Elle Yazılmış Rakamların Kalem Bazlı Tanınmasının Sınıflandırılması

Ege Arda Öztürk
Bilgisayar Mühendisliği Bölümü
Başkent Üniversitesi
Ankara, Türkiye
egeardozturk@gmail.com

***Abstract*—Sınıflandırma algoritmaları, makine öğrenmesi ve veri madenciliği gibi çalışma alanlarının önemli unsurlarındandır. Sınıflandırma işlemleri için kullanılan en yaygın algoritmalar arasında olan KNN, SVM ve Rastgele Karar Ormanları algoritmalarının Elle Yazılmış Rakamları Kalem Bazlı Tanıma (Pen-based Recognition of Handwritten Digits) isimli veri kümesini sınıflandırmak için kullanılması, birbirleri ile farklılıklarının bulunması, bu farklılıkların sebeplerinin tespitinin yapılması ve ilgili veri kümesiyle benzer özellikte olan veri kümeleri ile çalışan kişilerin ya da organizasyonların bilgilendirilmesi bu makale kapsamında gerçekleştirilmiştir.**

Anahtar Kelimeler—sınıflandırma, elle yazılmış rakam verisi, SVM , Rastgele Karar Ormanları, KNN

I. GİRİŞ

Bu makale, UCI Machine Learning Repository'ye ait olan, Elle Yazılmış Rakamları Kalem Bazlı Tanıma (Pen-based Recognition of Handwritten Digits) isimli veri kümesinin sınıflandırılması ve bu sınıflandırma sonucu oluşan model yardımı ile, doğruluk oranı yüksek olacak şekilde ilgili veri kümesine dair tahminler yapılması işlemlerini kapsayacak şekilde yazılmıştır. Veri kümesi, 44 farklı kişiden, WACOM PL-100V isimli, LCD ekran entegre edilmiş ve basınca duyarlı tablet ile 250 farklı sayı örneği çizmesi istenerek ve bu çizimler toplanarak hazırlanmıştır. Bu verilerden 30 kişiye ait olanlar training (eğitim) için kullanılmakta, diğer 14 kişiye ait olan veriler bağımsız test için kullanılmaktadır. Veri kümesinde toplam 10992 veri bulunmaktadır ve eksik veri yoktur, veri kümesinin nitelik (attribute) sayısı 17 olarak hesaplanmış bulunmaktadır. Bu niteliklerden 16 tanesi, çizilen rakamların 8 ayrı bölümünden alınmış koordinatların (x,y) değerleri iken, son nitelik veri setinin sınıf niteliğidir (class attribute) ve [0-9] aralığındaki rakamları temsil etmektedir. [1].

Veri Kümesi Nitelikleri

1-Digit (sınıf niteliği)

2.X1 – Elle çizilmiş rakamlardan seçilen ilk koordinatın x-eksenindeki kesim noktası değeri

3.Y1 – Elle çizilmiş rakamlardan seçilen ilk koordinatın y-eksenindeki kesim noktası değeri

4.X2 – Elle çizilmiş rakamlardan seçilen ikinci koordinatın x-eksenindeki kesim noktası değeri

5.Y2 – Elle çizilmiş rakamlardan seçilen ikinci koordinatın y-eksenindeki kesim noktası değeri

6.X3 – Elle çizilmiş rakamlardan seçilen üçüncü koordinatın x-eksenindeki kesim noktası değeri

7.Y3 – Elle çizilmiş rakamlardan seçilen üçüncü koordinatın y-eksenindeki kesim noktası değeri

8.X4 – Elle çizilmiş rakamlardan seçilen dördüncü koordinatın x-eksenindeki kesim noktası değeri

9.Y4 – Elle çizilmiş rakamlardan seçilen dördüncü koordinatın y-eksenindeki kesim noktası değeri

10.X5 – Elle çizilmiş rakamlardan seçilen beşinci koordinatın x-eksenindeki kesim noktası değeri

11.Y5 – Elle çizilmiş rakamlardan seçilen beşinci koordinatın y-eksenindeki kesim noktası değeri

12.X6 – Elle çizilmiş rakamlardan seçilen altıncı koordinatın x-eksenindeki kesim noktası değeri

13.Y6 – Elle çizilmiş rakamlardan seçilen altıncı koordinatın y-eksenindeki kesim noktası değeri

14.X7 – Elle çizilmiş rakamlardan seçilen yedinci koordinatın x-eksenindeki kesim noktası değeri

15.Y7 – Elle çizilmiş rakamlardan seçilen yedinci koordinatın y-eksenindeki kesim noktası değeri

16.X8 – Elle çizilmiş rakamlardan seçilen sekizinci koordinatın x-eksenindeki kesim noktası değeri

17.Y8 – Elle çizilmiş rakamlardan seçilen dokuzuncu koordinatın y-eksenindeki kesim noktası değeri

İlgili veri kümesi içerisinde, 8 adet farklı koordinat değerini temsil eden 16 nitelik ve ayrıca 1 tane, bu koordinat değerlerinin elle yazılmış hangi rakamdan alındığını sınıflandıran bir sınıf niteliğini barındırmakta olduğu için, sınıflandırma algoritmalarını kullanmak için uygun bir veri seti olarak seçilmiştir. İlgili koordinat değerleri ile veri görselleştirilebilir.

Veri kümesinin sınıflandırılması için 3 adet önemli sınıflandırma algoritması kullanılmıştır: Destek Vektör Makinesi, Rastgele Karar Ormanı ve KNN. Bu algoritmaların gerekli eğitim ve test etme işlemleri sonucunda ortaya çıkan doğruluk (accuracy), hassasiyet (precision) ve F1-skoru hesaplamaları sonucu hangi algoritmanın veri kümesini sınıflandırmada daha başarılı olacağına karar verilmiştir.

Bu makele kapsamındaki Bölüm 2'de yapılan çalışmalara dair literatür araştırmalarından, bu literatür araştırmalarının yapılma sebeplerinden, ve literatür araştırmaları sonucu kullanılan veri madenciliği teorilerinden bahsedilmiştir. Bölüm 3'te ise çalışmanın işleyişi, ilgili hesaplamalar ve bu hesaplamaların sonucunda çıkan hassasiyet, F1-skoru ve

doğruluk sonuçları anlatılmıştır. Son olarak Bölüm 4'te, çalışmalar sonucu yapılan çıkarımlardan ve bu çalışmaların şirket, devlet ya da devlet olmayan organizasyonlardaki

II. LİTERATÜR İNCELEMELERİ

Pise-Kulkarni, Sınıflandırıcıların (classifiers) ve makine öğrenmesi algoritmalarının performansının ya da sıralamalarının tahmin edilebilmesi ve yapılacak olan sınıflandırma işlemlerine en uygun modelin seçilebilmesi, veri madenciliği araştırmalarının konularından birisi olduğunu belirtmektedir. [2]. Bu sebeple, yapılan çalışmada doğru sınıflandırma algoritmasını seçmek önemlidir. Çalışma kapsamında, seçilen veri kümesinin 10 adet sınıfı olması ve veri kümesinin etiketli (labeled) olmasından ötürü, denetimli (supervised) sınıflandırma algoritmaları arasından araştırma yapılmaya başlanmıştır.

Aragon vd.'nin belirttiğine göre, Veri kümesinin sınıf niteliği olmayan diğer nitelikleri, koordinat olarak belirlendiği ve koordinat verileri arasında uzaklık hesaplarken en optimal seçenek Öklid uzaklığı (Euclidean distance) tabanlı algoritmalardır [3]. Bu nedenle, çalışma kapsamında denetimli algoritmalar arasından Öklid uzaklığı tabanlı algoritmalar tercih edilmiştir.

Veri setinin grafikleştirilmesi sürecinde, bazı niteliklerde çok sayıda uç değer (outlier) olduğu tespit edilmiştir. Bu sebeple algoritma seçimi sırasında uç değerlere karşı dayanıklı (robust) olan algoritmalar tercih edilmiştir.

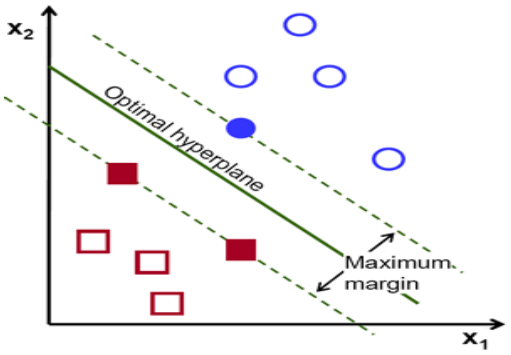
Veri kümesindeki sınıflar, homojen olarak dağılmıştır. Bu durum algoritma seçiminde, heterojen olarak dağılan veri kümeleri için kullanılması zor olan algoritmaları eleme zorunluluğunu ortadan kaldırmıştır.

Bu değerlendirilmeler sonucunda, 3 adet algoritma belirlenmiştir: SVM (Support Vector Machine), KNN (K-Nearest Neighbour) ve Rastgele Karar Ormanları (Random Forest) algoritmaları.

A. *Destek Vektör Makineleri (Support Vector Machine)*

Gongde vd., SVM'nin (Support Vector Machine) makine öğrenmesinde, özellikle sınıflandırma işlemlerindeki uygunluğu nedeniyle öncü bir araç olduğunu belirtmektedir [4]. Gongde vd, SVM'nin sunduğu garantilenmiş yakınsama (guarenteed convergence), iyi genelleme yetisi (good generalization capability) ve çok boyutlu (high-dimensional) verinin işlenmesindeki başarısı sayesinde veri bilimi araştırma dünyasında önemli bir yer edindiğinden bahsetmektedir. [4].

SVM algoritmasında temel amaç, N-boyutlu bir uzayda veri noktalarını ayrı bir şekilde sınıflandıracak olan hiperdüzlemi (hyperplane) bulmaktır. Sınıflandırma problemlerinde asıl amaç, farklı sınıfları birbirlerinden ayrıştırmak ve yeni gelecek verilerin hangi sınıfa ait olduğunu belirleyebilmek olduğu için, SVM algoritmasında sınıfları birbirinden ayırmak için bir hiperdüzlem çizilir. Bu çizilen hiperdüzleme paralel olan ve eşit uzaklıkta olan 2 farklı hiperdüzlem ile oluşan alana marjin (margin) adı verilir. Marjin ne kadar büyük olursa, sınıflar birbirinden o kadar iyi ayrılırlar.

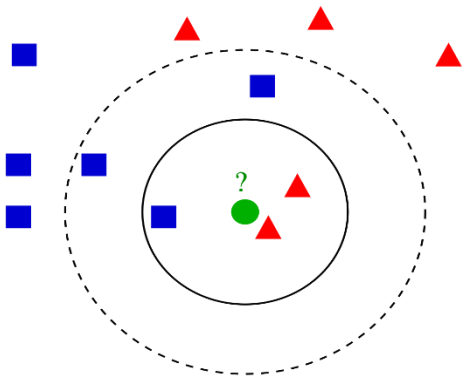


Görsel 1. İki farklı sınıfı birbirinden ayıran hiperdüzlem görüntüsü. Hiperdüzleme paralel olan yeşil noktalar, hiperdüzleme paralel ve eşit uzaklıkta olan ve marjini oluşturan hiperdüzlemleri temsil etmektedir.

Araştırmalar sonucunda kullanılacak ilk algoritma olarak SVM algoritması seçilmiştir. SVM algoritması, çalışmada kullanılacak olan veri kümesi olan El İle Yazılmış Rakamları Kalem Bazlı Tanıma veri kümesini sınıflandırmak için uygun bir algoritma olarak görülmüştür. SVM algoritması, hiperdüzlem çizimi yaparken veri kümesindeki tüm değerleri kullanmadığı ve genelde optimal değerleri kullandığı için, veri kümesinde bulunan uç değerlerden fazla etkilenmeyeceği öngörülmüştür. Aynı zamanda genelde Öklid uzaklığı tabanlı bir algoritma olduğu için, veri kümesinin bağımsız değişkenlerinin koordinat olarak belirlenmesi ve bu sebeple Öklid uzaklığı kullanılarak yapılan hesaplamaların veri kümesine daha uygun olması, SVM algoritmasını kullanmakta karar kılınmasını sağlamıştır.

B. *K- En Yakın Komşu Algoritması*

Genuer vd. KNN (K-Nearest Neighbour) algoritmasının parametrik olmayan, hem regresyon hem sınıflandırma problemlerini çözmek için kullanılan bir denetimli öğrenme algoritması olduğundan bahsetmiştir. Genuer vd.'ne göre, algoritmanın parametrik olmaması algoritmayı basit kılsa da, bu basitlik çoğu zaman daha etkili bir algoritma olmasını sağlamaktadır [5]. Bu algoritma kapsamında, tahminde bulunmak istediğimiz veri birimine (bu sınıflandırma algoritmaları için ilgili sınıf değeri olarak seçilir) en yakın K tane farklı veri birimi seçilmesi ve bu seçilen k tane veri biriminin sahip oldukları bağımlı değişkenleri yardımıyla tahminde bulunulmak istenen veri biriminin tahmin edilmesi sağlanır.



Görsel 2. KNN algoritmasına dair bir örnek gösterim. Yeşil ile gösterilen veri birimi, tahmin yapılmak istenen birimdir. Eğer k değeri 3 seçilirse, en yakınındaki 3 veri biriminden 2 tanesi kırmızı olduğu için, tahmin edilecek birim kırmızı olarak tahmin edilir. Eğer k değeri 5 seçilirse, en yakınındaki 5 veri biriminden 3 tanesi mavi olduğu için, veri birimi mavi olarak tahmin edilir.

KNN algoritmasındaki en önemli işlem, ilgili k değişkeninin seçilmesidir.K değişkeninin en optimal değerini bulmak için birtakım denemeler yapılır. Bu işlem, bir k değerinden sonra sınıflandırma işlemlerinin doğruluk değerleri düşmeye başlıyorsa, k değişkenine, doğruluk değerinin düşmeye başlamasından önceki k değerinin atanmasıyla sonlandırılır.

Araştırmalar sonucunda KNN algoritması kullanılacak ikinci algoritma olarak seçilmiştir. KNN algoritması, uç değerlerden etkilenen bir algoritmadır. Fakat veri kümesindeki uç değer sayısının çok yüksek olmaması ve uç değer tespitlerinin sadece 2 sınıf için yapılmış olmasından dolayı bu sorun görmezden gelinmiştir. Aynı zamanda, Aragon vd.'ne göre KNN algoritması genelde Öklid uzaklığı tabanlı bir algoritma olduğu için, çalışmada kullanılan veri kümesi için uygun bir seçenek olarak görülmüştür. Aragon vd'nin yaptığı çalışmalarda yüksek performans verdiği tespit edildiği için, ilgili veri setini sınıflandırırken kullanılacak algoritmalardan biri olmasında karar kılınmıştır [3].

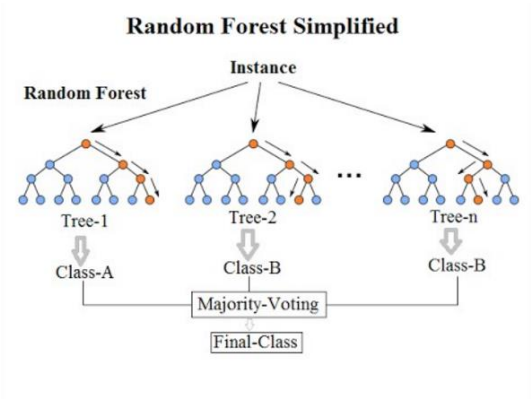
C. Rastgele Karar Ormanları

Rastgele Karar Ormanları (Random Forest) algoritması, birden çok karar ağacı üzerinden, her bir karar ağacının farklı bir sınıf üzerinde eğitilmesi ve bu karar ağaçları yardımıyla tahmin yapılabilmesi sağlanarak kullanılan bir denetimli sınıflandırma algoritmasıdır. Esnek bir algoritma olmasından ve kolay kullanılabilir olmasından dolayı, veri madenciliği araştırmalarında yaygın olan bir algoritmadır.

Biau-Scornet'e göre rastgele karar ormanları, genel amaçlı (general-purpose) bir sınıflandırma ve regresyon modeli olarak oldukça başarılıdır. Biau-Scornet, Rastgele Karar Ormanlarının geniş çaplı tahmin problemlerine uygulanabilir olması ve bunu yaparken oldukça az sayıda parametre üzerinde değişiklik yapılması sayesinde, bu algoritmanın popülerliğinin artmakta olduğunu söylemektedirler [7].

Rastgele Karar Ormanları algoritmasında, ilk olarak her sınıf için birer karar ağacı belirlenir. Her bir karar ağacı için tahmin sonucu oluştuktan sonra, eğer algoritma sınıflandırma işlemi için kullanılıyorsa bu sonuçların modu, regresyon için kullanılıyorsa bu sonuçların ortalaması alınır. Son olarak da bu mod ya da ortalama işlemleri sonucunda en yüksek değere sahip olan karar ağacında bulunan sınıf, tahmin sonucu olarak belirlenir.

Rastgele karar ormanları, birlik öğrenmesi (ensemble learning) tekniğini kullanır. Yani problem çözümü için birden çok sınıflandırıcı oluşturup bunları birleştirme yöntemiyle işlemlerini gerçekleştirir. İçerisinde birçok karar ağacı (decision tree) bulundurur. Bu nedenle isminde "orman" vardır. Bu orman içerisinde, torbalama (bagging) algoritması ile eğitim sürecini gerçekleştirilir. Torbalama algoritması, basit bir birleştirme (ensemble) algoritmasıdır. Birden çok makine öğrenme sınıflandırıcısının sonuçlarını toplayarak daha doğru bir sonuç elde etme esasına dayanır. Torbalama algoritması, karar ağaçlarındaki sonuçların daha doğru bir sonuç döndürmesini sağlamak için, oluşturulan her karar ağacının sonucu birbirleri ile karşılaştırılarak en doğru sonuç aralarından seçilir. Rastgele karar ormanlarına dair daha detaylı bilgiler Genuer et al. (2008) tarafından oluşturulmuş araştırmadan elde edilebilir.



Görsel 3. Rastgele Karar Ormanları algoritması diyagramı

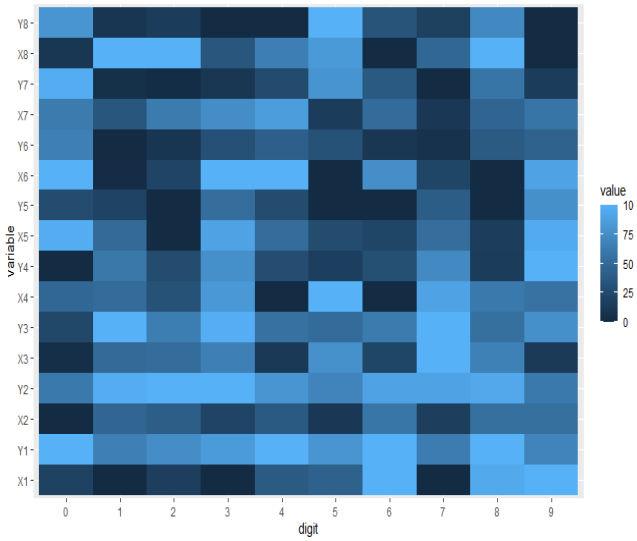
Rastgele karar ormanları, makine öğrenmesi ve veri madenciliği alanlarındaki büyük problemlerden biri olan aşırı öğrenme (over-fitting) sorunun yaşanmasını engeller. Aynı zamanda torbalama algoritması ile, uç değerlerin sonucu etkilemesini engelleyerek doğruluğun uç değerler sebebiyle azalmasını engeller. Bu faydalardan dolayı, çalışma kapsamında üçüncü sınıflandırma algoritması olarak Rastgele Karar Ormanları çalışmaya dahil edilmiştir.

Bu algoritmaların birbiriyle karşılaştırmasına dair literatür araştırmaları gerçekleştirilmiştir. Thanh-Kappas'a göre, bu üç algoritma sınıflandırıcı olarak yüksek doğruluk sonuçları elde etmek için en önde gelen algoritmalardır. Thanh-Kappas'ın çalışmasına göre, SVM ortalama olarak eğitim örneklem sayısına en az hassaslık göstermesine rağmen, en yüksek doğruluk sonuçlarını elde etmiştir. SVM algoritmasını doğruluk sonuçları için Rastgele Karar Ormanları ve KNN algoritması takip etmiştir. KNN algoritması ve Rastgele Karar Ormanları algoritmasının sonuçları arasında, eğitim örneklem sayısındaki artışa göre fark artmıştır. Her üç sınıflandırıcı algoritma için de, eğitim örnekleminin büyüklüğü yeterli iken doğruluk sonuçları ortalama olarak aynı ve yüksek çıkmıştır [7].

III. SINIFLANDIRMA ÇALIŞMALARI VE BULUŞLARI

Sınıflandırma çalışmalarına başlamadan önce, veri kümesindeki uç değer tespitleri ve eksik veri olup olmadığına dair tespit işlemleri gerçekleştirilmiştir. Bu işlemler sonucunda, rakamlar üzerinden seçilen birinci koordinatın y-ekseni değerlerini temsil eden "Y1" değişkeninde, rakamlar üzerinden seçilen ikinci koordinatın y-ekseni değerlerini temsil eden "Y2" değişkeninde ve rakamlar üzerinden seçilen yedinci koordinatın x-ekseni değerlerini temsil eden "X7" değişkeninde uç değer tespit edilmiştir. Veri kümesinde eksik değer olmadığı tespit edilmiştir.

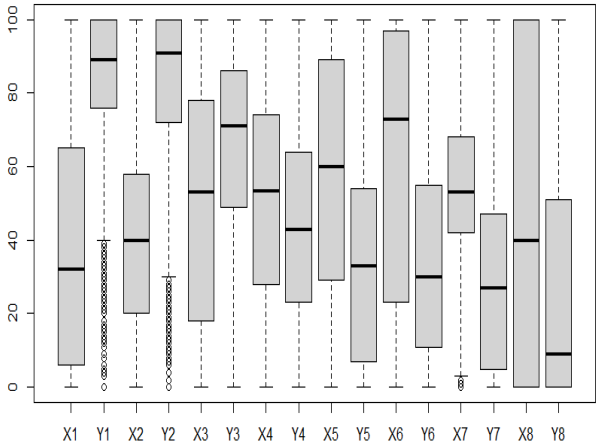
Veri kümesi, UCI Makine Öğrenme Kaynağı (UCI Machine Learning Repository) sayfasında, veri setinin eğitim (training) ve deneme (test) klasörleri ayrı olarak bulunmaktadır. Böylece veri kümesinin eğitim ve deneme kümelerine ayrılması da gerçekleştirilmiştir.



Görsel 4. Çalışma kapsamındaki veri kümesine dair heatmap

SVM Modeli Çıktıları			
Accuracy: 0.9791			
Sınıf	***F1 Skoru***	***Precision***	***Recall***
0	0.9832	1	0.9669
1	0.9564	0.9486	0.9643
2	0.9811	0.9654	0.9973
3	0.98667	0.98230	0.99107
4	0.9917	0.9972	0.9863
5	0.98209	0.98209	0.98209
6	0.99851	0.99703	1
7	0.96034	0.99123	0.93123
8	0.97953	0.96264	0.99702
9	0.97337	0.96765	0.97917

Tablo 1. SVM modeline dair doğruluk (accuracy), kesinlik (precision), duyarlılık (recall) ve F1 skoru değerleri



Görsel 5. Çalışma kapsamındaki veri kümesine dair boxplot grafiği. Grafikte görüldüğü üzere Y1, Y2, ve X7 değişkenlerinde uç değerler bulunmaktadır.

A. *SVM Modeli İmplementasyonu ve Sonuçları*

Çalışma kapsamındaki veri kümesine ait eğitim örneklemi kullanılarak, SVM modeli oluşturulmuş ve eğitilmiştir.

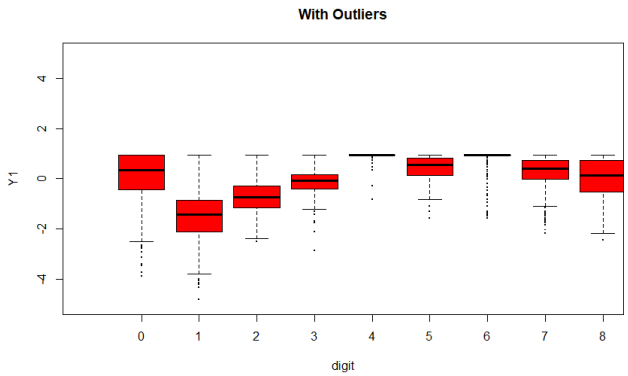
Eğitim sonucunda oluşan model ile tahminler yapılmıştır ve çalışmada kullanılan veri kümesine ait test örneklemleri ile karşılaştırılmıştır. Sonuç olarak model tarafından 0 rakamına dair 12 tane, 1 rakamına dair 13 tane, 2 rakamına dair 1 tane, 3 rakamına dair 3 tane, 4 rakamına dair 5 tane, 5 rakamına dair 6 tane, 6 rakamına dair 0 tane, 7 rakamına dair 25 tane, 8 rakamına dair 1 tane, 9 rakamına dair 4 tane hatalı tahmin yapılmıştır. SVM modeline dair kesinlik (precision), duyarlılık (recall), F1 skoru ve doğruluk (accuracy) değerleri *Tablo 1*'de gösterilmiştir.

B. *KNN Modeli İmplementasyonu ve Çıktıları*

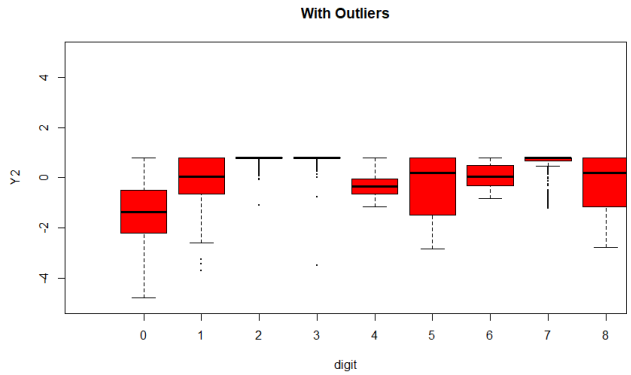
Çalışma kapsamındaki veri kümesine ait eğitim örneklemi kullanılarak, SVM modeli oluşturulmuş ve eğitilmiştir.

Eğitim sonucunda oluşan model yardımıyla tahminler yapılmıştır. 0 rakamı için 10 tane, 1 rakamı için 27 tane, 2 rakamı için 2 tane, 3 rakamı için 3 tane, 4 rakamı için 1 tane, 5 rakamı için 7 tane, 6 rakamı için 0 tane, 7 rakamı için 17 tane, 8 rakamı için 2 tane ve 9 rakamı için de 17 tane yanlış tahmin yapılmıştır.

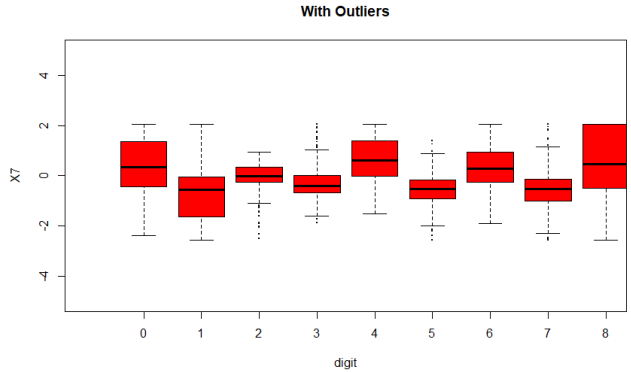
Bu sonuçlara bakarak ve KNN algoritmasının uç değerlere karşı dayanıksız bir algoritma olduğu düşünülürse, yanlış tahmin sayısı fazla olan sınıfların, daha önce uç değer tespiti yapılmış değişkenlerinde, diğer sınıflara göre daha fazla uç değere sahip olduğu yaklaşık olarak söylenebilmektedir. Bunu kanıtlamak için gerekli kutu grafiği (boxplot) hesaplamaları, *Görsel 6*, *Görsel 7* ve *Görsel 8*'de görülmektedir. İlgili kutu grafiklerinden daha uygun sonuç alabilmek için, veri kümesindeki veriler standartize edilmiştir. Yani verilerin orijinal değerleri z-skoru (z-score) değerine dönüştürülmüştür.



Görsel 6. Y1 değişkenine dair, her sınıfa ait uç değerleri gösteren kutu grafiği



Görsel 7. Y2 değişkenine dair, her sınıfa ait uç değerleri gösteren kutu grafiği



Görsel 8. X7 değişkenine dair, her sınıfa ait uç değerleri gösteren kutu grafiği

Görsel 6, Görsel 7 ve *Görsel 8*'deki kutu grafiği sonuçları ile, KNN modeli yardımıyla yapılan tahminlerdeki yanlış tahmin sonuçları karşılaştırıldığında, kutu grafiğinden uzakta uç değerleri daha fazla olan değişkenler için yaklaşık olarak daha fazla hatalı tahmin yapıldığı görülmektedir.

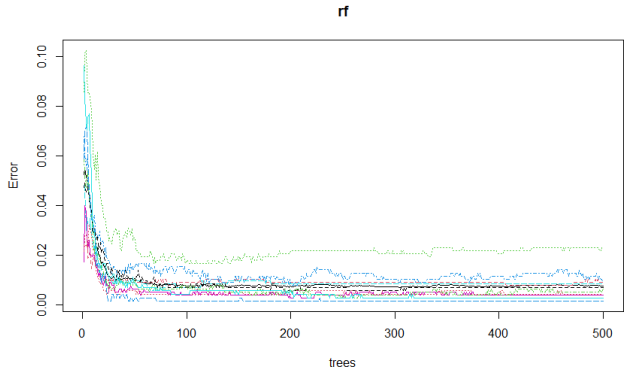
KNN modeline dair doğruluk (accuracy), kesinlik (precision), duyarlılık (recall) ve F1 skoru değerleri *Tablo 2*'de gösterilmiştir.

KNN Modeli Çıktıları			
Accuracy: 0.972			
Sınıf	***F1 Skoru***	***Precision***	***Recall***
0	0.9847	0.9972	9.9725
1	0.93352	0.94134	0.92582
2	0.9628	0.9330	0.9945
3	0.97368	0.95690	0.99107
4	0.9806	0.9916	0.9698
5	0.97041	0.96188	0.0.97910
6	0.99408	0.98824	1
7	0.96774	0.98854	0.94780
8	0.99257	0.99110	0.99405
9	0.96229	0.97554	0.94940

Tablo 2. KNN modeline dair doğruluk (accuracy), kesinlik (precision), duyarlılık (recall) ve F1 skoru değerleri

C. Rastgele Karar Ormanları İmplementasyonu ve Çıktıları

Çalışma kapsamındaki veri kümesine ait eğitim örneklemi kullanılarak, Rastgele Karar Ormanları modeli oluşturulmuş ve eğitilmiştir. Uygulanan model için 500 tane karar ağacı sayısı belirlenmiştir. Eğitim sürecindeki hata miktarı ve ağaç sayısının birbiriyle ilişkisini gösteren grafik, *Görsel 9*'da gösterilmiştir.



Görsel 9. Rastgele Karar Ağacı modelinin eğitimi sırasındaki ağaç sayısı ile hata miktarı arasındaki ilişkiyi gösteren grafik. Ağaç sayısı arttıkça hata oranın azaldığı görülmüştür.

Eğitim sonucunda oluşan model yardımıyla yapılan tahminlerden, 0 rakamı için 16 tane, 1 rakamı için 29 tane, 2 rakamı için 6 tane, 3 rakamı için 5 tane, 4 rakamı için 1 tane, 5 rakamı için 21 tane, 6 rakamı için 0 tane, 7 rakamı için 38 tane, 8 rakamı için 0 tane ve 9 rakamı için 4 tane yanlış tahmin yapılmıştır. Rastgele Karar Ormanı modeline dair doğruluk (accuracy), kesinlik (precision), duyarlılık (recall) ve F1 skoru değerleri Tablo 3'te gösterilmiştir.

Rastgele Karar Ağacı Modeli Çıktıları			
Accuracy: 0.9648			
Sınıf	***F1 Skoru***	***Precision***	***Recall***
0	0.9775	1	0.9559
1	0.91382	0.91008	0.91758
2	0.9534	0.9251	0.9835
3	0.97784	0.97067	0.98512
4	0.9973	0.9973	0.9973
5	0.96764	1	0.93731
6	1	1	1
7	0.93948	0.98788	0.89560
8	0.97533	0.95184	1
9	0.94964	0.91922	0.98214

Tablo 3. Rastgele Karar Ormanları modeline dair doğruluk (accuracy), kesinlik (precision), duyarlılık (recall) ve F1 skoru değerleri

Rastgele Karar Ağaçları sınıflandırma algoritmasının en büyük sıkıntılarından birisi, eğitim veri kümesindeki verilerin değer aralığı ile test veri kümesindeki verilerin değer aralığının birbirinden çok uzak olduğu durumlarda verimsizleşmesidir. Çalışma kapsamındaki veri kümesinde, böyle bir durum söz konusu olmadığı için herhangi bir problem yaşanmamıştır. Fakat bu durumu göstermek için, eğitim veri kümesindeki değer aralığından daha uzaktaki bir değer aralığı olan ikinci bir test veri kümesiyle,

implementasyonu yapılmış model ile tahminler yapılmıştır. Sonuçları *Tablo 4*'te görülmektedir.

Rastgele Karar Ormanları Modeli Farklı Test Veri Kümesi ile Çıktıları			
Accuracy: 0.1801			
Sınıf	***F1 Skoru***	***Precision***	***Recall***
0	N/A	N/A	0
1	N/A	N/A	0
2	N/A	N/A	0
3	N/A	0.95690	0
4	N/A	N/A	0
5	N/A	N/A	0
6	N/A	N/A	0
7	N/A	N/A	0
8	0.55766	0.40859	0.87798
9	0.21530	0.12068	0.99702

Tablo 4. Farklı veri seti ile oluşturulmuş Rastgele Karar Ormanları modeline dair doğruluk (accuracy), kesinlik (precision), duyarlılık (recall) ve F1 skoru değerleri

Tablo 4'e bakarak,s onuçlarda gözle görülür bir düşüş gözlemlenebilir.

IV. SONUÇ VE ÖNERİLER

Elde edilen sonuçlar incelendiğinde, SVM, KNN ve Rastgele Karar Ormanları, Elle Yazılmış Rakamların Kalem Bazlı Tanınması isimli veri kümesi için, tahmin yapmak amacıyla kullanılmaya uygun algoritmalardır. Elde edilen sonuçlar sonucunda, aralarından seçilebilecek en uygun model SVM olmuştur. KNN modeli de SVM modelini takip etmektedir. Çalışma kapsamındaki veri kümesinin değişkenleri koordinat tabanlı oldukları için, Öklid Mesafesi algoritması bazlı sınıflandırma algoritmaları, bu veri kümesinin sınıflandırılması için daha uygun olmuştur. Öklid Mesafe algoritması tabanlı algoritması tabanlı olan SVM ve KNN modelleri yardımıyla yapılan tahminlerden daha yüksek doğruluk sonuçları alınmış olması da, bu durumu tescillemektedir. Bu iki algoritma arasından SVM modelinin daha uygun olmasının sebebi de, KNN algoritmasının uç değerlerden etkilenen bir algoritma olmasının, SVM modelinin ise uç değerlere karşı dirençli bir algoritması olması dolayısıyla gerçekleştiği söylenebilmektedir.

Bu iki algoritmanın yanı sıra, Rastgele Karar Ormanları algoritmasının aldığı doğruluk sonucu da oldukça yüksektir. Rastgele Karar Ormanları modeli ile eğitim sürecinde daha yüksek bir ağaç (tree) parametresi kullanarak, daha yüksek doğruluk sonuçları alınabilir fakat, bu şekilde zaman ve kaynaktan tasarruf edilmesi çok mümkün değildir. Bu nedenle ilgili veri kümesi için, diğer iki modelin tercih edilmesi daha doğru bir karar olacaktır.

Sınıflandırma işlemleri gerçekleştirmek üzerine çalıştığımız veri kümesi, uç değerleri olan, değişkenleri koordinat tabanlı olan ve eğitim veri kümesindeki değer aralığı ile test kümesindeki değer aralığı birbirinden çok uzak olmayan bir veri kümesiydi. Bir organizasyonun sınıflandırma işlemleri gerçekleştirmesi gerekeceği durumlarda, bu organizasyonun elindeki veri kümesi bu tarz bir karakteristiğe sahip ise, makale boyunca üzerinde durulan bu sınıflandırma algoritmaları uygun seçenekler olarak tespit edilmiştir. Eğer uç değerler olması gerekenden çok fazla olursa KNN algoritmasının, eğer eğitim veri kümesinin değer aralığı ile test veri kümesinin değer aralığı arasındaki fark artarsa da Rastgele Karar Ormanları algoritmasının performansının düşeceği öngörülerek çalışmaların ilerletilmesi önemli olacaktır.

REFERENCES

[1] E. Alpaydin, Fevzi. Alimoglu (1998). UCI Machine Learning Repository [http://archive.ics.uci.edu/ml]. İstanbul Türkiye,Boğaziçi Üniversitesi Bilgisayar Mühendisliği Bölümü

[2] N. Pise and P. Kulkarni, "Algorithm selection for classification problems," 2016 SAI Computing Conference (SAI), 2016, pp. 203-211, doi: 10.1109/SAI.2016.7555983.I. S. Jacobs and C. P. Bean, "Fine particles, thin films and exchange anisotropy," in Magnetism, vol. III, G. T. Rado and H. Suhl, Eds. New York: Academic, 1963, pp. 271–350.

[3] Aragon, Nyssa, William Arbuthnot Sir Lane and Fan Zhang. "Classification of Hand-Written Numeric Digits." (2013).

[4] Guo, Gongde & Wang, Hui & Bell, David & Bi, Yaxin. (2004). KNN Model-Based Approach in Classification. M. Young, The Technical Writer's Handbook. Mill Valley, CA: University Science, 1989.

[5] Genuer, Robin & Poggi, Jean-Michel & Tuleau, Christine. (2008). Random Forests: some methodological insights. 6729.

[6] Thanh Noi, P., & Kappas, M. (2017). Comparison of Random Forest, k-Nearest Neighbor, and Support Vector Machine Classifiers for Land Cover Classification Using Sentinel-2 Imagery. Sensors (Basel, Switzerland), 18(1), 18. https://doi.org/10.3390/s18010018

[7] Biau, G., & Scornet, E. (2016). A random forest guided tour. Test, 25(2), 197-227.

[8] Thanh Noi, P., & Kappas, M. (2017). Comparison of Random Forest, k-Nearest Neighbor, and Support Vector Machine Classifiers for Land Cover Classification Using Sentinel-2 Imagery. Sensors (Basel, Switzerland), 18(1), 18. https://doi.org/10.3390/s18010018