T.C.
ADIYAMAN ÜNİVERSİTESİ
LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ

YÜKSEK LİSANS TEZİ

SENTETİK VERİ ÖRNEKLEME YÖNTEMLERİNE MATEMATİKSEL YAKLAŞIMLAR

ABDULLAH DAL

MATEMATİK ANABİLİM DALI

ADIYAMAN, 2021

T.C.
ADIYAMAN ÜNİVERSİTESİ
LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ

SENTETİK VERİ ÖRNEKLEME YÖNTEMLERİNE MATEMATİKSEL YAKLAŞIMLAR

Abdullah DAL

Yüksek Lisans Tezi

Matematik Anabilim Dalı

Uygulamalı Matematik Bilim Dalı

Bu tez 03/12/2021 tarihinde aşağıdaki jüri üyeleri tarafından oybirliği/oyçokluğu ile kabul edilmiştir.

Doç. Dr. İbrahim Halil GÜMÜŞ
I. Danışman

Dr. Öğr. Üyesi Serkan GÜLDAL
II. Danışman

Prof. Dr. Seyit TEMİR
Üye

Dr. Öğr. Üyesi Yaşar NACAROĞLU
Üye

Dr. Öğr. Üyesi Hüseyin KUTLU
Üye

Prof. Dr. Tayfun SERVİ
Enstitü Müdürü

ÖZET

Yüksek Lisans Tezi

SENTETİK VERİ ÖRNEKLEME YÖNTEMLERİNE MATEMATİKSEL YAKLAŞIMLAR

Abdullah DAL

Adıyaman Üniversitesi
Lisansüstü Eğitim Enstitüsü
Matematik Anabilim Dalı

1. Danışman : Doç. Dr. İbrahim Halil GÜMÜŞ
2. Danışman : Dr. Öğr. Üyesi Serkan GÜLDAL
Yıl : 2021, Sayfa sayısı: 37

Jüri : Doç. Dr. İbrahim Halil GÜMÜŞ
: Dr. Öğr. Üyesi Serkan GÜLDAL
: Prof. Dr. Seyit TEMİR
: Dr. Öğr. Üyesi Yaşar NACAROĞLU
: Dr. Öğr. Üyesi Hüseyin KUTLU

Bu tez çalışmasında dengesiz dağılıma sahip veri kümelerinin makine öğrenimi algoritmalarında performans kayıplarını iyileştirmeye yönelik bir metot önerilmiştir. Veri kümelerindeki dengesizliği azaltmak veya tamamen kaldırmak için birçok çalışma yapılmıştır (RUS, ROS, SMOTE). Geliştirilen metotta benzer şekilde azınlık sınıfa ait mevcut örnekler, yeniden sentetik olarak çoğaltılmıştır ve veri kümeleri dengelenmiştir. Yeniden örnekleme işlemi için, azınlık sınıfa ait örnekler arasında, Öklid uzaklık metriğiyle tüm veri noktaları için en yakın komşular tespit edilmiştir. Bu komşular arasında yeterli sayıda en yakın komşular arasında olmak üzere, diğer yöntemlerden farklı olarak Ağırlıklı Geometrik Ortalama kullanılarak istenen sayıda yeni sentetik örnekler oluşturulmuştur. Bu şekilde dengelenen veri kümelerinin makine öğrenim performanslarında karşılaştırılan metotlara göre ciddi iyileştirmeler gözlemlenmiştir.

Anahtar Kelimeler: Yeniden Örnekleme; Dengesiz Veri; SMOTE

ABSTRACT

MSc Thesis

MATHEMATİCAL APPROACHESTO SYNTHETİC DATA SAMPLİNG METHODS

Abdullah DAL

Adiyaman University
Graduate School of Natural and Applied Sciences
Department of mathematıcs

Supervisor 1 : Assoc. Prof. İbrahim Halil GÜMÜŞ
Supervisor 2 : Asst. Prof. Serkan GÜLDAL
Year : 2021, Number of pages: 37

Jury : Assoc. Prof. İbrahim Halil GÜMÜŞ
: Asst. Prof. Serkan GÜLDAL
: Prof. Seyit TEMİR
: Asst. Prof. Yaşar NACAROĞLU
: Asst. Prof. Hüseyin KUTLU

In this thesis, a method is proposed to improve performance losses in machine learning algorithms of unevenly distributed datasets. Many studies have been done to reduce or completely remove the imbalance in datasets (RUS, ROS, SMOTE). Similarly, in the developed method, existing samples belonging to the minority class were reproduced synthetically and the datasets were balanced. For the resampling process, the nearest neighbors for all data points were determined using the Euclidean distance metric among the samples belonging to the minority class. Different from the other methods, a desired number of new synthetic samples were created using the Weighted Geometric Average, among these neighbors, in a sufficient number among the nearest neighbors. Significant improvements were observed in the machine learning performance of datasets balanced in this way, compared to the methods compared.

Key Words: Resampling; Unbalanced Data; SMOTE

BEYAN

"Sentetik Veri Örnekleme Yöntemlerine Matematiksel Yaklaşımlar" başlıklı tezimde çalışmaların tamamen akademik kurallara ve etik değerlere sadık kalınarak yürütüldüğünü ve yazımda yararlandığım eserlerin kaynakçada gösterilenlerden oluştuğunu ayrıca alıntılardan bilimsel etiğe uygun atıf yaparak yararlanmış olduğumu beyan ederim.

ABDULLAH DAL

TEŞEKKÜR

Tez çalışmam boyunca yardım ve desteklerini benden esirgemeyen danışman hocalarım sayın Doç. Dr. İbrahim Halil GÜMÜŞ ve Dr. Öğr. Üyesi Serkan GÜLDAL'a teşekkürü büyük bir borç bilirim.

Ayrıca zaman zaman tez içeriği ile ilgili sorularımda bana yardımcı olan Öğretim Görevlisi Mustafa YAVAŞ hocama saygılarımı sunarım.

Son olarak eğitim hayatım boyunca maddi ve manevi destekleriyle beni hiçbir zaman yalnız bırakmayan annem, babam, kardeşlerim ve eşime teşekkür ederim.

İÇİNDEKİLER

ŞEKİLLER DİZİNİ

ÇİZELGELER DİZİNİ

1. GİRİŞ

Son yıllarda makine öğrenmesi yöntemleri kullanılarak veri sınıflandırma işlemlerinde büyük gelişmeler yaşanmıştır. Teknolojik gelişmeler arttıkça, internet ortamında ve diğer ortamlarda verilerin boyutu da hızla artmaktadır. İnternet ve diğer dijital platformlarda artan verilerin manuel olarak işlenmesi ve analiz edilmesi mümkün değildir. Bu veriler yapay zekâ tabanlı makine öğrenmesi yöntemleri ile sınıflandırılabilmekte ve geçmiş verilerden tahmine dayalı analizler yapılabilmektedir [1]. Mevcut verileri kullanarak yeni veriler için en uygun modeli bulmak için makine öğrenmesi yöntemleri sürekli olarak geliştirilmektedir. Verilerin analiz edilmesi, veriden faydalı bilgilerin çıkarılması ve yorumlanması işlemleri makine öğrenmesi tabanlı veri madenciliği ile yapılabilmektedir [2]. Tıbbi teşhis, yüz tanıma, metin sınıflandırma, sahte işlemler, spam filtreleme gibi birçok alanda makine öğrenmesi yöntemleri kullanılmaktadır [3]. Bu alanlardaki kullanım amacı, hayatımızdaki karmaşık olaylara uygulanabilir çözümler üretmektir. Ancak bu alanlarda yapılan çalışmalarda makine öğrenmesi ve derin öğrenme tabanlı algoritmalar birçok sorunu da beraberinde getirmektedir. Örneğin dengesiz ve sınıflandırılmamış veriler bu duruma örnek olarak verilebilir. Dengesizlik problemi iki sınıftan birinin diğerine göre daha az örneğe sahip olması durumudur. Özellikle tıbbi alanda kullanılan veri kümelerinin çoğu dengesiz dağılıma sahiptir. Dengesiz dağılıma sahip bir veri kümesi sınıflandırıcı algoritmaların başarı performansını olumsuz yönde etkilemektedir. Bu dağılımı dengelemek ve sınıflandırmak için birçok çalışma yapılmıştır. Bu çalışmalar veri ve algoritma düzeyinde olup, yeniden örnekleme yöntemi ile örneklem azaltma ve örneklem çoğaltma işlemleridir. Bu çalışmanın sonuçlarına göre, dengesiz veri kümesinin dengelenmesinde daha başarılı sonuçlar veren uygulamaların, veri kümesindeki örnekleri artırarak kullanılan yöntemler olduğu gözlemlenmiştir. Daha az örneklem içeren veri kümelerinin, daha çok örneklem içeren veri kümelerine nazaran sonuçlarının daha tutarsız olduğu gözlemlenmiştir. Chawla, Bowyer [4]'in SMOTE algoritmasının veri kümesindeki örnekleri sistematik olarak dengeleyen ve uygulamanın en doğru şekilde çalışmasını sağlayan yöntemlerden biri olduğu görmüştür. Bu çalışmada azınlık sınıfa ait mevcut

örnekler, yeniden sentetik olarak çoğaltılmıştır ve veri kümeleri dengelenmiştir. Yeniden örnekleme işlemi için, azınlık sınıfa ait örnekler arasında, Öklid uzaklığı metriğiyle tüm veri noktaları için en yakın komşular tespit edilmiştir. Bu komşuların birer vektör olduğu düşünülerek iki örnek arasında Ağırlıklı Geometrik Ortalama kullanılıp istenen sayıda yeni sentetik örnekler oluşturulmuştur. Bu işlem sonucunda veri kümeleri dengeli hale getirilmiştir. Ayrıca, dengesiz veri kümelerini dengelemek için Rastgele Az Örnekleme (RUS), Rastgele Aşırı Örnekleme (ROS) ve Sentetik Azınlık Örnekleme Tekniği (SMOTE) yöntemleri de kullanılmıştır. Orijinal ve dengelenmiş veri kümeleri Rastgele Orman (Random Forest) algoritması ile sınıflandırılmış ve sonuçları kıyaslanmıştır. Çalışma sonucunda, yeniden örnekleme yaklaşımı ile dengelenen veri kümelerinin tüm performans değerlerinde artış gözlemlenmiştir. Çalışmada önerilen yaklaşım ile yeniden örneklenerek dengelenen veri kümesi, ham veri kümesi ve diğer yöntemlere kıyasla sınıflandırma performansını iyileştirdiği gösterilmiştir.

2. ÖNCEKİ ÇALIŞMALAR

Makine öğrenmesi günümüzde problemleri çözmek ve beraberinde birçok konuda hayatımızı kolaylaştırmak için kullanılmış olsa da zaman zaman yetersiz kaldığı durumlarda olmaktadır. Yetersiz kalmasındaki en büyük nedenlerden birisi makineleri eğitmek için kullanılan dengesiz veri kümeleridir. Dengesiz veri sınıfların eşit dağılmadığı, yani her sınıf için eşit sayıda verinin olmadığı veri kümesidir. Makine öğrenmesi algoritmaları dengesiz veri kümeleriyle karşılaştığı durumlarda eşit olmayan dağılımı dikkate almayarak performans kayıpları olmaktadır. Yani sınıflandırma algoritmaları veri kümesi ile eğitilirken ağırlıklı olan sınıfa önem vererek, azınlık sınıfını göz ardı edebilmektedir. Dengesiz veri kümelerini dengeli hale getirmek için çeşitli yöntemler ve teknikler geliştirilmiştir. Bunlara kronolojik olarak göz atalım.

Chawla ve diğerleri (2002) dengesiz veri kümelerini dengeli hale getirmek için SMOTE algoritmasını ortaya koymuşlardır [4]. Bu algoritma azınlık sınıfındaki veriler arasında ağırlıklı aritmetik ortalama işlemi yaparak sentetik veriler üretme işlemi yapmaktadır. Bu algoritma, veri kümesindeki dengesizliği kaldıran en iyi algoritmalar arasındadır. Han ve diğerleri (2005) SMOTE yöntemini kullanarak azınlık sınıfında bulunan örneklerin oluşturduğu veri kümesinin sınır çizgilerini kullanarak Borderline-SMOTE yöntemini ortaya atmışlardır [5]. Nguyen ve diğerleri (2011) çalışmalarında azınlık veri kümesi elemanı üretmek için destek vektörlerinin yardımıyla SVM-SMOTE modelini geliştirmişlerdir [6]. Batista ve diğerleri (2004) SMOTE ile rastgele örneklem azaltmada en yakın komşular (ENN) yöntemini kullanarak SMOTE-ENN yöntemini geliştirmişlerdir [7].

Veri sınıflarındaki örnekleri arttırıp azaltma metotlarının performans değerleri karşılaştırılırken, Karar Ağaçları, Destek Vektör Makineleri (DVM), K-en yakın komşu (k-NN) ve Rastgele Orman gibi sınıflandırma algoritmalarından yararlanılmaktadır. Yapılan çalışmalarda gerçekleştirilen deneyler sonucunda dengesiz veri kümelerinin dengeli veri kümelerine dönüştürmek için kullanılan SMOTE tabanlı örneklem çoğaltma yönteminin veri kümelerini dengeli hale getirdiği ve sınıflandırıcıların performans değerlerini arttırdığı gözlemlenmektedir.

3. TEMEL KAVRAMLAR

3.1. Metrik Uzay

Tanım 3.1 (Metrik ve Metrik Uzay) : X boş olmayan bir küme olsun. $d: X \times X \to \mathbb{R}$ fonksiyonu için,

A1-) $d(x, y) = 0 \Leftrightarrow x = y$,

A2-) $d(x, y) = d(y, x)$ simetri özelliği ve

A3-) $d(x, y) \leq d(x, z) + d(z, y)$ üçgen eşitsizliği

şartları sağlanıyorsa d ye X üzerinde bir metrik, (X , d) uzayına da metrik uzay denir. d fonksiyonu her (x, y) çiftine negatif olmayan bir reel sayı tekabül ettirir. X in elemanlarına (X, d) metrik uzayın noktaları ve d(x, y) reel sayısına da x ile y arasındaki uzaklık denir.

Yukarıdaki A1, A2, A3 şartlarına metrik aksiyomları denir. A1 aksiyomu, bir noktanın kendisine uzaklığının sıfır olduğunu ifade eder. A2 aksiyomu ise x noktasının y noktasına olan uzaklığı ile y noktasının x noktasına olan uzaklığının eşit olduğunu gösterir. Düzlemde bir doğru üzerinde bulunmayan farklı üç nokta alındığında, bu noktalar yardımıyla oluşturulan üçgenin bir kenarının uzunluğu diğer iki kenarının uzunlukları toplamından küçüktür. A3 deki aksiyom olan üçgen eşitsizliği deyimi elementer geometrideki bu hususta ileri gelmektedir [8].

Metrik Fonksiyonun Önemli Bazı Özellikleri

1) Metrik fonksiyonu negatif değer alamayan bir fonksiyon yani $d(x, y) \geq 0$ dır. Gerçekten üçüncü metrik aksiyomundan dolayı her $x, y \in X$ için $d(x, x) \leq d(x, y) + d(y, x)$ yazabiliriz. A1 ve A2 aksiyomlarından dolayı buradan $0 \leq 2d(x, y)$ veya $d(x, y) \geq 0$ elde edilir.

2) $x, y, z_1, z_2, ..., z_n \in X$ ve A3 aksiyomunu tekrar uygularsak

$d(x, y) \leq d(x, z_1) + d(z_1, y)$

$$d(x,y) \le d(x,z_1) + d(z_1,z_2) + d(z_2,y)$$

$$\vdots$$

$$d(x,y) \le d(x,z_1) + d(z_1,z_2) + ... + d(z_n,y)$$

yazabiliriz. Şu hâlde bu eşitsizliğe genelleştirilmiş üçgen eşitsizliği denir [8].

Örnek 3.1

$\mathbb{R}$ reel sayılar kümesi olsun. $\mathbb{R}$ üzerinde her $x, y \in \mathbb{R}$ için $d : \mathbb{R}x\mathbb{R} \to \mathbb{R},\ d(x,y) = |x-y|$ şeklinde tanımlanan d fonksiyonunun metrik şartlarını sağladığı kolaylıkla gösterilebilir. Bundan dolayı bu metriğe reel sayıların mutlak değeri veya alışılmış metriği denir [9].

Tanım 3.2 Minkowski Eşitsizliği

$\mathbb{R}^n$ de herhangi $x = (x_1, x_2, ..., x_n)$ ve $y = (y_1, y_2, ...y_n)$ gibi iki nokta verilmiş olsun. Bu noktaları sağlayan $\sqrt{\sum_{i=1}^{n} |x_i + y_i|^2} \le \sqrt{\sum_{i=1}^{n} |x_i|^2} + \sqrt{\sum_{i=1}^{n} |y_i|^2}$ eşitsizliğine Minkowski Eşitsizliği denir [9].

Örnek 3.2

$\mathbb{R}$ 'nin kendisiyle kartezyen çarpımı olan $\mathbb{R}x\mathbb{R} = \mathbb{R}^2$ üzerinde, yani düzlemde $\forall x = (x_1, x_2),\ y = (y_1, y_2) \in R^2$ için $d(x,y) = \sqrt{(y_1 - y_2)^2 + (x_1 - x_2)^2}$ şeklinde tanımlanan d fonksiyonuna $\mathbb{R}^2$ nin alışılmış metriği veya öklid metriği denir.

Benzer şekilde $\mathbb{R}^n = \mathbb{R}x...x\mathbb{R}$ olmak üzere, her $x = (x_1, x_2, ..., x_n)$, $y = (y_1, y_2, ..., y_n) \in \mathbb{R}^n$ için $d(x,y) = \sqrt{\sum_{i=1}^{n} (x_i - y_i)^2}$ şeklinde tanımlanan d fonksiyonuna $\mathbb{R}^n$ nin metriği olduğunu gösterelim.

Çözüm 3.2

Her $x=(x_1,x_2,....,x_n)$, $y=(y_1,y_2,...y_n)$ ve $z=(z_1,z_2,...,z_n)\in\mathbb{R}^n$ için

$A_1)$ i=1,2,3,...,n için $(x_i-y_i)^2\geq 0 \Rightarrow \sum_{i=1}^{n}(x_i-y_i)^2 \geq \sqrt{\sum_{i=1}^{n}(x_i-y_i)^2}=d(x,y)\geq 0$

$A_2)$ $d(x,y)=\sqrt{\sum_{i=1}^{n}(x_i-y_i)^2}=\sqrt{\sum_{i=1}^{n}(-1)^2(y_i-x_i)^2}=\sqrt{\sum_{i=1}^{n}(y_i-x_i)^2}=d(y,x).$

$A_3)$ $d(x,y)=\sqrt{\sum_{i=1}^{n}(x_i-y_i)^2}=\sqrt{\sum_{i=1}^{n}((x_i-z_i)+(z_i-y_i))^2}$ eşitliğinin sağ tarafına Minkowski eşitsizliği uygulanırsa

$\sqrt{\sum_{i=1}^{n}((x_i-z_i)+(z_i-y_i))^2}\leq\sqrt{\sum_{i=1}^{n}(x_i-z_i)^2}+\sqrt{\sum_{i=1}^{n}(z_i-y_i)^2}$ olur. Buradan

$\sqrt{\sum_{i=1}^{n}(x_i-y_i)^2}\leq\sqrt{\sum_{i=1}^{n}(x_i-z_i)^2}+\sqrt{\sum_{i=1}^{n}(z_i-y_i)^2}\Rightarrow d(x,y)\leq d(x,z)+d(z,y)$

elde edilir. Böylece $(\mathbb{R}^n,d)$ metrik uzaydır [9].

3.2. Skaler Ortalamalar

a ve b pozitif sayılar olmak üzere bu sayıların aritmetik, geometrik ve harmonik ortalamaları sırasıyla şu şekildedir. $A(a,b)=\frac{a+b}{2}$, $G(a,b)=\sqrt{a.b}$, $H(a,b)=\left(\frac{a^{-1}+b^{-1}}{2}\right)^{-1}$ şeklinde tanımlanır. Bunların ortalama ($M(a,b)$) olarak adlandırılması için bazı özelliklere sahip olması gerekmektedir. Bu özelliklerden bazıları şunlardır.

(a) $M(a,b)>0$

(b) $a\leq b$ ise $a\leq M(a,b)\leq b$ olmalıdır.

(c) $M(a,b)=M(b,a)$

(d) $M(a,b)$, a , b değerleri monoton artan ifadelerdir.

(e) Bütün α pozitif sayıları için $M(\alpha.a, \alpha.b) = \alpha.M(b, a)$

(f) M(a, b) fonksiyonu a, b için sürekli bir fonksiyondur.

Yukarıda bahsedilen üç ortalama arasında $\forall a, b$ için

$$H(a,b) \le G(a,b) \le A(a,b)$$

eşitsizliği mevcuttur.

Ek olarak $0 \le \alpha \le 1$, a ve b pozitif sayılar olmak üzere $a\nabla_\alpha b = \alpha.a + (1-\alpha).b$, $a\#_\alpha b = a^\alpha.b^{1-\alpha}$, $a!_\alpha b = \left(\alpha.a^{-1} + (1-\alpha).b^{-1}\right)^{-1}$ ifadelerine de sırasıyla ağırlıklı aritmetik ortalama, ağırlıklı geometrik ortalama ve ağırlıklı harmonik ortalama olarak bilinmektedir.

Bunlar arasında da

$$a!_\alpha b \le a\#_\alpha b \le a\nabla_\alpha b$$

eşitsizlikleri mevcuttur . Bu eşitsizlikler matematikte Skaler Young eşitsizlikleri olarak bilinir [10].

3.3. Sınıflandırma Yöntemleri

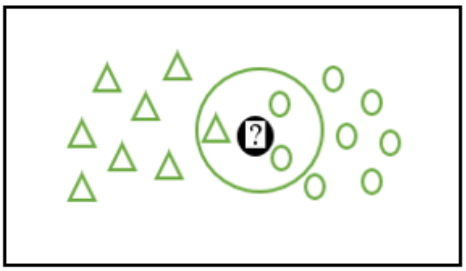
Makine öğrenmesinde verilerin sınıflandırılması için bir çok algoritma ve yöntem kullanılmaktadır. Bu yöntemlerin hepsinin asıl hedefi veriyi en iyi şekilde sınıflandırmak ve en iyi şekilde çalışan örüntüyü elde etmektir. Bunların en popüler olanlarını inceleyelim.

3.3.1. k-en Yakın Komşu

Makine öğrenmesindeki gözetimli öğrenme algoritmalarından biri k-en yakın komşuluk (k-NN) algoritmasıdır. Makine öğrenmesinde çoğunlukla sınıflandırma problemlerinin çözümünde kullanılmaktadır. k-NN algoritması, 1967 yılında T. M. Cover ve P. E. Hart tarafından ortaya atılmıştır [11]. Temel prensip olarak veri kümesindeki elemanların hepsini birbiriyle eşleyerek uzaklık hesaplayan ve bu

uzaklıklarına göre sınıflandırma yapan bir algoritmadır. Bu uzaklıkları hesaplarken genelde birkaç tip uzaklık fonksiyonu kullanılmaktadır. Bunların birkaçı Öklid Uzaklık, Manhattan Uzaklık ve Minkowski uzaklığıdır. k-NN eski ve basit bir yöntem olması sebebiyle en çok kullanılan yöntemlerdendir. Basit olması yanında eksik ve dezavantaj sağladığı yönleri de bulunmaktadır. Bu dezavantajların başında veri kümesi kalabalık olan örneklerde k-NN algoritması, veri kümesindeki tüm noktaların birbirine olan uzaklığının hesaplanıp kaydedilmesi ve karşılaştırma işlemi yapılması nedeniyle uzun bir süreye ihtiyaç duymaktadır. Ayrıca bu verileri kaydetmesi için yüksek bellek alanlarına ihtiyaç duyulmaktadır. k-NN algoritması kısaca şu şekilde çalışmaktadır.

Önce k parametresi belirlenir. Bu parametre kullanılarak veri kümesine yeni katılan noktanın en yakın olan k tane komşusu bulunur. Örneğin k=3 alındığında komşusu aranan noktanın en yakın 3 komşusu ele alınan metrik sayesinde hesaplanarak Şekil 3.3.1.1 de gösterildiği gibi belirlenir. Bulunan komşu elemanlara bakılarak çoğunluk etiketi yeni elemanımızın etiketi olarak belirlenir ve böylece veri kümesine yeni katılan verinin hangi sınıfta olduğuna karar verilmiş olur. Bu işlem tüm noktalar için tekrar edilir. Burada seçilen k sayısının mümkün olduğunca tek sayı seçilmesinde fayda görülmektedir. Çünkü çift olarak alınırsa eşitlik durumunda yeni noktanın hangi etikete sahip olduğunu tahmin etmek zor olmaktadır [19].

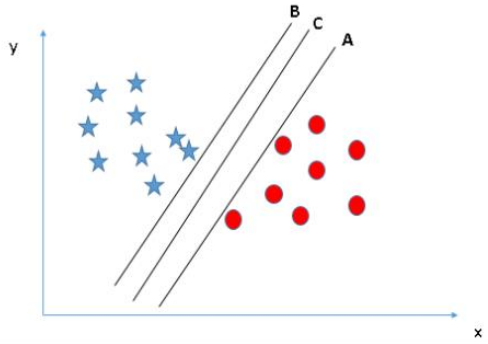


Şekil 3.3.1.1 k-NN algoritmasının basit gösterimi

3.3.2. Destek Vektör Makinesi (DVM)

Destek Vektör Makineleri, sınıflandırma problemlerinde kullanılan gözetimli öğrenme algoritmalarından bir diğeridir. Buradaki amaç veri kümesini iki veya daha

fazla sınıfa ayırmaktır. Bunun için veri kümesinde iki ayrı sınıfa ait olduğu bilinen birbirine en yakın iki noktadan geçen Şekil 3.3.2'deki gibi birbirine paralel iki karar sınırı ve bu iki karar sınırının tam ortasına da bir hiper düzlem çizilerek kümeyi iki sınıfa ayırma işlemi yapılır [19].



Şekil 3.3.2.1 İki boyutlu özellik vektörleri için bir DVM modeli

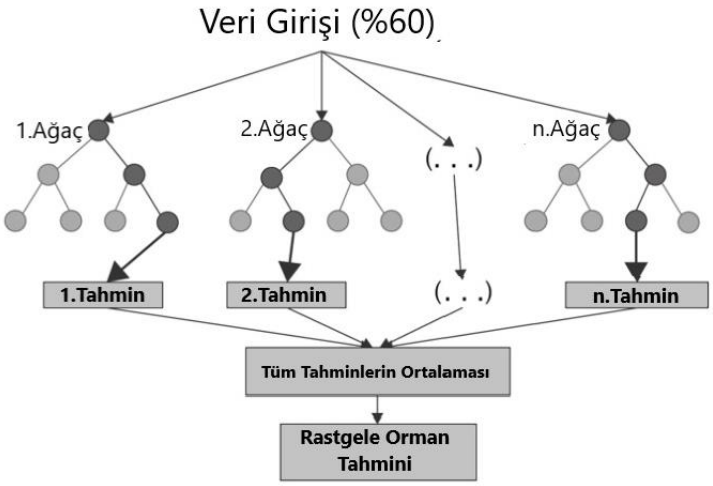
3.3.3. Karar Ağaçları

Karar ağaçları da sınıflandırma ve regresyon işlemlerinde kullanılan ve daha çok kalabalık ve karışık veri barındıran kümeleri sınıflandırma da kullanılan yöntemlerden birisidir. Gelen veri kümelerini belli başlı özelliklerine göre önce bir alt sınıfa daha sonra bunları da başka bir alt sınıfa ayırma işlemi yapar. Karar ağaçları ortaya veri sınıfının tüm ihtimaller doğrultunda çıkmış olan diyagramın kendisi olur. Açık ve anlaşılır bir hali olsa da yeni eklenen elemanlar sayesinde diyagram her seferinde değişmektir [19].

3.3.4. Rastgele Orman Algoritması

Rastgele Orman algoritması, tek bir sınıflandırıcı yerine çoklu sınıflandırıcı tahminleri kullanarak sonuçlar üreten ve sınıflandırıcının tahminlerinden gelen oylarla yeni bir veri örneğini sınıflandıran güçlü bir öğrenme algoritmasıdır [12]. Ayrıca rastgele orman algoritması Şekil 3.3.4.1'de görüldüğü üzere karar ağaçlarının özelleşmiş bir halidir. Karar ağacı veri kümesinin belirli nitelikler altında sınıflandırma işlemidir. Rastgele orman algortiması daha iyi tahmin performansı elde etmek için birden çok karar ağacı modelini kullanır.

Ağaçların her bir düğümünde bir bölünme elde etmek için rastgele bir değişken alt kümesini arar. Sınıflandırma için, girdi vektörü algoritmadaki her ağaca iletilir ve her ağaç bir sınıf için oy verir. Algoritma en çok oyu alan sınıfı seçer [13].



Şekil 3.3.4.1 Rastgele Orman Algoritması

3.4. Karışıklık Matrisi ve Değerlendirme Metrikleri

Karışıklık matrisi, gerçek değerlerin bilindiği bir test verisi üzerinde, sınıflandırma modelinin performansını belirlemek için kullanılan bir matristir. Yani sınıflandırma yapacak olan modelin ne kadar başarılı olduğunu ortaya koyar. Karışıklık matrisinin bir ekseni modelin tahmin edebildiği etiket, diğer ekseni ise gerçek etikettir. Örneğin ikili bir sınıflandırma modelinde iki sınıf öngörülsün: "spam" ve "not spam":

	spam (tahmin edilen)	not spam(tahmin edilen)
spam(gerçek)	**TP**	**FN**
not spam(gerçek)	**FP**	**TN**

Çok sınıflı sınıflandırmalar için karışıklık matrisi daha fazla satır ve sütuna sahip olur. Değerlendirme ölçütlerinde sıklıkla kullanılan terimlerin bir kısmını açıklayalım.

Doğru Pozitif (TP): Sınıflandırıcı tarafından doğru şekilde tanımlanmış pozitif veri sayısı (Örneğin "spam" olan mesajların "spam" gibi kabul edilip kaydedilmiş veri sayısı)

Yanlış Pozitif (FP): Sınıflandırıcı tarafından doğru kabul edilip aslında yanlış olan veri sayısı (Örneğin "not spam" olan mesajların "spam" gibi kabul edilip kaydedilmiş veri sayısı)

Doğru Negatif (TN): Sınıflandırıcı tarafından doğru şekilde tanımlanmış negatif veri sayısı (Örneğin "not spam" olan mesajların "not spam" gibi kabul edilip kaydedilmiş veri sayısı)

Yanlış Negatif (FN): Sınıflandırıcı tarafından yanlış kabul edilip aslında doğru olan veri sayısı (Örneğin "spam" mesajların "not spam" gibi kabul edilip kaydedilmiş veri sayısı)

Yapılan modelleri değerlendirmek, doğruluk durumlarını belirlemek ya da sapma oranlarını belirlemek için bir takım değerlendirme ölçütleri kullanılır.

3.4.1. Doğruluk (Accuracy): Doğruluk, doğru sınıflandırılmış örneklerin sayısının toplam sayıya bölünmesiyle elde edilir. Doğruluk, sınıflandırıcının ne sıklıkta doğru tahmin ettiğini bizlere veren bir ölçüdür. 1'e yaklaştıkça doğru tahminlerde bulunduğunu, 0' a yaklaştıkça da yanlış tahminlerde bulunduğu anlaşılır.

$$\text{Doğruluk} = \frac{TP + TN}{TP + TN + FP + FN}$$

Ayrıca doğruluk oranı üzerinden hata oranını da tahmin etmek mümkündür. Bu da 1'den doğruluk oranının çıkarılması ile elde edilir [19].

$$\text{Hata Oranı} = 1 - \text{Doğruluk} = \frac{FP + FN}{TP + TN + FP + FN}$$

3.4.2. Kesinlik (Precision): Kesinlik, doğru pozitif tahminlerin toplam pozitif tahmin sayısına oranıdır. Bu oran 1'e eşit olduğu vakit sınıflandırıcının üzerinde çalıştığı modelde yanlış örnek olmadığı hepsinin doğru olduğu anlamına gelir. 0'a yaklaştıkça da doğru örnek sayısının azaldığı görülmektedir [19].

$$\text{Kesinlik} = \frac{TP}{TP + FP}$$

3.4.3. Duyarlılık (Recall): Duyarlılık, doğru pozitif tahminlerin toplam pozitif örnek sayısına oranıdır. Bu oran 1'e eşit olduğu vakit sınıflandırıcının üzerinde çalıştığı modelde yanlış örnek olmadığı hepsinin doğru olduğu anlamına gelir. 0'a yaklaştıkça da modelde yanlış örnek sayısının arttığı görülür [19].

$$\text{Duyarlılık} = \frac{TP}{TP + FN}$$

3.4.4. F-Ölçü: Kesinlik ve Duyarlılık oranlarının harmonik ortalamasıdır. Sınıflandırıcının ne kadar iyi çalıştığını bize gösteren bir orandır. Ayrıca sınıflandırıcıları karşılaştırmada da kullanılır [19].

$$F - \text{Ölçü} = \frac{2.\text{kesinlik.duyarlılık}}{\text{kesinlik} + \text{duyarlılık}}$$

3.4.5. ROC Eğrisi: Konulan tanının ne derece güvenilir olduğunu ölçmeye yarayan bir grafik türüdür. Doğru pozitiflerin, yanlış pozitiflere olan oranı ile ifade edilir. ROC eğrisini çizmek için Y eksenine tanı testinin gerçek pozitif değeri (duyarlılık), X eksenine ise yanlış pozitif değerleri (1-özgüllük) yerleştirilir. Bütün kesim noktalarındaki doğru pozitif sayısının yanlış pozitif sayısına karşılık gelen noktalar birleştirilerek ROC eğrisi çizilir. Çizilen ROC eğrisi y=x fonksiyonuna yaklaştıkça konulan tanının gerçek dışı olduğu, y=x den uzaklaşıp arasındaki alan arttıkça da konulan tanının gerçeğe yakın olduğu şeklinde yorumlanır [19].

4. BULGULAR ve TARTIŞMA

4.1. Kullanılan Veri Kümeleri

Veri kümesi, belirli bir konunun ayrıntılarını sayısal değerler veya sözel ifadeler ile derlenip toplanarak dosya şekline getirilmesidir. Bu çalışmada kullanılan farklı büyüklük, farklı örneklem sayısı ve farklı dengesizlik oranlarına sahip 4 gerçek hayat veri kümesi kullanılmıştır. Bu veri kümeleri, KEEL (Knowledge Extraction on Evolutionary Learning) açık kaynaklı yazılım aracı sitesinden alınmıştır [14]. Çalışmada kullanılan veri kümelerinin özellikleri ve kullanılan niteliklerin sayısı Çizelge 4.1.1'de verilmiştir. Çizelge 4.1.1'de belirtilen dengesizlik oranı çoğunluk sınıfının azınlık sınıfına bölünmesiyle elde edilmektedir.

Çizelge 4.1.1'de belirtilen nitelik kavramını açıklayalım. Veri kümesindeki her bir farklı verinin hangi kriterlere göre birbirinden ayrıştığını ifade eden özelliklerdir. Bu nitelikler bir kişiye ait boy, kilo, yaş, kan grubu, cinsiyet gibi değişik özellikler şeklinde arttırılabilir. Matematiksel olarak yaklaşıldığında sınıflandırma yapabilmek için nitelik sayısının çok olması avantajımıza olur. Çünkü çok sayıda kullanacağımız nitelik sayesinde daha objektif ve gerçeğe daha yakın sonuçla üretiriz. O halde çalışmamızda kullandığımız veri kümeleri hakkında biraz bilgi sahibi olalım.

Pima Indians Diabetes isimli veri kümesi, 768 kayıtlı kişinin diyabet olup olmadığını söyleyen ve içeriğinde kişilere ait 8 adet niteliğin yer aldığı bir veri kümesidir. Veri kümesinin nitelikleri hamilelik durumu, kan basıncı, deri kalınlığı, insülin, beden kitle endeksi, glikoz, yaş ve kişinin soyundaki diyabet görülme oranıdır [14].

Breast Cancer Wisconsin isimli veri kümesi Wisconsin Hastanesi doktorlarından Dr. William Wolberg tarafından iğne ucu kadar büyüklükteki bir meme kanseri kitlesi taşıyan ya da taşımayan 683 kişi hakkındaki bilgilerin ve niteliklerinin bulunduğu bir veri kümesidir. Bu veri kümemizdeki niteliklerimiz şu şekildedir; kanser kitlesinin yarıçapı, çevresi, alanı, içbükeyliği, içbükey nokta sayısı, dokusu, pürüzsüzlüğü, yoğunluğu, fraktal boyutu, simetri açısıdır [14].

Vehicle Silhouettes isimli veri kümemiz 1986-1987 yıllarında JB Siebert tarafından toplanmıştır. 846 adet motorlu araçtan alınan 2 boyutlu görüntülerin 18 adet niteliğin yer aldığı iki sınıfa ayrıştırılmış dengesiz bir veri kümesidir [14].

Yeast isimli veri kümemiz de proteinlerin hücresel lokalizasyon yerlerini tahmin etmek için 1484 kişiye ait 8 adet niteliğin yer aldığı iki sınıfa ayrıştırılmış dengesiz bir veri kümesidir [14].

Çizelge 4.1.1 Veri kümelerinin dağılımı

Veriler	Numune Sayısı	Nitelikler	Çoğunluk Grup	Azınlık Grup	Dengesizlik Oranı
Pima	768	8	500	268	1.87
Wisconsin	683	9	444	239	1.86
Vehicle	846	18	628	218	2.88
Yeast	1484	8	1055	429	2.46

Çizelge 4.1.1' de gösterilen veri kümelerinin örnek dağılımı ve dengesizlik oranı dikkate alındığında, veri kümelerinin dengesiz bir dağılıma sahip olduğu görülmektedir. Bundan dolayı yeniden örnekleme yöntemi ve çalışmada önerilen diğer yöntemler kullanılarak çoğunluk ve azınlık sınıfları birbirine yaklaştırılmış, azınlık sınıfına ait örnekler yapay olarak yeniden örneklenmiş, böylece veri kümeleri dengelenmiştir.

4.2. Yeniden Örnekleme Yöntemleri

Sınıf dengesizliği problemini ortadan kaldırmak için veri düzeyinde birçok yöntem önerilmiştir [15, 16]. Birçok çalışma da ele alınan veri kümelerini dengelemek için rastgele örnek indirgeme yöntemi RUS, rastgele örnek kopya yöntemi ROS ve sentetik örnekleme yöntemi SMOTE kullanılmıştır.

RUS, çoğunluk sınıfı örneklerini rastgele kaldırarak sınıfları dengelemenin sezgisel olmayan bir yoludur. Çoğunluk sınıfından rastgele örnekleri ortadan kaldırarak en aza indirmekten oluşur. Bu eleme denetimsiz bir şekilde yapıldığından, sınıflandırıcı için faydalı örneklerin veri kümesinden çıkarılması riski vardır. Bu

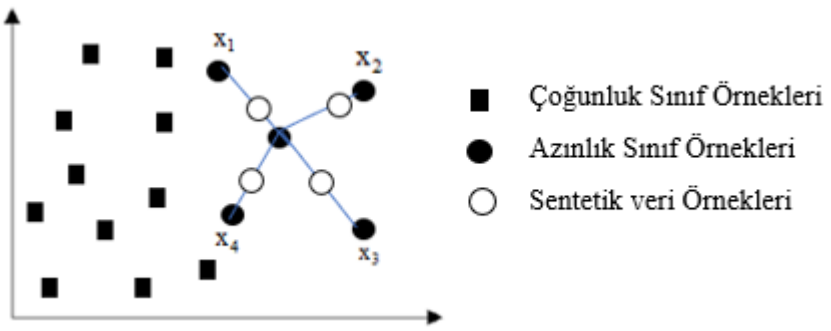
yöntem, basitliği nedeniyle sıklıkla kullanılır ve öğrenme aşamasının hızını arttırır [17].

ROS, dengesizlik problemiyle başa çıkmanın en basit ve en eski yöntemidir. Bu yöntemde sınıflandırıcı istenen orana ulaşana kadar eğitirken, azınlık sınıf örneklerini daha büyük sınıfa yaklaştırmak için rastgele kopyalama işlemi yapılır [18].

SMOTE, ham veri kümesini [4] dengelemek için aşırı örnekleme yaklaşımına sahiptir. ROS'dan farklı olarak, azınlık sınıfı örneklerinin basit bir kopyasını uygulamak yerine SMOTE, azınlık sınıfı örneklerinden sentetik veri örnekleri oluşturarak veri sınıfını dengeler. SMOTE algoritması azınlık sınıfına ait her bir örnek için, Öklid mesafesini kullanarak k tane en yakın komşu bulur. Örnek ile k en yakın komşu arasındaki fark alınır, (0,1) aralığından rastgele bir sayı (α) seçilir ve bulunan fark ile çarpılır. Aşağıdaki formül kullanılarak yeni sentetik numuneler elde edilir.

$$x_{new} = x_i + \left(x_j - x_i\right) \times \alpha \tag{4.1}$$

Burada x_i herhangi bir azınlık örneğini, x_j ise x_i nin en yakın komşularından seçilen rastgele komşusunu ve x_{new} ise yeni sentetik örneği temsil eder. Ve bu işlem istenen sayıdaki sentetik veri oluşturulana kadar devam eder. (1) ile tanıtılan işlemin x_j ise x_i nin ağırlıklı aritmetik ortalaması olduğu görülmektedir.



Şekil 4.2.1. SMOTE çalışma yöntemi

4.3. Önerilen Yöntem

Bu tez çalışmasında veri kümelerindeki dengesizlik sorununu çözmek için önceki yaklaşımlardan farklı bir yöntem kullanılmıştır. Sınıf dağılımını dengelemek için azınlık sınıfı örnekleri sentetik olarak farklı bir ortalama türü tanımlanarak elde edilmiştir. Çalışmanın önerdiği yaklaşımda, azınlık sınıfına ait örnekler, sentetik veri üretirken en yakın komşu örneklemleri temel almaktadır. En yakın komşu çiftler arasındaki mesafe, Öklid uzaklık metriği ile ölçülür. Kendi tanımladığımız yöntemle elde edilen bölgede bulunan veriler arasında ağırlıklı geometrik ortalama kullanılarak ihtiyaç duyulan sayı kadar sentetik veri üretilir. Çalışmada geliştirilen yöntem adımları şu şekildedir;

- İlk olarak, veri kümelerinde dengesizlik oranı çoğunluk sınıfındaki örnek sayısının azınlık sınıfındaki örnek sayısına oranı ile belirlenir. Bu orana dengesizlik oranı denir. Veri kümesi dengesiz ise diğer adımlar uygulanır.
- Veri kümelerini dengelemek için azınlık sınıfından yeterli sayıda sentetik veri üretilir. Bu aşamada, iki örnek $x = [x_1, x_2, ..., x_n]^T$ ve $y = [y_1, y_2, ..., y_n]^T$ arasındaki mesafeyi ölçmek için Öklid metriği kullanılır. Öklid uzaklık metriği

$$d(x, y) = \sqrt{\sum_{i=1}^{n} (x_i - y_i)^2} \tag{4.2}$$

şeklinde elde edilir.

- Veri kümesine göre anlamlı bölge genişliği bölge içinde kalan örneklerin sayısına göre tanımlandı. Algoritma bölgesi ise kaba kuvvet algoritma prensibine göre tanımlandı. Kaba kuvvet algoritması kullanılarak eksik veri nokta sayısına ulaşıncaya kadar algoritmada kullanılacak alan genişletildi. İstenen sayıdaki noktaya ulaşıldığında bölge içinde kalan tüm noktalar sentetik veri üretimi için kullanıldı.

- Yeni sentetik veri üretmek için kendi tanımladığımız bir ağırlıklı geometrik ortalama kullanıyoruz. Pozitif bileşenli vektörler için ağırlıklı geometrik ortalamayı şu şekilde tanımlıyoruz. $x = [x_1, x_2, ..., x_n]^T$, $y = [y_1, y_2, ..., y_n]^T \in \mathbb{R}_+^n$ ve $\alpha \in (0,1)$ için ağırlıklı geometrik ortalama

$$x * y = \begin{bmatrix} x_1^{\alpha} y_1^{1-\alpha} \\ x_2^{\alpha} y_2^{1-\alpha} \\ \vdots \\ x_n^{\alpha} y_n^{1-\alpha} \end{bmatrix} \tag{4.3}$$

şeklinde edilir. Yeni sentetik veri

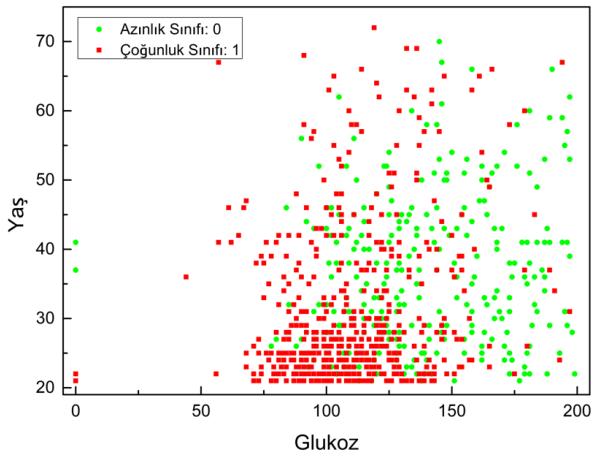
$$S_{new} = x * y \tag{4.4}$$

şeklinde bulunmuş olur. Böylece (4) numaralı formül ile istenen sayıdaki sentetik veri oluşturulana kadar ya da başka bir deyişle veri kümesi dengeli bir hal alana kadar işlem tekrarlanır.

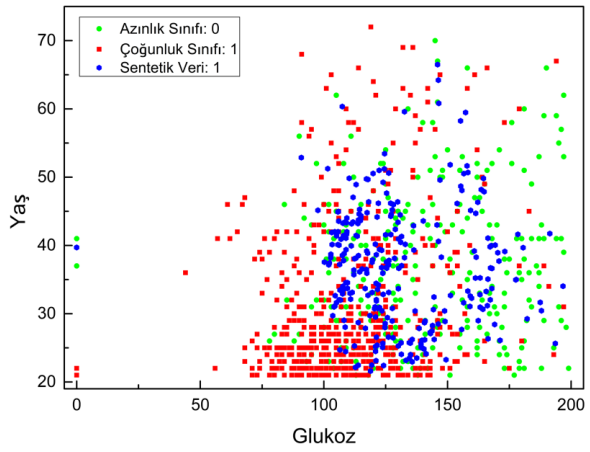
4.4. Sonuç

Bu bölümde, ağırlıklı geometrik ortalamaya dayalı olarak geliştirilen yeniden örnekleme yaklaşımının bilgisayar ortamında hazırlamış olduğumuz deneysel sonuçlarını sunuyoruz. Çalışmanın deneysel sonuçları için Şekil 4.4.1'de verilen dengesiz veri kümesi kullanılmıştır. Örneğin, Pima veri kümesinde 768 hasta örneğinden 500'ü çoğunluk (0) sınıfına ve 268'i azınlık (1) sınıfına aittir. Bu sayılara bakıldığında veri kümesinin dengesiz bir dağılıma sahip olduğu açık bir şekilde görülmektedir. Dengesizlik oranını istenen orana indirgemek ve dengeli bir veri kümesi elde etmek için yukarıda bahsedilen yeniden örnekleme yaklaşımı veri kümesi üzerine uygulanmıştır. Azınlık sınıfı örnekleri sentetik bir şekilde üretilmiş ve dengeli hale yaklaştırılmıştır. Azınlık sınıfı numunelerden 267 sentetik numune üreterek toplam 535 numune elde edilmiştir. Dengeli veri kümesi sınıfının dağılımını

daha kolay bir şekilde görselleştirmek için Yaş ve Glikoz miktarı öznitelikleri esas alınmış ve Şekil 4.4.1 ve Şekil 4.4.2'de yeniden örnekleme öncesi (ham veri) ve sonrası iki parametre baz alınarak gösterilmiştir. Bunlar iki boyutlu olacak şekilde grafik üzerinde gösterilmiştir.



Şekil 4.4.1 İşlenmemiş ham veri



Şekil 4.4.2 Yeniden örneklenen veriler

Örnek olarak kullanılan Pima veri kümesinde 500 çoğunluk ve 268 azınlıkta veri bulunmaktayken 267 sentetik veri, ağırlıklı geometrik ortalama ile üretilerek daha dengeli bir veri haline getirildi.

Şekil 4.4.2'de görüldüğü gibi azınlık sınıflarının çoğunluk grubuna yaklaştırıldığı ve azınlık örneklerinin yoğunlukta bulunduğu bölgede en yakın komşu değerlerinin örneklendiği görülmektedir.

Sınıflandırma performansını iyileştirmek için azınlık grubuna ait örnekler yeniden örneklenmiş ve tüm veri kümeleri dengelenmiştir. Çalışmamızda kullandığımız veri kümeleri ve algoritmalar Wolfram Mathematica programında Machine Learning fonksiyonları kullanılarak kodlanmıştır. Tüm sınıflandırma ve diğer işlemler için Windows Server 2019 Standard Edition kullanılmaktadır. Sunucu donanım bilgileri; Sistem Türü: x64 tabanlı PC, İşlemci(ler): Intel(R) Xenon(R) CPU E5-2640 v3 @ 2.60 GHz 2.60 GHz (2 İşlemci), Toplam Fiziksel Bellek: 63.362 MB.

Çoğunluk ve azınlık sınıfı örneklerinin birbirine yaklaştırılması sonucu dengelenen veri kümeleri, Rastgele Orman algoritması ile sınıflandırıldı. Ayrıca ham ve diğer yöntemlerle elde edilen veri kümeleri de sınıflandırılarak her sınıfın ortalama performans değerleri karşılaştırılmıştır. Rastgele Orman algoritmasının eğitim seti(3/5) ile test setini(2/5) rastgele seçmesinden dolayı bu işlemler Rastgele Orman Algoritmasında 1500 defa uygulanıp bu değerlerin ortalaması alınıp sonuçları 4 farklı performans metriği ile Doğruluk, Kesinlik, Hatırlatma, F-Ölçü ve ROC değerleri çizelge 4.4.1 ile çizelge 4.4.4 arasında özetlenmiştir. Örnekleme yöntemlerinden en iyi performanslar kalın font ile belirtilmiştir.

Çizelge 4.4.1 Pima veri kümesi sınıflandırma sonuçlarının performans değerleri

Veri Kümesi	**Doğruluk**	**Kesinlik**	**Duyarlılık**	**F-Ölçü**	**ROC**
Ham Veri	0.749	0.731	0.699	0.706	0.813
RUS	0.737	0.739	0.737	0.736	0.818
ROS	0.757	0.698	0.741	0.707	0.817
SMOTE	0.786	0.787	0.782	0.785	0.863
Bizim Yaklaşım	**0.792**	**0.793**	**0.792**	**0.792**	**0.873**

Çizelge 4.4.2 Wisconsin veri kümesi sınıflandırma sonuçlarının performans değerleri

Veri Kümesi	Doğruluk	Kesinlik	Duyarlılık	F-Ölçü	ROC
Ham Veri	0.705	0.526	0.497	0.424	0.710
RUS	0.956	0.955	0.943	0.948	0.975
ROS	0.943	0.942	0.947	0.942	0.974
SMOTE	0.961	0.962	0.945	0.953	0.974
Bizim Yaklaşım	**0.969**	**0.970**	**0.953**	**0.961**	**0.981**

Çizelge 4.4.3 Vehicle veri kümesi sınıflandırma sonuçlarının performans değerleri

Veri Kümesi	Doğruluk	Kesinlik	Duyarlılık	F-Ölçü	ROC
Ham Veri	0.972	0.972	0.957	0.963	0.996
RUS	0.961	0.962	0.961	0.961	0.991
ROS	0.977	0.918	0.984	0.946	**0.998**
SMOTE	**0.984**	0.983	**0.984**	**0.983**	**0.998**
Bizim Yaklaşım	**0.984**	**0.984**	0.983	**0.983**	**0.998**

Çizelge 4.4.4 Yeast veri kümesi sınıflandırma sonuçlarının performans değerleri

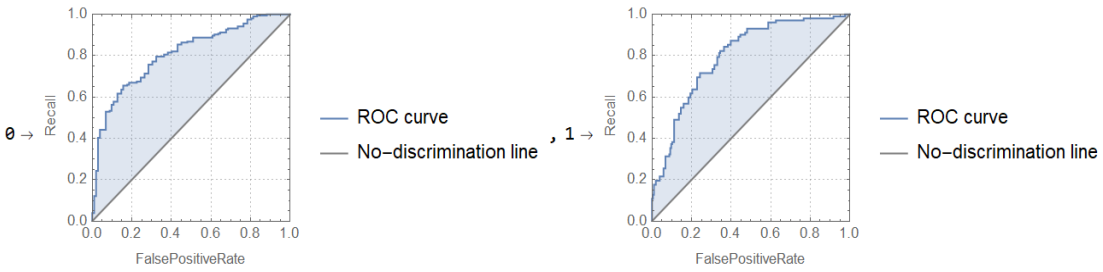
Veri Kümesi	Doğruluk	Kesinlik	Duyarlılık	F-Ölçü	ROC
Ham Veri	0.761	0.725	0.650	0.662	0.780
RUS	0.709	0.713	0.709	0.707	0.780
ROS	0.766	0.642	0.722	0.655	0.796
SMOTE	0.772	0.774	0.773	0.772	0.855
Bizim Yaklaşım	**0.775**	**0.777**	**0.775**	**0.774**	**0.861**

Çizelge 4.4.1 ve Çizelge 4.4.4 arasında gösterilen sınıflandırma sonuçlarının performans değerleri incelendiğinde önerilen yöntemimizin ham haline ve diğer yöntemlere göre daha başarılı olduğu açıkça görülmektedir. Çizelge 4.4.1' deki ham ve yeni yaklaşım ile yeniden örneklenmiş veri kümesinin performans değerlerini karşılaştırdığımızda, genel doğruluk değeri 0.749'dan 0.792'ye yükseldi. Diğer değerler, Hassasiyet 0,731'den 0,793'e, Duyarlılık 0,699'dan 0,792'ye, F-Ölçü 0,706'dan 0,792'ye ve ROC 0,813'ten 0,873'e yükseldi. Aynı şekilde yeni

yaklaşımdan elde edilen bilgiler doğrultusunda Doğruluk, Kesinlik, Duyarlılık, F-Ölçü ve ROC performans değerlerini Çizelge 4.4.2 ile Çizelge 4.4.4. arasında incelediğimizde; Çizelge 4.4.2'te sonuçlar sırasıyla 0.969, 0.970, 0.953, 0.961 ve 0.981'dir. Çizelge 4.4.3'te sonuçlar sırasıyla 0.984, 0.984, 0.983, 0.983 ve 0.998'dir. Çizelge 4.4.4'te ise sonuçlar sırasıyla 0.775, 0.777, 0.775, 0.774 ve 0.861'dir.

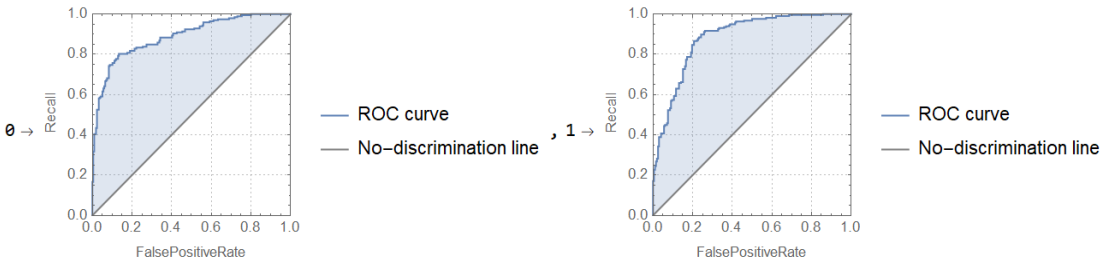
Ham ve diğer yöntemlerle kıyaslandığında yeni yaklaşımın tüm sonuç değerlerinde iyileşme vardır. Tüm veri kümeleri için Duyarlılık değerlerinde daha yüksek bir artış olması özellikle dikkat çekicidir. Ayrıca yeniden örneklenen veri kümelerinde Kesinlik, Duyarlılık ve F-Ölçü değerleri birbirine yakın sonuçlar üretmiş ve performans değerlerini birbirine yaklaştırmıştır. RUS, ROS ve SMOTE yöntemlerinin performans değerleri birbirine yakın olmasına rağmen SMOTE yöntemi diğerlerinden daha başarılı olmuştur.

Ek olarak, Pima veri kümesinin ROC eğrisi, sırasıyla ham ve yeniden örneklenmiş olarak Şekil 4.4.3 ve 4.4.4'te örnek olarak gösterilmiştir.



(a) (b)

Şekil 4.4.3 Ham verinin ROC eğrileri. (a) çoğunluk sınıfı ve (b) azınlık sınıfı



(a) (b)

Şekil 4.4.4 Yeniden örneklenen azınlık sınıfı verilerinin ROC eğrisi

Şekil 4.4.3 ve 4.4.4'de, ham ve yeniden örneklenmiş veri kümelerinin AUC değerleri, ROC eğrisi ile hesaplanmaktadır. Rastgele Orman sınıflandırmasının bir sonucu olarak, sınıf 0'ın AUC değeri 0,813'ten 0,873'e ve sınıf 1'in AUC değeri 0,814'ten 0,874'e yükselmiştir. Sınıfların AUC değerleri ve ROC eğrileri incelendiğinde, yeniden örneklenen veri kümesinin her iki sınıfta da daha başarılı olduğu görülmektedir. Ayrıca ROC eğrisinin sol üst köşeye yaklaşması, Duyarlılık oranının yüksek olduğunu, dolayısıyla eğrinin altında kalan alanın yüksek olduğunu gösterir. Buna dayanarak, yeniden örneklenen verinin sınıflandırılması daha başarılı bir ayrım yaptığından, yeniden örneklenen veri kümesinin ham veri kümesine göre eğri altında daha fazla alana sahip olduğu görülmektedir.

5. SONUÇ ve ÖNERİLER

Bu çalışmadaki amacımız, makine öğrenmesinde ya da veri madenciliğinde sıklıkla karşımıza çıkan dengesiz veri kümelerinin en verimli şekilde dengeli hale getirerek performans iyileştirilmesi sağlamaktı. Bunun için dört farklı dengesizlik oranına sahip veri kümesi kullanılarak veri kümelerinin dengelenmesi ve sınıflandırılması sonucunda daha yüksek performans değerleri sağlayan sentetik bir örnek çoğaltma yöntemi önerilmiştir. Önerilen yöntemde azınlık grubuna ait örneklerin en yakın komşuları Öklid uzaklık metriği kullanılarak belirlenmiş ve bizim tanımladığımız Ağırlıklı Geometrik Ortalama kullanılarak istenilen sayıdaki örnekte yeni sentetik veriler üretilmiş ve veri kümeleri dengeli hale getirilmiştir. Uygulanan yöntemle veri kümeleri Rastgele Orman algoritması kullanılarak sınıflandırılmıştır. Ham ve yeniden örnekleme yöntemi ile elde edilen veri kümelerinin performans değerleri karşılaştırılmış ve önerdiğimiz yöntemle elde edilen veri kümesinin hemen hemen tüm metriklerde daha başarılı sonuçlar verdiği görülmüştür. Ayrıca yeniden örneklenen veri kümesinin eğri altında ham veri kümesine göre daha fazla alana sahip olduğu ve bu da daha başarılı bir ayrım yapıldığını göstermiştir. Çalışmada önerilen yöntem ile RUS, ROS ve SMOTE yöntemleri de karşılaştırılmış ve yeni yöntemin daha başarılı olduğu görülmüştür. Çalışmada yapılan deneyler sonucunda önerilen yöntem kullanılarak dengelenen veri kümelerinin sınıflandırma sonucunda performans değerlerini arttırdığı tespit edilmiştir.

Örnek olarak hastalık teşhis modellerinin oluşturulması şu an sağlık sektörünün gelecek için en büyük yatırımlarının başında yer almaktadır. Doğru bir tahmin modeli elde etmek için eldeki veri kümesinin dengeli ve yeterli sayıda olması modelin daha doğru sonuçlar vermesi açısından çok önemlidir. Aksi takdirde yetersiz sayıdaki ve dengesiz bir örnek veri kümesi üzerinden oluşturulan modelin güvenilirliği tartışılmalıdır. Bu performans kayıpları maddi kayıplara yol açabileceği gibi sağlık sektörü gibi insan hayatı söz konusu olan sektörlerde can kayıplarına da yol açabilmektedir.

KAYNAKLAR

[1] E. Alpaydin, *Introduction to machine learning*. MIT press, 2020.

[2] D. T. Larose and C. D. Larose, *Discovering knowledge in data: an introduction to data mining*. John Wiley & Sons, 2014.

[3] K. Kowsari, K. Jafari Meimandi, M. Heidarysafa, S. Mendu, L. Barnes, and D.Brown, "Text classification algorithms: A survey," *Information,* vol. 10, no. 4, p. 150, 2019.

[4] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "SMOTE: synthetic minority over-sampling technique," *Journal of artificial intelligence research,* vol. 16, pp. 321-357, 2002.

[5] H. Han, W.-Y. Wang, and B.-H. Mao, "Borderline-SMOTE: a new over-sampling method in imbalanced data sets learning," in *International conference on intelligent computing*, 2005: Springer, pp. 878-887.

[6] H. M. Nguyen, E. W. Cooper, and K. Kamei, "Borderline over-sampling for imbalanced data classification," *International Journal of Knowledge Engineering and Soft Data Paradigms,* vol. 3, no. 1, pp. 4-21, 2011.

[7] G. E. Batista, R. C. Prati, and M. C. Monard, "A study of the behavior of several methods for balancing machine learning training data," *ACM SIGKDD explorations newsletter,* vol. 6, no. 1, pp. 20-29, 2004.

[8] M. Bayraktar, *Fonksiyonel Analiz*. Ankara: Gazi Kitabevi, 2006

[9] C. Yıldız, *Genel Topoloji*. Ankara: Gazi Kitabevi, 2005.

[10] R. Bhatia, *Positive Definite Matrices*. İn Princeton Series in Applied Mathematics. Princeton University Press, Princeton 2007

[11] Cover, T.M. and Hart, P.E., "Nearest neighbor pattern classification". *IEEE Transactions on Information Theory*, IT13(1):21–27 (1967).

[12] L. Breiman, "Random forests," *Machine learning,* vol. 45, no. 1, pp. 5-32, 2001.

[13] A. Liaw and M. Wiener, "Classification and regression by random Forest," *R news,* vol. 2, no. 3, pp. 18-22, 2002.

[14] " K. D. Repository ", *Imbalanced Dataset*,

https://sci2s.ugr.es/keel/imbalanced.php [Erişim tarihi: 15- Eylül-2021]

[15] R. Longadge and S. Dongre, "Class imbalance problem in data mining review," *arXiv preprint arXiv:1305.1707,* 2013.

[16] S. M. Abd Elrahman and A. Abraham, "A review of class imbalance problem," *Journal of Network and Innovative Computing,* vol. 1, no. 2013, pp. 332-340, 2013.

[17] S. Vluymans, *Dealing with imbalanced and weakly labelled data in machine learning using fuzzy and rough set methods*. Springer, 2019.

[18] A. Fernández, S. García, M. Galar, R. C. Prati, B. Krawczyk, and F. Herrera, *Learning from imbalanced data sets*. Springer, 2018.

[19] A. Burkov (2021), *100 Sayfada Makine Öğrenmesi*.(Çev.:A. Okatan, T. Karatekin, K. Okatan) , Ankara: Papatya Yayıncılık

[20] M. S. Shelke, P. R. Deshmukh, and V. K. Shandilya, "A review on imbalanced data handling using undersampling and oversampling technique," *International Journal of Recent Trends in Engineering and Research,* vol. 3, no. 4, pp. 444-449, 2017.

[21] I. Mani and I. Zhang, "kNN approach to unbalanced data distributions: a case study involving information extraction," in *Proceedings of workshop on learning from imbalanced datasets*, 2003, vol. 126: ICML United States.

[22] Y. Sun, M. S. Kamel, A. K. Wong, and Y. Wang, "Cost-sensitive boosting for classification of imbalanced data," *Pattern Recognition,* vol. 40, no. 12, pp. 3358-3378, 2007.

[23] L. Yijing, G. Haixiang, L. Xiao, L. Yanan, and L. Jinling, "Adapted ensemble classification algorithm based on multiple classifier system and feature selection for classifying multi-class imbalanced data," *Knowledge-Based Systems,* vol. 94, pp. 88-104, 2016.

[24] M. M. Rahman and D. N. Davis, "Addressing the class imbalance problem in medical datasets," *International Journal of Machine Learning and Computing,* vol. 3, no. 2, p. 224, 2013.

[25] M. Ercire, "Classification of short-term power quality disturbances with wavelet analysis and random forest method," Ph.D Doctoral 2019.

[26] S. Narkhede, "Understanding auc-roc curve," *Towards Data Science,* vol. 26, pp. 220-227, 2018.

KİŞİSEL BİLGİLİER

Adı Soyadı : Abdullah DAL
Doğum Yeri : Malatya
Doğum Tarihi : 21.05.1991
Medeni Hali : Evli
Yabancı Dili : İngilizce
E-posta : m.abdullah.dal@gmail.com

Eğitim Durumu

Derece	Alan	Üniversite	Mezuniyet Yılı
Lise	Fen bilimleri	Fatin Rüştü Zorlu Anadolu Lisesi	2009
Lisans	Matematik	Hacettepe Üniversitesi	2014
Lisans	İşletme	Anadolu Üniversitesi	2015
Yüksek Lisans	Matematik	Adıyaman Üniversitesi	-

İş Deneyimleri

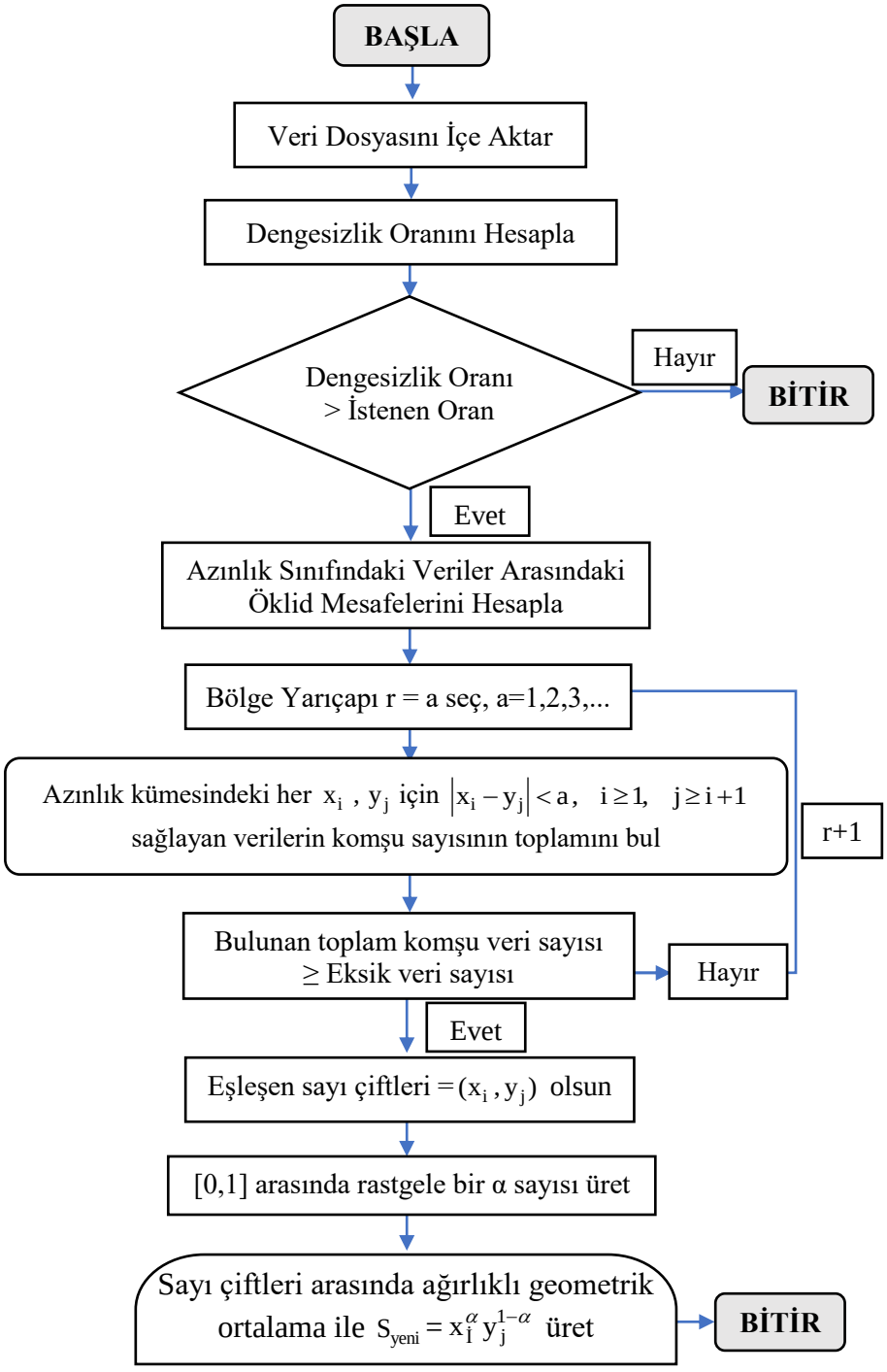
Kurum	Görevi	Başlangıç Tarihi
MEB	Matematik Öğretmeni	2016

Yayınlar

- **A. Dal**, İ. H. Gümüş, S. Güldal, M. Yavaş, "A New Resampling Approach Based On Weighted Geometric Mean For Unbalanced Data", Adıyaman Üniversitesi Fen Bilimleri Dergisi Sayı 11/Cilt 2 (2021)

EKLER

Ek 1. Algoritmamızın Akış Diyagramı



Ağırlıklı Geometrik Ortalama ile sentetik veri örnekleme yönteminin akış diyagramı