0	0	0	0	0	0	0	0	0	0	0	0	100
	mango	0	0	0	0	0	0	0	0	0	100	0	0	0	0	0	0	0	0	0	0	0	0	100
	maş fasulyesi	0	0	0	0	0	0	0	0	0	0	100	0	0	0	0	0	0	0	0	0	0	0	100
	merc...	0	0	2	0	0	0	0	0	0	0	0	97	0	0	0	0	0	0	0	0	1	0	100
	muz	0	0	0	0	0	0	0	0	0	0	0	0	100	0	0	0	0	0	0	0	0	0	100
	msır	0	0	0	0	0	0	0	0	0	0	0	0	0	98	0	0	2	0	0	0	0	0	100
	nar	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100	0	0	0	0	0	0	0	100
	nohut	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100	0	0	0	0	0	0	100
	pamuk	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100	0	0	0	0	0	100
	papaya	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100	0	0	0	0	100
	pirinç	0	0	0	0	0	15	0	0	0	0	0	0	0	0	0	0	0	0	85	0	0	0	100
	portakal	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100	0	0	100
siyah merc...	0	0	0	0	0	0	0	0	0	0	0	2	0	0	0	0	0	0	0	0	98	0	100	
üzüm	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100	100	
Σ	100	101	100	99	100	113	98	100	100	100	100	101	100	98	100	100	102	100	89	100	99	100	2200	

Destek vektör makineleri hata matrisi incelendiğinde (Çizelge 4.28) veri setinde bulunan güve fasulyesinin iki tanesi mercimek olarak, güvercin bezelyesinin bir tanesi fasulye olarak, jütün dört tanesi pirinç olarak, kahvenin iki tanesi jüt olarak, mercimeğin iki tanesi güvercin bezelyesi, bir tanesi siyah mercimek olarak, mısırın iki tanesi pamuk olarak, pirincin 15 tanesi jüt olarak, siyah mercimeğin iki tanesi mercimek olarak yanlış tahmin edilmiştir. Genel olarak bakıldığında destek vektör makineleri modelinde pirinç ve jüt arasında yanlış tahmin yoğunluğu tespit edilmiştir.

4.2.4. Karar ağaçları modeli ve sonuçları

Çalışmada karar ağaçları algoritması Şekil 4.11’de görüldüğü gibi Orange yazılımında modellenerek test ve skor verileri alınmıştır.



Şekil 4.38. Orange yazılımı ile tasarlanan karar ağaçları modeli.

Orange yazılımı ile tasarlanan karar ağaçları modeli (Şekil 4.11) çalıştığında eğitim süresi 0.289 sn., test süresi 0.000 sn., AUC değeri 0.991, doğruluk skoru 0.978, F1 skoru 0.978, kesinlik değeri 0.978, hassasiyet değeri 0.978, kayıp oranı 0.604 ve özgüllük 0.999 olarak bulunmuştur. Test ve skor tablosu 10 katlı çapraz doğruluk örneklemesini kullanarak üretilen sonuçlar Çizelge 4.29’da verilmiştir.

Çizelge 4.30. Karar ağaçları modelinin test ve skor sonuçları

Kullanılan Model	Karar Ağaçları
Eğitim Süresi (sn.)	0.289
Test Süresi (sn.)	0.000
AUC	0.991
Doğruluk	0.978
F1 Skoru	0.978
Kesinlik	0.978
Hassasiyet	0.978
Kayıp Oranı	0.604
Özgüllük	0.999

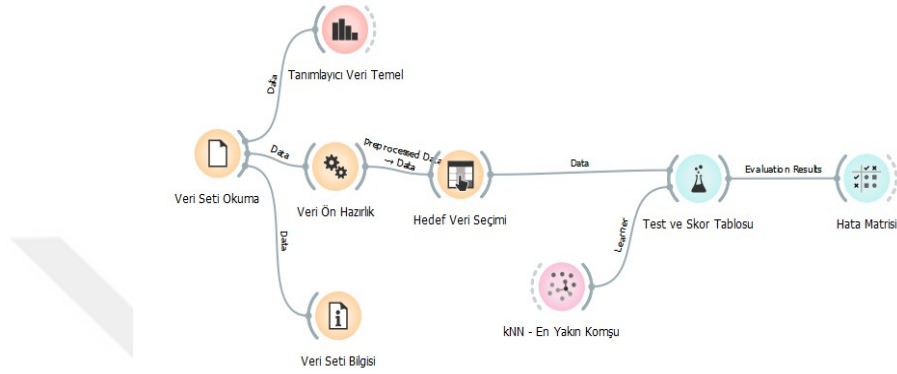
Çizelge 4.31. Karar ağaçları modeli hata matrisi

	Predicted																						Σ
	elma	fasulye	güve fasulyesi	güvercin bezelyesi	hindistan cevizi	jüt	kahve	karpuz	kavun	mango	maş fasulyesi	mercü...	muz	msur	nar	nohut	pamuk	papaya	pirinç	portakal	siyah mercü...	üzüm	
Actual	elma	100	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100
	fasulye	0	99	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100
	güve fasulyesi	0	0	95	2	0	0	0	0	0	0	2	0	0	0	0	0	0	0	0	1	0	100
	güvercin bezelyesi	0	1	0	99	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100
	hindistan cevizi	0	0	0	0	100	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100
	jüt	0	0	0	0	1	92	0	0	0	0	0	0	0	0	0	0	1	6	0	0	0	100
	kahve	0	0	0	0	0	2	98	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100
	karpuz	0	0	0	0	0	0	0	100	0	0	0	0	0	0	0	0	0	0	0	0	0	100
	kavun	0	0	0	0	0	0	0	0	100	0	0	0	0	0	0	0	0	0	0	0	0	100
	mango	0	0	0	0	0	0	0	0	0	100	0	0	0	0	0	0	0	0	0	0	0	100
	maş fasulyesi	0	0	0	0	0	0	0	0	0	0	99	0	0	0	0	0	0	0	0	1	0	100
	mercü...	0	0	3	0	0	0	0	0	0	0	0	96	0	0	0	0	0	0	0	1	0	100
	muz	0	0	0	0	0	0	0	0	0	0	0	0	100	0	0	0	0	0	0	0	0	100
	msur	0	0	0	0	0	0	0	0	0	0	0	0	0	99	0	1	0	0	0	0	0	100
	nar	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100	0	0	0	0	0	0	100
	nohut	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100	0	0	0	0	0	100
	pamuk	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100	0	0	0	0	100
	papaya	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	98	0	0	0	100
	pirinç	0	0	0	0	0	5	0	0	0	0	0	0	0	0	0	0	0	0	95	0	0	100
	portakal	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	99	0	100
	siyah mercü...	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	99	100
	üzüm	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100
Σ	100	100	98	103	102	100	98	100	100	100	99	99	100	99	100	100	101	99	101	99	102	100	2200

Karar ağaçları modeli hata matrisi incelendiğinde (Çizelge 4.30) veri setinde bulunan fasulyenin bir tanesi güvercin bezelyesi olarak, güve fasulyesinin iki tanesi güvercin bezelyesi, iki tanesi mercimek bir tanesi siyah mercimek olarak, güvercin bezelyesinin bir tanesi fasulye olarak, jütün bir tanesi hindistan cevizi, bir tanesi papaya, atı tanesi pirinç olarak, kahvenin iki tanesi jüt olarak, maş fasulyesinin bir tanesi siyah mercimek olarak, mercimeğin üç tanesi güve fasulyesi, bir tanesi siyah mercimek olarak, mısırın bir tanesi pamuk olarak, papayanın bir tanesi güvercin bezelyesi, bir tanesi jüt olarak, pirincin beş tanesi jüt olarak, portakalın bir tanesi hindistan cevizi olarak, siyah mercimeğin bir tanesi mercimek olarak yanlış tahmin edilmiştir. Karar ağaçları modeli birçok farklı üründe yanlış tahminler üretmiştir. Genel olarak bakıldığında karar ağaçları modelinde pirinç ve jüt arasında yanlış tahmin yoğunluğu tespit edilmiştir.

4.2.5. K-en yakın komşu modeli ve sonuçları

Çalışmada k-en yakın komşu algoritması Şekil 4.12’de görüldüğü gibi Orange yazılımında modellenerek test ve skor verileri alınmıştır.



Şekil 4.39. Orange yazılımı ile yapılan k-en yakın komşu modeli.

Orange yazılımı ile tasarlanan k-en yakın komşu modeli (Şekil 4.12) çalıştığında eğitim süresi 0.077 sn., test süresi 0.079 sn., AUC değeri 0.999, doğruluk skoru 0.980, F1 skoru 0.980, kesinlik değeri 0.980, hassasiyet değeri 0.980, kayıp oranı 0.115 ve özgüllük 0.999 olarak bulunmuştur. Test ve skor tablosu 10 katlı çapraz doğruluk örneklemesini kullanarak üretilen sonuçlar Çizelge 4.31’de verilmiştir.

Çizelge 4.32. K-en yakın komşu modeli test ve skor verileri

Kullanılan Model	K-En Yakın Komşu
Eğitim Süresi (sn.)	0.077
Test Süresi (sn.)	0.079
AUC	0.999
Doğruluk	0.980
F1 Skoru	0.980
Kesinlik	0.980
Hassasiyet	0.980
Kayıp Oranı	0.115
Özgüllük	0.999

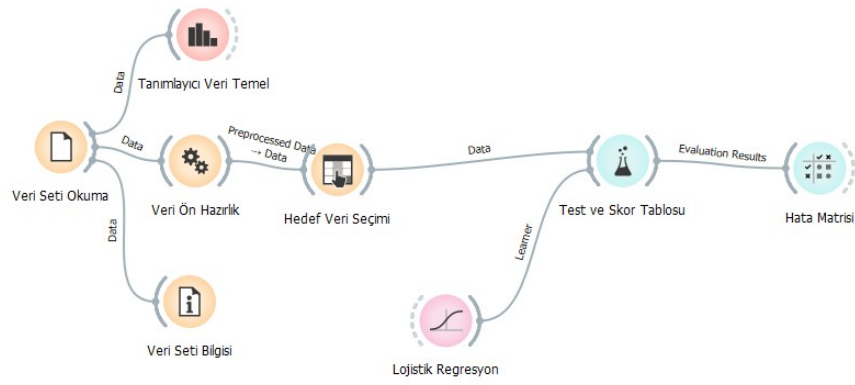
Çizelge 4.33. K-en yakın komşu modeli hata matrisi

	Predicted																							Σ
	elma	fasulye	güve fasulyesi	güvercin bezelyesi	hindistan cevizi	jüt	kahve	karpuz	kavun	mango	maş fasulyesi	mercimek...	muz	mısır	nar	nohut	pamuk	papaya	pirinç	portakal	siyah mercimek...	üzüm		
elma	100	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100	
fasulye	0	100	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100	
güve fasulyesi	0	0	96	0	0	0	0	0	0	0	0	3	0	0	0	0	0	0	0	0	1	0	100	
güvercin bezelyesi	0	6	0	93	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	100	
hindistan cevizi	0	0	0	0	100	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100	
jüt	0	0	0	0	0	90	0	0	0	0	0	0	0	0	0	0	0	0	10	0	0	0	100	
kahve	0	0	0	0	0	1	99	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100	
karpuz	0	0	0	0	0	0	0	100	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100	
kavun	0	0	0	0	0	0	0	0	100	0	0	0	0	0	0	0	0	0	0	0	0	0	100	
mango	0	0	0	0	0	0	0	0	0	100	0	0	0	0	0	0	0	0	0	0	0	0	100	
maş fasulyesi	0	0	0	0	0	0	0	0	0	0	100	0	0	0	0	0	0	0	0	0	0	0	100	
mercimek...	0	0	1	0	0	0	0	0	0	0	0	98	0	0	0	0	0	0	0	0	1	0	100	
muz	0	0	0	0	0	0	0	0	0	0	0	0	100	0	0	0	0	0	0	0	0	0	100	
mısır	0	0	0	0	0	0	0	0	0	0	0	0	0	97	0	0	0	3	0	0	0	0	100	
nar	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100	0	0	0	0	0	0	0	100	
nohut	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100	0	0	0	0	0	0	100	
pamuk	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	99	0	0	0	0	0	100	
papaya	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	99	0	0	0	0	100	
pirinç	0	0	0	0	0	15	0	0	0	0	0	0	0	0	0	0	0	0	85	0	0	0	100	
portakal	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100	0	0	100	
siyah mercimek...	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100	0	100	
üzüm	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100	100	
Σ	100	106	97	93	100	107	99	100	100	100	100	101	100	98	100	100	102	99	95	100	103	100	2200	

K-en yakın komşu modeli hata matrisi incelendiğinde (Çizelge 4.32) veri setinde bulunan güve fasulyesinin üç tanesi mercimek, bir tanesi siyah mercimek olarak, güvercin bezelyesinin altı tanesi fasulye, bir tanesi siyah mercimek olarak, jütün 10 tanesi pirinç olarak, kahvenin bir tanesi jüt olarak, mercimeğin bir tanesi güve fasulyesi, bir tanesi siyah mercimek olarak, mısırın üç tanesi pamuk olarak, pamuğun bir tanesi mısır olarak, papayanın bir tanesi jüt olarak, pirincin 15 tanesi jüt olarak yanlış tahmin edilmiştir. K-en yakın komşu modeli genel olarak bakıldığında pirinç ve jüt arasında ve güvercin bezelyesi ile fasulye arasında yanlış tahmin yoğunluğu tespit edilmiştir.

4.2.6. Lojistik regresyon modeli ve sonuçları

Çalışmada lojistik regresyon algoritması Şekil 4.13’de görüldüğü gibi Orange yazılımında modellenerek test ve skor verileri alınmıştır.



Şekil 4.40. Orange yazılımı ile yapılan k-en yakın komşu modeli.

Orange yazılımı ile tasarlanan lojistik regresyon modeli (Şekil 4.13) çalıştığında eğitim süresi 173.3 sn., test süresi 0.012 sn., AUC değeri 1.000, doğruluk skoru 0.984, F1 skoru 0.984, kesinlik değeri 0.984, hassasiyet değeri 0.984, kayıp oranı 0.048 ve özgüllük 0.999 olarak bulunmuştur. Test ve skor tablosu 10 katlı çapraz doğruluk örnekleme kullanarak üretilen sonuçlar Çizelge 4.33’de verilmiştir.

Çizelge 4.34. Lojistik Regresyon modeli test ve skor verileri

Kullanılan Model	Lojistik Regresyon
Eğitim Süresi (sn.)	173.300
Test Süresi (sn.)	0.012
AUC	1.000
Doğruluk	0.984
F1 Skoru	0.984
Kesinlik	0.984
Hassasiyet	0.984
Kayıp Oranı	0.048
Özgüllük	0.999

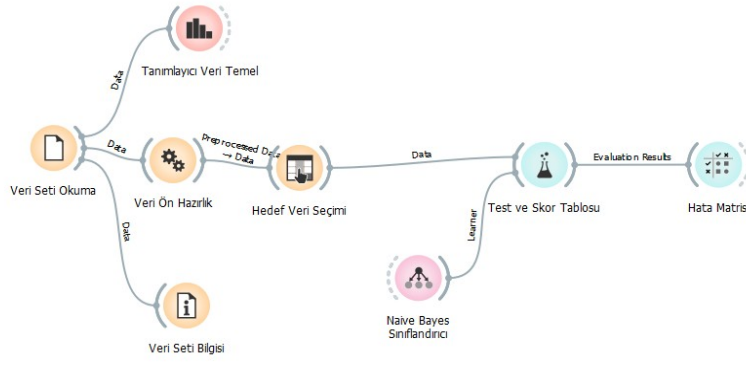
Çizelge 4.35. Lojistik regresyon modeli hata matrisi

	Predicted																						Σ
	elma	fasulye	güve fasulyesi	güvercin bezelyesi	hindistan cevizi	jüt	kahve	karpuz	kavun	mango	maş fasulyesi	merc...	muz	mısır	nar	nohut	pamuk	papaya	pirinç	portakal	siyah merc...	üzüm	
elma	100	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100
fasulye	0	100	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100
güve fasulyesi	0	0	98	0	0	0	0	0	0	0	0	2	0	0	0	0	0	0	0	0	0	0	100
güvercin bezelyesi	0	0	2	0	98	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100
hindistan cevizi	0	0	0	0	100	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100
jüt	0	0	0	0	0	92	1	0	0	0	0	0	0	0	0	0	0	1	6	0	0	0	100
kahve	0	0	0	0	0	1	99	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100
karpuz	0	0	0	0	0	0	0	100	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100
kavun	0	0	0	0	0	0	0	0	100	0	0	0	0	0	0	0	0	0	0	0	0	0	100
mango	0	0	0	0	0	0	0	0	0	100	0	0	0	0	0	0	0	0	0	0	0	0	100
maş fasulyesi	0	0	0	0	0	0	0	0	0	0	100	0	0	0	0	0	0	0	0	0	0	0	100
merc...	0	0	1	0	0	0	0	0	0	0	0	99	0	0	0	0	0	0	0	0	0	0	100
muz	0	0	0	0	0	0	0	0	0	0	0	0	100	0	0	0	0	0	0	0	0	0	100
mısır	0	0	0	0	0	0	0	0	0	0	0	0	0	97	0	0	2	0	0	0	1	0	100
nar	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100	0	0	0	0	0	0	0	100
nohut	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100	0	0	0	0	0	0	100
pamuk	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100	0	0	0	0	0	100
papaya	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	96	1	0	1	1	100
pirinç	0	0	0	0	0	11	0	0	0	0	0	0	0	0	0	0	0	0	89	0	0	0	100
portakal	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100	0	0	0	100
siyah merc...	0	0	0	0	0	0	0	0	0	0	0	2	0	1	0	0	0	0	0	0	97	0	100
üzüm	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100	100
Σ	100	102	99	98	100	104	100	101	100	100	100	103	100	98	100	100	102	97	96	100	99	101	2200

Lojistik regresyon modeli hata matrisi incelendiğinde (Çizelge 4.34) veri setinde bulunan güvercin bezelyesinin iki tanesini fasulye olarak, kahvenin bir tanesini jüt olarak, güve fasulyesinin iki tanesini mercimek olarak, jütün bir tanesini kahve, bir tanesini papaya altı tanesini pirinç olarak, mercimeğin bir tanesini güve fasulyesi olarak, mısırın iki tanesini pamuk bir tanesini siyah mercimek olarak, papayanın birer tane karpuz, pirinç siyah mercimek ve üzüm olarak, pirincin 11 tanesini jüt olarak, siyah mercimeğin iki tanesini mercimek bir tanesini mısır olarak tahmin edilmiştir. Lojistik regresyon modelinde en çok yanlış tahmin sayısı jüt ve pirinç arasında olmuştur.

4.2.7. Naive bayes sınıflandırıcı modeli ve sonuçları

Çalışmada naive bayes sınıflandırıcı algoritması Şekil 4.14'te görüldüğü gibi Orange yazılımında modellenerek test ve skor verileri alınmıştır.



Şekil 4.41. Orange yazılımı ile tasarlanan naive bayes sınıflandırıcı modeli.

Orange yazılımı ile tasarlanan naive bayes sınıflandırıcı modeli (Şekil 4.14) çalıştığında eğitim süresi 0.063 sn., test süresi 0.013 sn., AUC değeri 0.999, doğruluk skoru 0.943, F1 skoru 0.943, kesinlik değeri 0.945, hassasiyet değeri 0.943, kayıp oranı 0.146 ve özgüllük 0.997 olarak bulunmuştur. Test ve skor tablosu 10 katlı çapraz doğruluk örneklemesini kullanarak üretilen sonuçlar Çizelge 4.35’te verilmiştir.

Çizelge 4.36. Naive bayes sınıflandırıcı modeli test ve skor verileri

Kullanılan Model	Naive Bayes
Eğitim Süresi (sn.)	0.063
Test Süresi (sn.)	0.013
AUC	0.999
Doğruluk	0.943
F1 Skoru	0.943
Kesinlik	0.945
Hassasiyet	0.943
Kayıp Oranı	0.146
Özgüllük	0.997

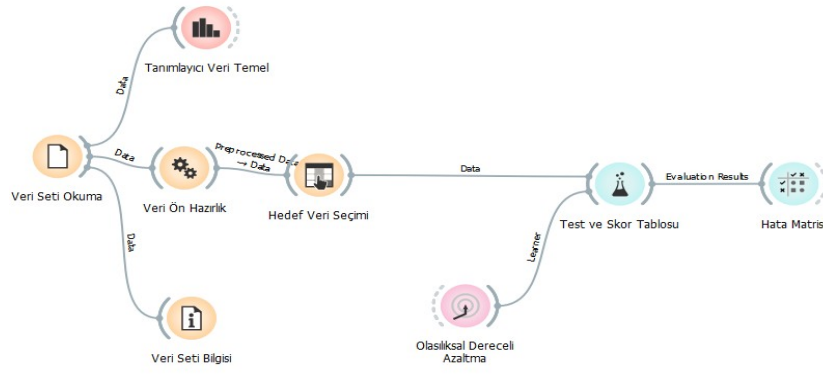
Çizelge 4.37. Naive bayes modeli hata matrisi

	Predicted																						Σ
	elma	fasulye	güve fasulyesi	güvercin bezelyesi	hindistan cevizi	jüt	kahve	karpuz	kavun	mango	maş fasulyesi	merc...	muz	mısır	nar	nohut	pamuk	papaya	pirinç	portakal	siyah merc...	üzüm	
elma	100	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100
fasulye	0	97	1	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100
güve fasulyesi	0	0	93	0	0	0	0	0	0	1	0	0	0	1	0	0	0	0	0	0	5	0	100
güvercin bezelyesi	0	20	1	77	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	1	0	100
hindistan cevizi	0	0	0	0	100	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100
jüt	0	0	0	0	0	83	0	0	0	0	0	0	0	0	0	0	0	0	17	0	0	0	100
kahve	0	0	0	0	0	6	93	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	100
karpuz	0	0	0	0	0	0	0	99	1	0	0	0	0	0	0	0	0	0	0	0	0	0	100
kavun	0	0	0	0	0	0	0	0	100	0	0	0	0	0	0	0	0	0	0	0	0	0	100
mango	0	0	1	0	0	0	1	0	0	98	0	0	0	0	0	0	0	0	0	0	0	0	100
maş fasulyesi	0	0	1	0	0	0	0	0	0	0	98	1	0	0	0	0	0	0	0	0	0	0	100
merc...	0	0	1	0	0	0	0	0	0	0	0	94	0	0	0	0	0	0	0	0	5	0	100
muz	0	0	0	0	0	0	0	0	0	0	0	0	100	0	0	0	0	0	0	0	0	0	100
mısır	0	1	1	0	0	0	1	0	0	0	0	2	0	84	0	0	11	0	0	0	0	0	100
nar	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100	0	0	0	0	0	0	0	100
nohut	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100	0	0	0	0	0	0	100
pamuk	0	0	0	0	0	0	0	0	0	0	0	0	0	7	0	0	93	0	0	0	0	0	100
papaya	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100	0	0	0	0	100
pirinç	0	0	0	0	0	29	0	0	0	0	0	0	0	0	0	0	0	0	71	0	0	0	100
portakal	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100	0	0	100
siyah merc...	0	0	0	1	0	0	0	0	0	0	0	5	0	0	0	0	0	0	0	0	94	0	100
üzüm	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100	100
Σ	100	118	99	80	100	118	95	99	101	99	98	102	100	94	100	100	104	100	88	100	105	100	2200

Naive bayes modeli hata matrisi incelendiğinde (Çizelge 4.36) veri setinde bulunan fasulyenin bir tanesi güve fasulyesi, iki tanesi güvercin bezelyesi olarak, güve fasulyesinin bir tanesi mango, bir tanesi mısır, beş tane siyah mercimek olarak, güvercin bezelyesinin 20 tanesi fasulye, bir tanesi güve fasulyesi, bir tanesi mısır, bir tanesi siyah mercimek olarak, jütün 17 tanesi pirinç olarak, kahvenin altı tanesi jüt, bir tanesi mısır olarak, karpuzun bir tanesi kavun olarak, mangonun bir tanesi güve fasulyesi, bir tanesi kahve olarak, maş fasulyesinin bir tanesi güve fasulyesi, bir tanesi mercimek olarak, mercimeğin bir tanesi güve fasulyesi, beş tanesi siyah mercimek olarak, mısırın 11 tanesi pamuk, iki tanesi mercimek, bir tanesi fasulye, bir tanesi güve fasulyesi, bir tanesi kahve olarak, nohutun yedi tanesi mısır olarak, pirincin 29 tanesi jüt olarak, siyah mercimeğin bir tanesi güvercin bezelyesi, beş tanesi mercimek olarak yanlış tahmin edilmiştir. Naive bayes modeli birçok farklı üründe yanlış tahminler üretmiştir. Genel olarak bakıldığında naive bayes modelinde pirinç ve jüt arasında, güvercin bezelyesi ile fasulye arasında, pamuk ile mısır arasında yanlış tahmin yoğunluğu tespit edilmiştir.

4.2.8. Olasılıksal dereceli azalma modeli ve sonuçları

Çalışmada olasılıksal dereceli azalma algoritması Şekil 4.15’de görüldüğü gibi Orange yazılımında modellenerek test ve skor verileri alınmıştır.



Şekil 4.42. Orange yazılımı ile tasarlanan olasılıksal dereceli azalma modeli.

Orange yazılımı ile tasarlanan olasılıksal dereceli azalma modeli (Şekil 4.15) çalıştığında eğitim süresi 0.582 sn., test süresi 0.015 sn., AUC değeri 0.953, doğruluk skoru 0.910, F1 skoru 0.908, kesinlik değeri 0.916, hassasiyet değeri 0.910, kayıp oranı 3.124 ve özgüllük 0.996 olarak bulunmuştur. Test ve skor tablosu 10 katlı çapraz doğruluk örneklemesini kullanarak üretilen sonuçlar Çizelge 4.37’de verilmiştir.

Çizelge 4.38. Olasılıksal dereceli azalma modeli test ve skor verileri

Kullanılan Model	Olasılıksal Dereceli Azalma
Eğitim Süresi (sn.)	0.582
Test Süresi (sn.)	0.015
AUC	0.953
Doğruluk	0.910
F1 Skoru	0.908
Kesinlik	0.916
Hassasiyet	0.910
Kayıp Oranı	3.124
Özgüllük	0.996

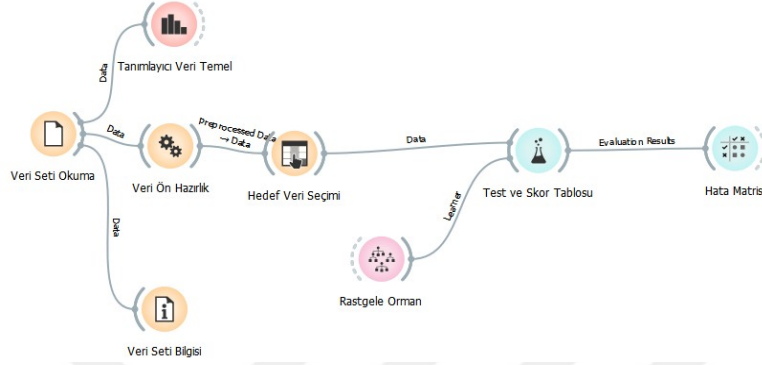
Çizelge 4.39. Olasılıksal dereceli azalma modeli hata matrisi

	Predicted																						Σ
	elma	fasulye	güve fasulyesi	güvercin bezelyesi	hindistan cevizi	jüt	kahve	karpuz	kavun	mango	maş fasulyesi	merc...	muz	mısır	nar	nohut	pamuk	papaya	pirinç	portakal	siyah merc...	üzüm	
Actual	elma	100	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100
	fasulye	0	100	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100
	güve fasulyesi	0	0	87	3	0	1	0	0	0	4	1	1	0	3	0	0	0	0	0	0	0	100
	güvercin bezelyesi	0	8	0	89	0	2	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	100
	hindistan cevizi	0	0	0	0	100	0	0	0	0	0	0	0	0	0	0	100	0	0	0	0	0	100
	jüt	0	0	0	0	0	61	4	0	0	0	0	0	0	0	0	0	0	5	30	0	0	100
	kahve	0	0	0	0	0	0	100	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100
	karpuz	0	0	0	0	0	2	1	79	15	0	0	0	0	1	0	0	2	0	0	0	0	100
	kavun	0	0	0	0	0	0	0	0	100	0	0	0	0	0	0	0	0	0	0	0	0	100
	mango	0	0	0	0	0	0	0	0	0	100	0	0	0	0	0	0	0	0	0	0	0	100
	maş fasulyesi	0	0	0	0	0	0	0	0	0	0	100	0	0	0	0	0	0	0	0	0	0	100
	merc...	0	0	5	0	0	0	0	0	0	0	1	93	0	0	0	0	0	0	0	0	1	100
	muz	0	0	0	0	0	0	0	0	0	0	0	0	100	0	0	0	0	0	0	0	0	100
	mısır	0	1	0	0	0	9	2	1	0	0	0	0	2	65	0	0	20	0	0	0	0	100
	nar	0	0	0	0	10	6	0	0	0	0	0	0	0	0	84	0	0	0	0	0	0	100
	nohut	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100	0	0	0	0	0	100
	pamuk	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	98	0	0	0	0	100
	papaya	0	0	0	0	0	3	0	0	0	1	0	8	0	0	0	0	0	80	8	0	0	100
	pirinç	0	0	0	0	0	4	0	0	0	0	0	0	0	0	0	0	0	0	96	0	0	100
	portakal	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	99	0	100
siyah merc...	0	0	2	1	0	5	0	0	0	0	0	16	0	2	0	0	0	0	0	0	74	0	100
üzüm	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100	
Σ	100	109	94	93	110	93	107	80	116	104	110	110	103	73	84	100	120	86	134	99	75	100	2200

Olasılıksal dereceli azalma modeli hata matrisi incelendiğinde (Çizelge 4.38) veri setinde bulunan güve fasulyesinin üç tanesi güvercin bezelyesi, bir tanesi jüt, dört tanesi mango, bir tanesi maş fasulyesi, bir tanesi mercimek, üç tanesi mısır olarak, güvercin bezelyesinin sekiz tanesi fasulye, iki tanesi jüt, bir tanesi mısır olarak, jütün dört tanesi mısır, beş tanesi papaya, 30 tanesi pirinç olarak, karpuzun 15 tanesi kavun, iki tanesi jüt bir tanesi kahve, iki tanesi pamuk, bir tanesi mısır olarak, mercimeğin beş tanesi güve bezelyesi, bir tanesi maş fasulyesi, bir tanesi siyah mercimek olarak, mısırın bir tanesi güve fasulyesi, dokuz tanesi jüt, iki tanesi kahve, bir tanesi karpuz, iki tanesi muz olarak, narın 10 tanesi hindistan cevizi, altı tanesi jüt olarak, pamuğun bir tanesi muz, bir tanesi mısır olarak, üç tanesi jüt, bir tanesi kavun, sekiz tanesi maş fasulyesi, sekiz tanesi pirinç olarak, pirincin dört tanesi jüt olarak, portakalın bir tanesi papaya olarak, siyah mercimeğin iki tanesi güve fasulyesi, bir tanesi güvercin bezelyesi, beş tanesi jüt, 16 tanesi mercimek, iki tanesi mısır olarak yanlış tahmin edilmiştir. Olasılıksal dereceli azalma modeli birçok farklı üründe yanlış tahminler üretmiştir. Genel olarak bakıldığında olasıksal dereceli azalma modelinde pirinç ve jüt arasında, karpuz ile kavun arasında, pamuk, mısır ve jüt arasında, siyah mercimek ile mercimek arasında yanlış tahmin yoğunluğu tespit edilmiştir.

4.2.9. Rastgele orman modeli ve sonuçları

Çalışmada rastgele orman algoritması Şekil 4.16’da görüldüğü gibi Orange yazılımında modellenerek test ve skor verileri alınmıştır.



Şekil 4.43. Orange yazılımı ile tasarlanan rastgele orman modeli.

Orange yazılımı ile tasarlanan rastgele orman modeli (Şekil 4.16) çalıştığında eğitim süresi 0.317 sn., test süresi 0.032 sn., AUC değeri 1.000, doğruluk skoru 0.995, F1 skoru 0.995, kesinlik değeri 0.995, hassasiyet değeri 0.995, kayıp oranı 0.052 ve özgüllük 1.000 olarak bulunmuştur. Test ve skor tablosu 10 katlı çapraz doğruluk örneklemesini kullanarak üretilen sonuçlar Çizelge 4.39’da verilmiştir.

Çizelge 4.40. Rastgele orman modeli test ve skor verileri

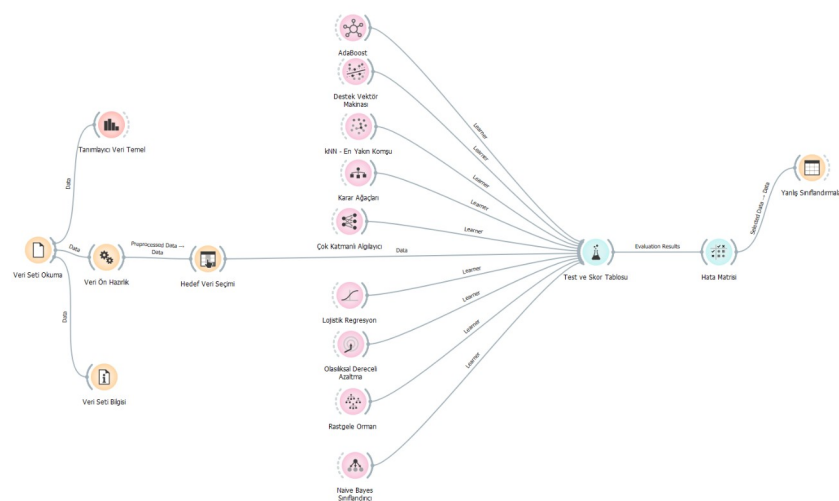
Kullanılan Model	Rastgele Orman
Eğitim Süresi (sn.)	0.317
Test Süresi (sn.)	0.032
AUC	1.000
Doğruluk	0.995
F1 Skoru	0.995
Kesinlik	0.995
Hassasiyet	0.995
Kayıp Oranı	0.052
Özgüllük	1.000

Çizelge 4.41. Rastgele orman modeli hata matrisi

	Predicted																						Σ
	elma	fasulye	güve fasulyesi	güvercin bezelyesi	hindistan cevizi	jüt	kahve	karpuz	kavun	mango	maş fasulyesi	merc...	muz	msır	nar	nohut	pamuk	papaya	pirinç	portakal	siyah merc...	üzüm	
Actual	elma	100	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100
	fasulye	0	100	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100
	güve fasulyesi	0	0	98	1	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	100
	güvercin bezelyesi	0	0	0	100	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100
	hindistan cevizi	0	0	0	0	100	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100
	jüt	0	0	0	0	0	97	0	0	0	0	0	0	0	0	0	0	0	3	0	0	0	100
	kahve	0	0	0	0	0	0	100	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100
	karpuz	0	0	0	0	0	0	0	100	0	0	0	0	0	0	0	0	0	0	0	0	0	100
	kavun	0	0	0	0	0	0	0	0	100	0	0	0	0	0	0	0	0	0	0	0	0	100
	mango	0	0	0	0	0	0	0	0	0	100	0	0	0	0	0	0	0	0	0	0	0	100
	maş fasulyesi	0	0	0	0	0	0	0	0	0	0	100	0	0	0	0	0	0	0	0	0	0	100
	merc...	0	0	2	0	0	0	0	0	0	0	0	98	0	0	0	0	0	0	0	0	0	100
	muz	0	0	0	0	0	0	0	0	0	0	0	0	100	0	0	0	0	0	0	0	0	100
	msır	0	0	0	0	0	0	0	0	0	0	0	0	98	0	0	2	0	0	0	0	0	100
	nar	0	0	0	0	0	0	0	0	0	0	0	0	0	100	0	0	0	0	0	0	0	100
	nohut	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100	0	0	0	0	0	0	100
	pamuk	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100	0	0	0	0	0	100
	papaya	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100	0	0	0	0	100
	pirinç	0	0	0	0	0	5	0	0	0	0	0	0	0	0	0	0	0	95	0	0	0	100
	portakal	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100	0	0	100
	siyah merc...	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100	0	100
	üzüm	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100	100
Σ	100	100	100	101	100	102	100	100	100	100	100	99	100	98	100	100	102	100	98	100	100	100	2200

Rastgele orman modeli hata matrisi incelendiğinde (Çizelge 4.40) veri setinde bulunan güve fasulyesinin bir tanesi güvercin bezelyesi, bir tanesi mercimek olarak, jütün üç tanesi pirinç olarak, mercimeğin iki tanesi güve fasulyesi olarak, mısırın iki tanesi pamuk olarak, pirincin beş tanesi jüt olarak yanlış tahmin ettiği tespit edilmiştir. Rastgele orman modelinde en çok yanlış tahmin sayısı jüt ve pirinç arasında olmuştur.

4.3. Tüm Modellerin Toplu Test Sonuçları



Şekil 4.44. Kullanılan makine öğrenmesi modellerinin test edilmesi.

Çalışmada kullanılan makine öğrenmesi algoritmalarının karşılaştırılması için Şekil 4.17'deki gibi tek bir test ve skor aracına bağlanması gerekmektedir. Bu sayede bir liste halinde modellerin eğitim ve test süreleri, AUC değeri, doğruluk, F1 skoru, kesinlik hassasiyet, kayıp oranı ve özgüllük değerleri hesaplanacaktır. Hesaplama sonucunda değerler Çizelge 4.41'de gösterilmiştir.

Çizelge 4.42. Kullanılan modellerin doğruluk sırasına göre test ve skor tablosu

Kullanılan Model	Eğitim Süresi (sn.)	Test Süresi(sn.)	AUC	Doğruluk	F1 skoru	Kesinlik	Hassasiyet	Kayıp Oranı	Özgüllük
Rastgele Orman	0.317	0.032	1.000	0.995	0.995	0.995	0.995	0.052	1.000
Adaboost	0.159	0.012	0.995	0.990	0.990	0.990	0.990	0.345	1.000
Destek Vektör Mak.	1.206	0.089	1.000	0.986	0.986	0.987	0.986	0.136	0.999
Çok Katmanlı Alg.	9.615	0.030	1.000	0.985	0.985	0.985	0.985	0.169	0.999
Lojistik Regresyon	173.300	0.012	1.000	0.984	0.984	0.984	0.984	0.048	0.999
K-En Yakın Komşu	0.077	0.079	0.999	0.980	0.980	0.980	0.980	0.115	0.999
Karar Ağaçları	0.289	0.000	0.991	0.978	0.978	0.978	0.978	0.604	0.999
Naive Bayes	0.063	0.013	0.999	0.943	0.943	0.945	0.943	0.146	0.997
O. Dereceli Azalma	0.582	0.015	0.953	0.910	0.908	0.916	0.910	3.124	0.996

Bu çalışmada kullanılan makine öğrenmesi modellerinden doğruluk oranı en yüksekten en düşüğe doğru Çizelge 4.42'deki gibi tespit edilmiştir.

Çizelge 4.43. Bütün modellerin doğruluk skoruna doğruluk oranı

Model	Doğruluk Oranı
Rastgele Orman	%99.5
Adaboost	%99.0
Destek Vektör Makineleri	%98.6
Çok Katmanlı Algılayıcı	%98.5
Lojistik Regresyon	%98.4
K-En Yakın Komşu	%98.0
Karar Ağaçları	%97.8
Naive Bayes Sınıflandırıcı	%94.3
Olasılıksal Dereceli Azalma	%91.0

Bu çalışmada kullanılan makine öğrenmesi modellerinin eğitim ve test süreleri toplanarak en hızlı çalışan model Çizelge 4.43'de 0.076 sn. ile naive bayes modeli olmuştur.

Çizelge 4.44. Modellerin eğitim ve test sürelerinin toplamı

Kullanılan Algoritma	Eğitim Süresi(sn.)	Test Süresi(sn.)	Toplam Süre(sn.)
Rastgele Orman	0.317	0.032	0.349
Adaboost Algoritması	0.159	0.012	0.171
Destek Vektör Makineleri	1.206	0.089	1.295
Çok Katmanlı Algılayıcı	9.615	0.030	9.645
Lojistik Regresyon	173.300	0.012	173.312
K-En Yakın Komşu	0.077	0.079	0.156
Karar Ağaçları	0.289	0.000	0.289
Naive Bayes	0.063	0.013	0.076
Olasılıksal Dereceli Azalma	0.582	0.015	0.597

4.4. Model Dosyası Oluşturulması ve Tahmin Ara Yüzü

Çalışmada en iyi tahminleyici model rastgele orman olarak tespit edildikten sonra bunu kullanıcıların deneyimleyebilmesi için python ile modelleyip bir ara yüze entegre edilmesi gerekmektedir. Bunun için gerekli basamaklar şu şekilde sıralanabilir:

1. Çalışmada kullanılacak temel python kütüphaneleri (pandas, numpy) projeye eklenir.
2. Çalışmada kullanılacak model algoritmaları sklearn kütüphanesinden çağırılır.
3. Çalışmada kullanılacak veri seti projeye eklenir. Veri seti bilgisi df.shape , df.columns, df.info() komutlarıyla kontrol sağlanır.
4. Çalışmada kullanılacak veri setinde tekrarlı veri olup olmadığı kontrol edilir. Tekrarlı veri bulunursa bunu silmesi istenilir.
5. Çalışmada kullanılacak veri seti eğitim ve test verisi olarak iki parçaya ayrılır. Çalışmada eğitim verisi %70, test verisi %30 olacak şekilde bölünmüştür.
6. Sklearn kütüphanesinden hangi algoritma fonksiyonu kullanılacaksa projeye dahil edilir. Çalışmada rastgele orman algoritması `ensembles.append(('RF' , RandomForestClassifier()))` komutu ile çağırılmıştır.

7. Veri setinden sıra dışı veriler temizlenir.

8. Model `save_model(model, 'RF_model.pkl')` komutu ile kaydedilmiştir.

Model oluşturma işlemi tamamlandıktan sonra projede kullanılacak .pkl dosyası bir ara yüze bağlanması gerekmektedir. Bunun için çok kullanışlı bir python kütüphanesi olan streamlit kullanılmıştır. Tasarlamak için şu adımlar izlenmiştir:

1. Çalışmada kullanılacak kütüphaneler (streamlit, pandas, numpy, pickle) projeye eklenmiştir.
2. Model dosyası `RF.model.pkl` projeye dahil edilmiştir.
3. Çalışmanın ara yüz tasarımı yapılır. Kodlama kısmı için streamlit kütüphanesinin dokümantasyonundan yararlanılmıştır.
4. Ara yüz tasarlanırken `st.number_input` fonksiyonu ile değerlerin girileceği bölümler oluşturulmuştur.
5. Tahmin et butonuna tıklandığında formdaki verileri model ile karşılaştırıp sonuç üreten `prediction = loaded_model.predict(single_pred)` fonksiyonu çalıştırılmıştır.
6. Sonuçlar ekrana `prediction.item().title()` fonksiyonu ile yazdırılmıştır.

Çalışma tamamlandığında veri setinden rastgele ürünler denenmiş olup doğru sonuçlar ürettiği teyit edilmiştir. Şekil 4.18'de tasarlanmış olan ürün tavsiye sisteminin bir ekran görüntüsü bulunmaktadır.

Ürün Tavsiye Sistemi

Tarlanızda yetiştirilecek en iyi ürünü tespit etmek ister misiniz? 

Azot

1

-

+

Sodyum

1

-

+

Potasyum

1

-

+

Sıcaklık Ortalaması

0,00

-

+

Nem ortalaması (%)

0,00

-

+

Ph

0,00

-

+

Yağış Ortalaması (mm)

0,00

-

+

Tahmin Et!

Bilgi

Mahsul tavsiyesi, hassas tarımın en önemli yönlerinden biridir. Mahsul önerileri bir dizi faktöre dayanmaktadır. Hassas tarım, ürün seçimi sorunlarını ele almak için bu kriterleri site bazında tanımlamayı amaçlamaktadır. "Sahaya özgü" metodoloji performansı iyileştirmiş olsa da, yine de sistemlerin sonuçlarını izleme ihtiyacı vardır. Hassas tarım sistemlerinin hepsi eşit yaratılmamıştır. Bununla birlikte, tarımda, hatalar önemli maddi ve sermaye kaybına neden olabileceğinden, yapılan tavsiyelerin doğru ve kesin olması çok önemlidir.

Nasıl Çalışır ?

Tüm parametreleri tamamladığınızda makine öğrenimi modeli, çeşitli parametrelere dayalı olarak belirli bir çiftlikte büyümeye en uygun mahsulleri tahmin edecektir.

Şekil 4.45. Ürün tavsiye sistemi ara yüzü.



5. TARTIŞMA ve SONUÇLAR

Bu çalışmada veri seti üzerinde destek vektör makinesi algoritması, olasılıksal dereceli azalma algoritması, çok katmanlı algılayıcı algoritması, adaboost algoritması, rastgele orman algoritması, k-en yakın komşu algoritması, lojistik regresyon algoritması naive bayes algoritması ve karar ağaçları algoritması olarak dokuz farklı makine öğrenmesi algoritması modellenip eğitim ve test süresi, AUC değeri, doğruluk skoru, F1 skoru, kesinlik değeri, hassasiyet değeri, kayıp oranı ve özgüllük değerleri hesaplanmıştır.

Kullanılan algoritmalar arasında doğruluk skoru olarak en yüksek model rastgele orman modeli %99.5, F1 skoru olarak rastgele orman modeli %99.5, AUC değeri olarak rastgele orman modeli, destek vektör makineleri modeli ve çok katmanlı algılayıcı modeli 1.000, hassasiyet olarak rastgele orman %99.5, özgüllük değeri olarak rastgele orman modeli ve adaboost modeli 1.000, kayıp oranı olarak en az skor lojistik regresyon modeli 0.048, kesinlik değeri olarak rastgele orman modeli %99.5 en başarılı model olarak tespit edilmiştir. Eğitim ve test süresi naive bayes modeli 0.076 sn. ile en hızlı tamamlayan model olarak tespit edilmiştir.

ROC eğrisi sınıflayıcı modelin bütün olası değerler üzerinde performansını özetlemek için kullanılan bir grafikdir. AUC değeri, bir parametrenin iki sınıf arasında ne kadar iyi ayırt edilebileceğinin bir ölçüsüdür. Çalışmada rastgele orman modeli 1.000, destek vektör makinesi modeli 1.000, lojistik regresyon modeli 1.000, çok katmanlı algılayıcı modeli 1.000, k-en yakın komşu modeli 0.999, naive bayes modeli 0.999, adaboost modeli 0.995, karar ağaçları modeli 0.991 ve olasılıksal dereceli azalma modeli 0.953 olarak AUC değerleri tespit edilmiştir. Bu değerler karşılaştırıldığında rastgele orman modeli, destek vektör makinesi modeli ve çok katmanlı algılayıcı modeli 1.000 skoru ile en başarılı modeller olarak tespit edilmiştir.

Doğruluk skoru çalışmada kullanılan makine öğrenmesi algoritmalarından (Çizelge 4.42) veri seti için %99.5 doğruluk oranı ile rastgele orman algoritması tespit edilmiştir. İkinci sırada ise %99.0 oran ile adaboost modeli bulunmaktadır. Sırasıyla destek vektör makinaları %98.6, çok katmanlı algılayıcı %98.5, lojistik regresyon modeli %98.4, k-en yakın komşu modeli %98.0, karar ağaçları modeli %97.8, naive

bayes modeli %94.3 ve son olarak olasılıksal dereceli azalma modeli %91.0 ile doğruluk oranları hesaplanmıştır.

Kesinlik, modelin ne kadar doğru olarak tahmin ürettiğinin bir ölçüsüdür. 0 ile 1 arasında değer alan kesinlik ne kadar yüksek olursa model o kadar doğru tahminler üretmiştir denilebilir. Çalışmada (Çizelge 4.41) rastgele orman modeli 0.995, adaboost modeli 0.990, destek vektör makinesi modeli 0.987, çok katmanlı algılayıcı modeli 0.985, lojistik regresyon modeli 0,984, k-en yakın komşu modeli 0.980, karar ağaçları modeli 0.978, naive bayes modeli 0.945 ve olasılıksal dereceli azalma modeli 0.916 değerinde kesinlik sonuçları elde edilmiştir. Kesinlik skoru olarak rastgele orman modeli 0.995 değerle diğer algoritmaların önüne geçmektedir.

Hassasiyet sınıflayıcı modelin ne kadar doğru pozitif değeri doğru tahmin ettiğinin bir ölçüsüdür. 0 ile 1 arasında değer alan gerçek pozitif değerlerin oranı ne kadar yüksek olursa modelin o kadar doğru tahminler üretmiştir denilebilir. Çalışmada (Çizelge 4.41) rastgele orman modeli 0.995, adaboost modeli 0.990, destek vektör makinesi modeli 0.986, çok katmanlı algılayıcı modeli 0.985, lojistik regresyon modeli 0,984, k-en yakın komşu modeli 0.980, karar ağaçları modeli 0.978, naive bayes modeli 0.943 ve olasılıksal dereceli azalma modeli 0.910 değerinde hassasiyet hesaplanmıştır. Hassasiyet olarak rastgele orman modeli 0.995 değeri ile en iyi model olarak tespit edilmiştir.

Kayıp oranı gerçek değeri 1 olmasına karşın 0 olarak tahmin edilenlerin yanlış negatif değerlerin oranıdır. Mümkün olduğunca küçük olması istenilmektedir. Çalışmada (Çizelge 4.41) lojistik regresyon modeli 0,048, rastgele orman modeli 0.052, k-en yakın komşu modeli 0.115, destek vektör makinesi modeli 0.136, naive bayes modeli 0.146, çok katmanlı algılayıcı modeli 0.169, adaboost modeli 0.345, karar ağaçları modeli 0.604 ve olasılıksal dereceli azalma modeli 3.124 değerinde kayıp oranı hesaplanmıştır. Kayıp oranı konusunda en başarılı model 0.048 değerle lojistik regresyon modeli olarak tespit edilmiştir.

Özgüllük modelin ne kadar doğru negatif değeri doğru tahmin ettiğinin bir ölçüsüdür. Çalışmada (Çizelge 4.41) rastgele orman modeli 1.000, adaboost modeli 1.000, lojistik regresyon modeli 0.999, destek vektör makinesi modeli 0.999, çok katmanlı algılayıcı modeli 0.999, k-en yakın komşu modeli 0.999, karar ağaçları modeli 0.999, naive bayes modeli 0.997 ve olasılıksal dereceli azalma modeli 0.996 değerinde

özgüllük tespit edilmiştir. Özgüllük olarak rastgele orman modeli ve adaboost modeli en iyi skoru elde etmiştir.

Çalışmada kullanılan modeller hız bakımından toplam süreleri (Çizelge 4.43) kıyas edildiğinde küçükten büyüğe sırasıyla naive bayes modeli 0.076 sn., k-en yakın komşu modeli 0.156 sn., adaboost modeli 0.171 sn., karar ağaçları modeli 0,289 sn., rastgele orman modeli 0.349 sn., olasılıksal dereceli azalma modeli 0.597 sn., destek vektör makinesi modeli 1.295 sn., çok katmanlı algılayıcı modeli 9.645 sn. ve lojistik regresyon modeli 173.312 sn. olarak tespit edilmiştir. Orange yazılımında modellenen lojistik regresyon modeli kullanılan veri seti için diğer bütün algoritmalarından daha yavaş çalışmaktadır. Tasarlanan modeller genel olarak hız bakımından bir saniyeden daha kısa sürede veri setinin eğitim ve test işlemini sonuçlandırmıştır. Bu işlemin süresi kullanılan bilgisayarın işlemcisinin gücüne göre değişiklik gösterebilmektedir.

Çalışmada kullanılan hata matrislerinin hepsinin sonuçları incelendiğinde ortak olarak jüt ve pirinç sıklıkla birbirine karıştırılmaktadır. Kullanılan modellerden adaboost modeli, destek vektör makineleri modeli, lojistik regresyon modeli, k-en yakın komşu modeli ve rastgele orman modellerinin hatalı tahminleri pirinç ve jüt üzerinde yoğunlaşmaktadır. Bu modeller için veri setinde daha fazla örneklem olması halinde daha doğru sonuçlar üretebilir. Çok katmanlı algılayıcı modeli, karar ağaçları modeli, naive bayes modeli ve olasılıksal dereceli azalma modellerinde ise hatalar dağınıktır ve veri setini geliştirip modeller ile daha iyi sonuçlar elde etmek oldukça güçtür.

Çalışma kapsamında kullanılan veri seti ile en doğru tahminleme %99.5 F1 skoru ile rastgele orman modeli ile yapılacağı tespit edilmiştir. Rastgele orman modeli için python dilinde makine öğrenmesi modeli oluşturulmuştur. Streamlit kütüphanesi ile tahminleme ara yüzü tasarımı tamamlanarak kullanıcı deneyimine açık hale getirilmiştir.

Mevcut veri seti üzerinde tahminleme için kullanılabilecek onlarca makine öğrenmesi modeli kullanılabilmektedir. Farklı veri setleri kullanılan makine öğrenmesi modelleri doğruluk skorları farklılık göstermektedir. Bu modellerin hangisinin en doğru sonucu vereceği test edilerek bulunabilir. Bu test süreci Orange, Knime, Weka veya Rapidminer gibi yazılımlar ile hızlı ve güvenilir sonuçları hangi model sağlıyorsa o model seçilerek çalışmaya yön vermek doğru olacaktır. Eğer araştırmacı python dilinde bir model belirleyip üzerinde parametreler ile çalışıp en iyi modeli tespit etme sürecinde

vakit ve maliyet kaybı yaşayacağını düşünüyorsa Orange gibi ücretsiz bir yazılım ile test ve seçim sürecini hızlandırabilir. Çalışmada Orange yazılımının kullanılması çalışmaya bu yönüyle özgünlük kazandırılmıştır.

Kullanılan destek vektör makinesi modeli, olasılıksal dereceli azalma modeli, çok katmanlı algılayıcı modeli, adaboost modeli, rastgele orman modeli, k-en yakın komşu modeli, naive bayes modeli ve karar ağaçları modeli Orange yazılımı ile 10 katlı çapraz doğruluk yöntemi ile test edilmiştir. Bütün algoritmalar için hata matrisleri incelenerek daha sonraki çalışmalarda veri setini hazırlarken hangi ürünler hakkında karmaşıklık olduğu açıklanmıştır. Yetiştirme şartları çok benzeyen ürünlerde çift tahmin sistemi geliştirilmesi mümkündür. Örneğin sadece jüt veya pirinç gibi bir sonuç yerine pirinç ve jütü beraber tavsiye edebilir.

Çalışmada kullanılan veri setindeki ürün sayısı çoğaltılarak Türkiye şartlarında yetiştirilebilecek bütün tarım ürünleri ile rastgele orman algoritmasının eğitilmesi sonucunda çiftçilerin bir web ara yüzü üzerinden toprağının değerlerini girerek yetiştirebileceği ürünler hakkında tavsiye veren bir tavsiye sistemi yapay zekâ modeli olarak bu çalışma kullanılabilir. Buna benzer olarak mevsimsel nitelikleri de veri setine ayrı bir sütun olarak eklenip bir yılda üreticinin üretebileceği birden fazla tarım ürünleri tavsiyesi alınabilir.

Çalışma kullanılarak sadece veri seti, ara yüz ve model değiştirilerek farklı çalışmalar yapılabilir. Hayvan yetiştiriciliği, meyve yetiştiriciliği, mantar yetiştiriciliği için gerekli şartların bulunduğu veriler toplanılarak bu yönde tavsiye veren sistemler tasarlanması mümkündür. Konu hakkında literatür çalışması yapılmış olup ülkemizde bu konuda daha önce yapılmış tavsiye sistemi çalışmasına rastlanmamıştır ve bu yönüyle de çalışma özgünlük kazanmaktadır.

KAYNAKLAR

- Alpaydin, E. 2014. *Introduction to Machine Learning*. MIT Press Chambridge.
- Anonim, 2017. Karışıklık Matrisi (Confusion Matrix), <https://veribilimcisi.com/2017/07/18/karisiklik-matrisi-nedir/>, Erişim tarihi: 23.11. 2020.
- Anonim, 2020a. Çapraz doğrulama (istatistik). [https://tr.wikipedia.org/wiki/%C3%87apraz_do%C4%9Frulama_\(istatistik\)](https://tr.wikipedia.org/wiki/%C3%87apraz_do%C4%9Frulama_(istatistik)), Erişim Tarihi: 23.11.2020.
- Anonim, 2020b. “Harvestify”, Github, https://github.com/Gladiator07/Harvestify/blob/master/Data-processed/crop_recommendation.csv, Erişim Tarihi: 08.08.2020.
- Anonim, 2021. Makine Öğrenmesi Algoritmaları | Microsoft Azure, <https://azure.microsoft.com/tr-tr/overview/machine-learning-algorithms/>. Erişim tarihi: 03.03.2021.
- Archer K. J. 2008. Emprical Characterization Of Random Forest Variable Importance Measure, Computational Statistical Data Analysis, *Computational Statistics & Data Analysis*, 52(4): 2249-2260.
- Ataş, M. 2011. *Hyperspectral Imaging And Machine Learning Of Texture Foods For Classification*, Ph.D. Thesis, Middle East Technical University, Department of Computer Engineering and Computer Science and Control, Ankara.
- Ay, Ş. 2019. Ensemble Learning—Bagging ve Boosting, <https://medium.com/deep-learning-turkiye/ensemble-learning-bagging-ve-boosting-50643428b22b>, Erişim Tarihi: 12.09.2020.
- Bağ, M. 2019. *Derin öğrenme kullanarak ip üzerinden ses hizmeti veren şebekelerde sahtekarlığa yönelik çağrılarının tespiti*, Yüksek Lisans Tezi, Elektrik-Elektronik Mühendisliği, Ankara
- Balta, A. 2018. *Makine öğrenmesi teknikleri ile ekolojik verilerin değerlendirilmesi*, Yüksek Lisans Tezi, Fırat Üniversitesi, Fen Bilimleri Enstitüsü, Elâzığ.
- Bilgili, L. 2018. *Gemi yaşam döngüsünde operasyonel gaz emisyonlarının makine öğrenmesi yöntemiyle tahmini*, Doktora Tezi, Yıldız Teknik Üniversitesi, Fen Bilimleri Enstitüsü, İstanbul.
- Bottou, L. 2010. Large-scale machine learning with stochastic gradient descent. *In Proceedings of COMPSTAT'2010*, Physica-Verlag HD, 177-186.
- Breiman, L. 2001. Random forests,machine learning, 2001 *Kluwer Academic Publishers*, 45(1): 5-32.
- Breiman, L., Cutler A. 2005. Random forest, http://www.stat.berkeley.edu/~breiman/RandomForests/cc_home.htm, Erişim Tarihi: 12.07.2011.
- Breiman, L. 2002. Manual on setting up, using, and understanding random forests V3.1, http://oz.berkeley.edu/users/breiman/Using_random_forests_V3.1.pdf, Erişim Tarihi: 20.09.2011.
- Burgos-Artizzu, X. P., Ribeiro, A., Tellaeche, A., Pajares, G., Fernández-Quintanilla, C. 2010. Analysis Of Natural Images Processing For The Extraction Of Agricultural Elements. *Image and Vision Computing*, 28(1): 138-149.
- Foody, G. M., Mathur, A. 2004. Toward intelligent training of supervised image classifications: directing training data acquisition for svm classification. *Remote Sensing of Environment*, 93: 107–117.
- Garip, E. 2017. *OECD ülkelerindeki karbondioksit (co2) emisyonunun makine öğrenmesi ile tahmin edilmesi*, Yüksek Lisans Tezi, İstanbul Medeniyet Üniversitesi, Fen Bilimleri Enstitüsü, İstanbul.

- Güneş, A. 2017. *Hiperspektral görüntüleme ve makine öğrenmesi teknikleri ile küflü kuru incirlerin tahribatsız olarak tespiti*, Doktora Tezi, Süleyman Demirel Üniversitesi, Fen Bilimleri Enstitüsü, Isparta.
- Heckerman, D. 1999. A Tutorial on Learning with Bayesian Networks. In Jordan, M.I. (ed.), *Adaptive Computation and Machine Learning*. MIT Press, 301–354.
- Horning, N. 2010. *Randomforests: an algorithm for image classification and generation of continuous fields data sets*, International Conference on Geoinformatics for Spatial Infrastructure Development in Earth and Allied Sciences (GISIDEAS) Hanoi, Vietnam
- Kara, F. 2012. *Arazi toplulaştırma çalışmalarında yapay zekâ uygulamalarının kullanım olanakları: Amasya ili Gümüşhacıköy ilçesi, Keçiköy beldesi uygulaması*, Yüksek Lisans Tezi, Ondokuz Mayıs Üniversitesi, Fen Bilimleri Enstitüsü, Samsun.
- Kavzoglu, T., Colkesen, I. 2009. A kernel functions analysis for support vector machines for land cover classification. *International Journal of Applied Earth Observation and Geoinformation*, **11**(5): 352–359.
- Martínez-Martínez, V., Gomez-Gil, F. J., Gomez-Gil, J., Ruiz-Gonzalez, R. 2015. An Artificial Neural Network based expert system fitted with Genetic Algorithms for detecting the status of several rotary components in agro-industrial machines using a single vibration signal. *Expert Systems with Applications*, **42**(17-18): 6433-6441.
- Özdamar, K. 1997. *Paket Programlar ile İstatistiksel Veri Analizi*, Anadolu Üniversitesi Yayınları, Eskişehir. 512.
- Öztürk, M. 2018. *Makine öğrenmesi ve görüntü işleme tekniklerini kullanarak drone ile yaprak sınıflandırma*, Yüksek Lisans Tezi, İstanbul Teknik Üniversitesi, Fen Bilimleri Enstitüsü, İstanbul.
- Partel, V., Kakarla, S. C., Ampatzidis, Y. 2019. Development and evaluation of a low-cost and smart technology for precision weed management utilizing artificial intelligence. *Computers and electronics in agriculture*, **157**: 339-350.
- Robindro, K., Sarma, S. K. 2013. JESS based expert system architecture for diagnosis of rice plant diseases: Design and prototype development. In 2013 4th International Conference on Intelligent Systems, *Modelling and Simulation*, 674-676.
- Romeo, J., Pajares, G., Montalvo, M., Guerrero, J. M., Guijarro, M., Cruz, J. M. 2013. A new Expert System for greenness identification in agricultural images. *Expert Systems with Applications*, **40**(6): 2275-2286.
- Sabzi, S., Abbaspour-Gilandeh, Y. 2018. Using video processing to classify potato plant and three types of weed using hybrid of artificial neural network and particle swarm algorithm. *Measurement*, **126**: 22-36.
- Singh, V., Misra, A. K. 2017. Detection of plant leaf diseases using image segmentation and soft computing techniques. *Information Processing In Agriculture*, **4**(1): 41-49.
- Sungur, C., Altun, A. 2010. Konya Bölgesindeki Don Olaylarına Karşı Mistleme Sisteminin Yapay Sinir Ağları ile Modellenmesi. *Selçuk Tarım Bilimleri Dergisi*, **24**(4): 70-75.
- Tatlidil, H. 1996. *Uygulamalı Çok Değişkenli İstatistiksel Analiz*, Cem Web Ofset Ltd. Şti., Ankara. 424.

- Vapnik, V. 2013. *The nature of statistical learning theory*. Springer science & business media.
- Yakupoğlu, T., Özdemir, N., Ekberli, İ. 2008. Toprak Erozyonu Çalışmalarında Bulanık Mantık Uygulamaları. *Anadolu Tarım Bilimleri Dergisi*, **23**(2): 121-130.
- Yigit, E., Sabancı, K., Toktas, A., Kayabasi, A. 2019. A Study On Visual Features Of Leaves In Plant Identification Using Artificial Intelligence Techniques. *Computers and electronics in agriculture*, **156**: 369-377.





EKLER

Ek 1. Kullanıcı ara yüzü tasarım kodları app.py dosyası

```
import streamlit as st , pandas as pd , numpy as np , os, pickle, warnings
#st.beta_set_page_config(page_title="Crop Recommender", page_icon="", layout='centered', initial_sidebar_state="collapsed")
def load_model(modelfile):
    loaded_model = pickle.load(open(modelfile, 'rb'))
    return loaded_model
def main():
    # title
    html_temp = """
    <div>
    <h1 style="color:MEDIUMSEAGREEN;text-align:left;"> Ürün Tavsiye Sistemi </h1>
    </div>
    st.markdown(html_temp, unsafe_allow_html=True)
    col1,col2 = st.beta_columns([2,2])
    i    with col2:        with st.beta_expander("  Bilgi", expanded=True):
        st.write("""  Ürün tavsiyesi..  """)
        ## Nasıl Çalışır
        Tüm parametreleri tamamladığınızda makine öğrenimi modeli, çeşitli parametrelere dayalı olarak belirli bir çiftlikte büyümeye en uygun
        mahsulleri tahmin edecektir.""
    with col1:
        st.subheader(" Tarlanızda yetiştirilecek en iyi ürünü tespit etmek ister misiniz? ")
        N = st.number_input("Azot", 1,10000)
        P = st.number_input("Sodyum", 1,10000)
        K = st.number_input("Potasyum", 1,10000)
        temp = st.number_input("Sıcaklık Ortalaması",0.0,100000.0)
        humidity = st.number_input("Nem ortalaması (%)", 0.0,100000.0)
        ph = st.number_input("Ph", 0.0,100000.0)
        rainfall = st.number_input("Yağış Ortalaması (mm)",0.0,100000.0)
        feature_list = [N, P, K, temp, humidity, ph, rainfall]
        single_pred = np.array(feature_list).reshape(1,-1)
        if st.button("Tahmin Et!"):
            loaded_model = load_model('RF_model.pkl')
            prediction = loaded_model.predict(single_pred)
            col1.write("""
            ## Sonuçlar
            """)
            col1.success(f" Sizin için önerilen ürün: {prediction.item().title()} ")
    #code for html
    st.warning("Not: Bu yapay zeka ile ürün tavsiye etme sistemi halen geliştirme aşamasındadır.")
    hide_menu_style = """
    <style>  #MainMenu {visibility: hidden;}  </style>
    hide_menu_style = """  <style>  #MainMenu {visibility: hidden;}  </style>
    st.markdown(hide_menu_style, unsafe_allow_html=True)
    if __name__ == '__main__':    main()
```

Ek 2. Model dosyası oluşturulması için kullanılan model.py dosyası kodları

```

import pandas as pd, import pandas_profiling as pp, import numpy as np, import seaborn as sns,import matplotlib.pyplot as plt, import warnings
import os, import plotly.graph_objects as go, import plotly.io as pio
import pickle
from sklearn.utils import resample
# Metrikler --- Metrics
from sklearn.metrics import accuracy_score, classification_report, confusion_matrix, auc, roc_curve
# Doğrulama -- Validation
from sklearn.model_selection import train_test_split, cross_val_score, KFold
from sklearn.pipeline import Pipeline, make_pipeline
# Kollektif Öğrenme Algoritmaları --- Ensembles
from sklearn.ensemble import RandomForestClassifier
warnings.filterwarnings('ignore')
sns.set_style("whitegrid", {'axes.grid': False})
pio.templates.default = "plotly_white"
#Eğitim ve Test verisine ayırma ---- Split Data to Training and Validation set #
def read_in_and_split_data(data, target):
    X = data.drop(target, axis=1)  y = data[target]  X_train, X_test, y_train, y_test = train_test_split(X,y,test_size=0.3, random_state=0)
    return X_train, X_test, y_train, y_test
# Algoritmala kontrolleri ---- Spot-Check Algorithms #
def ensemblemodels():
    ensembles = []
    ensembles.append(('Rastgele Orman' , RandomForestClassifier()))
    return ensembles
# Modelin Eğitimi ---- Train Model #
def fit_model(X_train, y_train,models):
    # Test options and evaluation metric
    num_folds = 10  scoring = 'accuracy'  results = []  names = []  for name, model in models:  kfold = KFold(n_splits=num_folds, shuffle=True,
random_state=0)  cv_results = cross_val_score(model, X_train, y_train, cv=kfold, scoring=scoring)
    results.append(cv_results)  names.append(name)  msg = "%s: %f (%f)" % (name, cv_results.mean(), cv_results.std())
    print(msg)  return names, results
# Modelin kaydedilmesi ---- Save Trained Model #
def save_model(model,filename):  pickle.dump(model, open(filename, 'wb'))
# Veri setinin yüklenmesi ---- Load Dataset
df = pd.read_excel('veri_seti.xls')
# Sıradışı verilerin temizlenmesi -- Remove Outliers
Q1 = df.quantile(0.25)  Q3 = df.quantile(0.75)  IQR = Q3 - Q1  df_out = df[~((df < (Q1 - 1.5 * IQR)) |(df > (Q3 + 1.5 * IQR)))).any(axis=1)]
# Veri setinin eğitim ve test verilerine bölünmesi --- Split Data to Training and Validation set
target = 'ürün'
X_train, X_test, y_train, y_test = read_in_and_split_data(df, target)
# Train model
pipeline = make_pipeline(StandardScaler(), RandomForestClassifier())
model = pipeline.fit(X_train, y_train)
y_pred = model.predict(X_test)
classification_metrics(pipeline, conf_matrix)
# save model
save_model(model, 'RF_model2.pkl')

```

ÖZ GEÇMİŞ

İlk ve orta öğrenimini 50. Yıl İlköğretim Okulu'nda, lise öğrenimini Hulusi Uçaçelik Anadolu Lisesi'nde (2005-2008) İzmir'in Ödemiş ilçesinde tamamladı. Lisans eğitimini Pamukkale Üniversitesinde Teknik Eğitim Fakültesi Bilgisayar Sistemleri Öğretmenliğinde 2009-2012 Yılları arasında tamamladı. 2019 yılında Van Yüzüncü Yıl Üniversitesi Ziraat Fakültesinde Zootekni Anabilim Dalı'nda yüksek lisans öğrenimine başladı. 2014 yılından itibaren Van'ın İpekyolu ilçesinde öğretmen olarak görev yapmaktadır. Evli ve bir çocuk babasıdır.





T.C
VAN YÜZÜNCÜ YIL ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ
LİSANSÜSTÜ TEZ ORJİNALLİK RAPORU

Tarih:17/06/2021

Tez Başlığı / Konusu:

Tarımda Makine Öğrenmesi Algoritmalarının Uygulamaları

Yukarıda başlığı/konusu belirlenen tez çalışmamın Kapak sayfası, Giriş, Ana bölümler ve Sonuç bölümlerinden oluşan toplam 45 sayfalık kısmına ilişkin, 17/06/2021 tarihinde şahsım/tez danışmanım tarafından Turnitin intihal tespit programından aşağıda belirtilen filtreleme uygulanarak alınmış olan orijinallik raporuna göre, tezin benzerlik oranı % 10 (on) dur.

Uygulanan filtreler aşağıda verilmiştir:

- Kabul ve onay sayfası hariç,
- Teşekkür hariç,
- İçindekiler hariç,
- Simge ve kısaltmalar hariç,
- Gereç ve yöntemler hariç,
- Kaynakça hariç,
- Alıntılar hariç,
- Tezden çıkan yayınlar hariç,
- 7 kelimedenden daha az örtüşme içeren metin kısımları hariç (Limit inatch size to 7 words)

Van Yüzüncü Yıl Üniversitesi Lisansüstü Tez Orijinallik Raporu Alınması ve Kullanılmasına İlişkin Yönergeyi inceledim ve bu yönergede belirtilen azami benzerlik oranlarına göre tez çalışmamın herhangi bir intihal içermediğini; aksinin tespit edileceği muhtemel durumda doğabilecek her türlü hukuki sorumluluğu kabul ettiğimi ve yukarıda vermiş olduğum bilgilerin doğru olduğunu beyan ederim.

Gereğini bilgilerinize arz ederim.

Tarih ve İmza

Adı Soyadı: Samet ERİŞ

Öğrenci No:18910001219

Anabilim Dalı: Zootečni Anabilim Dalı

Programı:

Statüsü: Y. Lisans ☐

Doktora ☐

DANIŞMAN ONAYI
UYGUNDUR

ENSTİTÜ ONAYI
UYGUNDUR

Prof. Dr. Abdullah YEŞİLOVA
(Unvan, Ad Soyad, İmza)

(Unvan, Ad Soyad, İmza)