



**T.C.
İSTANBUL ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ**



DOKTORA TEZİ

**DENGESİZ VERİ KÜMELERİYLE SINIFLANDIRMA İÇİN
KÜMELEMeye DAYALI YENİ BİR HİBRİT METODOLOJİ**

Abdullah MARAŞ

Enformatik Anabilim Dalı

Enformatik Programı

DANIŞMAN

Doç. Dr. Çiğdem EROL

Ocak, 2023

İSTANBUL

Bu alıřma 13.01.2023 tarihinde ařağıdaki jüri tarafından Enformatik Anabilim Dalı Enformatik Programında Doktora Tezi olarak kabul edilmiştir.

Tez Jürisi

Do. Dr. iğdem EROL (Danışman)
İstanbul Üniversitesi
Enformatik Anabilim Dalı

Prof. Dr. Adem KARAHOCA
MEF Üniversitesi
Mühendislik Fakültesi

Do. Dr. Murat GEZER
İstanbul Üniversitesi
Enformatik Bölümü

Dr. Öğr. Üyesi Yalın ÖZKAN
İstinye Üniversitesi
İktisadi, İdari ve Sosyal Bilimler Fakültesi

Do. Dr. Emre AKADAL
İstanbul Üniversitesi
İktisat Fakültesi

- **İntihal Programı Beyanı**

20.04.2016 tarihli resmi gazetede yayımlanan Lisansüstü Eğitim ve Öğretim Yönetmeliğinin 9/2 ve 22/2 maddeleri gereğince; Bu Lisansüstü teze, İstanbul Üniversitesi'nin abonesi olduğu intihal yazılım programı kullanılarak Fen Bilimleri Enstitüsü'nün belirlemiş olduğu ölçütlere uygun rapor alınmıştır.

- **Tezden Üretilmiş Yayınların Künye Bilgileri**

MARAŞ, A. ve EROL, Ç., 2022, *Emerging Trends in Classification with Imbalanced Datasets: A Bibliometric Analysis of Progression*, Bilişim Teknolojileri Dergisi, 15(3), 275-288.

ÖNSÖZ

Bu tez çalışmasının başlangıcından tamamlanmasına kadar beni destekleyen, motive eden, tüm zor günlerimde arkamda olduğunu hissettiren ve yeri geldiğinde hoş gören danışmanım Doç. Dr. Çiğdem EROL'a çok teşekkür ederim. Ayrıca doktora tez izleme komitemde yer alan Prof.Dr. Sevinç GÜLSEÇEN ve Prof. Dr. Adem KARAHOCA'ya destekleri ve çalışmama katkıları için teşekkür etmek istiyorum. Alandaki gelişimime katkı sağlayan ve güvenini eksik etmeyen Doç.Dr. Elif KARTAL'a teşekkürlerimi sunarım.

Bu uzun süreçte benden desteklerini esirgemeyen arkadaşlarım Erhan YUMUK, İlkin ECEM EMRE, Çiğdem YAZILI , Didem YAZILI ve Fırat SAYGIN'a da teşekkür etmek isterim.

Son olarak tezimi beni bugünlere getiren ve her zaman destekleyen canım Annem Sevim MARAŞ'a ithaf ederim.

Ocak, 2023

Abdullah MARAŞ

İÇİNDEKİLER

	Sayfa No
ÖNSÖZ	iv
İÇİNDEKİLER	vii
ŞEKİL LİSTESİ	viii
TABLO LİSTESİ	x
SİMGE VE KISALTMA LİSTESİ	xi
ÖZET	xiii
SUMMARY	xv
1. GİRİŞ	1
1.1. ÇALIŞMANIN AMACI	4
1.2. ÇALIŞMANIN KAPSAMI VE ÖNEMİ	5
1.3. ÇALIŞMANIN SINIRLILIKLARI	5
2. GENEL KISIMLAR	6
2.1. BİBLİYOMETRİK ANALİZ	6
2.1.1. Yıllık Bilimsel Üretim	7
2.1.2. Ortalama Toplam Alıntı	8
2.1.3. Ülkelerin Dengesiz Veri Kümeleri Probleminde Çalışmaları	9
2.1.4. En Çok Atıf Alan Yayınlar	10
2.1.5. En Sık Kullanılan Anahtar Kelimeler	13
2.1.6. Anahtar Kelime Birliktelikleri	16
2.1.7. En Çok Atıf Alan ve En Çok Katkıda Bulunan Yazarlar	16
2.1.8. En Çok Alıntı Yapılan Kaynaklar	19
2.1.9. Tematik Harita	19
2.2. DENGESİZ VERİ KÜMELERİNE İLİŞKİN YAPILAN TÜRKİYE'DEKİ TEZLER	20
2.3. MAKİNE ÖĞRENMESİ	22
2.3.1. Danışmanlı Öğrenme	24
2.3.2. Danışmansız Öğrenme	25

2.3.3. Yarı Danışmanlı Öğrenme	27
2.3.4. Pekiştirmeli Öğrenme	27
2.4. MAKİNE ÖĞRENMESİ YÖNTEMLERİ	28
2.4.1. K-En Yakın Komşu Algoritması	28
2.4.2. Destek Vektör Makineleri	30
2.4.3. Rastgele Orman Algoritması	33
2.4.4. K-ortalamlar Kümeleme Yöntemi	35
2.4.5. Bulanık C-ortalama Kümeleme Yöntemi	36
2.5. MODEL PERFORMANS ÖLÇÜLERİ	37
2.5.1. Karmaşıklık Matrisi	37
2.5.2. Doğruluk	38
2.5.3. Duyarlılık	39
2.5.4. Belirleyicilik	39
2.5.5. Kesinlik	39
2.5.6. F-ölçütü	40
2.5.7. Alıcı İşlem Karakteristik Eğrileri	40
2.5.8. Eğri Altında Kalan Alan	42
2.6. DENGESİZ VERİ KÜMELERİ ÇÖZÜMLERİ	42
2.6.1. Rastgele Aşağı Örnekleme	42
2.6.2. Sentetik Azınlık Yukarı Örnekleme	43
2.6.3. Borderline-SMOTE	44
3. MALZEME VE YÖNTEM	45
3.1. VERİ KÜMELERİ	45
3.1.1. Pima Indians Diyabet Veri Kümesi	46
3.1.2. KDD Cup 1999 Veri Kümesi	47
3.1.3. Credit Card Fraud Detection Veri Kümesi	49
3.2. VERİ ÖN İŞLEME	49
3.2.1. Pima Indians Diyabet Ön İşleme	49
3.2.2. KDD Cup 1999 Ön İşleme	51
3.2.3. Credit Card Fraud Detection Ön İşleme	52
3.3. YÖNTEM	52
4. BULGULAR	55

5. TARTIřMA VE SONUÇ	71
KAYNAKLAR	77
ÖZGEÇMİř	87



ŞEKİL LİSTESİ

	Sayfa No
Şekil 2.1: Bibliyometrik analiz metodolojik çerçeve [39].	7
Şekil 2.2: 1957-2021 yılları arası dengesiz veri kümeleri üzerine Scopus yayın sayıları [40].	8
Şekil 2.3: Yıllık ortalama toplam atıf sayısı [40].	8
Şekil 2.4: Ülkelere göre dengesiz veri kümesi problemi yayın miktarları [40].	9
Şekil 2.5: Dünya haritası üzerinde ülkelerin işbirliği [40].	10
Şekil 2.6: Dengesiz veri kümelerinde kullanılan anahtar kelimelerin sıklığı [40].	15
Şekil 2.7: Zamansal kümülatif kelime dinamiği [40].	15
Şekil 2.8: Anahtar kelime birliktelikleri ağı [40].	16
Şekil 2.9: Yazarların zaman içindeki üretkenlik grafiği [40].	17
Şekil 2.10: Yazarların yayınlarının tarihsel ilerleyişi [40].	18
Şekil 2.11: Yayın yapılan kaynakların alınan atıf sayıları grafiği [40].	19
Şekil 2.12: Tematik harita [40].	20
Şekil 2.13: Destek vektör makineleri doğrusal aşırıdüzlem örneği [114].	32
Şekil 2.14: Farklı derecelerden polinomial çekirdek fonksiyonları [115].	33
Şekil 2.15: Alıcı işlem karakteristik eğrisi [110, 127].	41
Şekil 3.1: Pima Indians diyabet veri kümesi değişken dağılımları.	47
Şekil 3.2: Pima Indians diyabet veri kümesi korelasyon matrisi.	50
Şekil 3.3: Veri temizleme sonrası KDD Cup 1999 veri kümesi korelasyon matrisi.	51
Şekil 3.4: Geliştirilen FuzzyCSampling yöntemi.	53

TABLO LİSTESİ

Sayfa No

Tablo 2.1: En çok atıf alan 20 makalenin özet bilgileri [40].	14
Tablo 2.2: Yayınlarda kullanılan en popüler yazar anahtar kelimeleri [40].	14
Tablo 2.3: En çok atıf alan yazarlar [40].	17
Tablo 2.4: Karmaşıklık matrisi.	38
Tablo 3.1: Veri ön işleme öncesi veri kümelerinin özellikleri.	46
Tablo 3.2: Pima Indians diyabet veri kümesi değişkenleri.	46
Tablo 3.3: KDD Cup 1999 veri kümesi değişkenleri [133].	48
Tablo 3.4: Değişkenlerdeki eksik olmayan kayıt sayıları.	50
Tablo 4.1: Veri kümelerinin özellikleri.	55
Tablo 4.2: Pima Indians dengesiz veri kümesi ile kurulan model performansı.	56
Tablo 4.3: Pima Indians veri kümesi rastgele aşağı örnekleme ile kurulan model performansı.	56
Tablo 4.4: Pima Indians veri kümesi SMOTE yöntemi ile kurulan model performansı.	57
Tablo 4.5: Pima Indians veri kümesi Borderline-SMOTE model performansı.	57
Tablo 4.6: Pima Indians veri kümesi FuzzyCSampling yöntemi model performansı.	58
Tablo 4.7: Pima veri kümesi FuzzyCSampling+Borderline-SMOTE performansı.	58
Tablo 4.8: Pima veri kümesi k-ortalamalar kümeleme-FuzzyCSampling model performansı.	59
Tablo 4.9: Pima veri kümesi k-ortalamalar kümeleme-FuzzyCSampling+ Borderline-SMOTE.	60
Tablo 4.10: KDD Cup 1999 dengesiz veri kümesi ile kurulan model performansı.	61
Tablo 4.11: KDD Cup 1999 veri kümesi rastgele aşağı örnekleme ile kurulan model performansı.	61
Tablo 4.12: KDD Cup 1999 veri kümesi SMOTE yöntemi ile kurulan model performansı.	62
Tablo 4.13: KDD Cup 1999 veri kümesi Borderline-SMOTE yöntemi performansı.	63
Tablo 4.14: KDD Cup 1999 veri kümesi FuzzyCSampling yöntemi performansı.	63
Tablo 4.15: KDD veri kümesi FuzzyCSampling+Borderline-SMOTE yöntemi performansı.	64
Tablo 4.16: KDD veri kümesi k-ortalamalar kümeleme-FuzzyCSampling yöntemi performansı.	65
Tablo 4.17: KDD veri kümesi k-ortalamalar kümeleme-FuzzyCSampling +BorderlineSMOTE.	65
Tablo 4.18: Credit Card Fraud dengesiz veri kümesi model performansı.	66
Tablo 4.19: Credit Card Fraud veri kümesi rastgele aşağı örnekleme model performansı.	67
Tablo 4.20: Credit Card Fraud veri kümesi SMOTE yöntemi performansı.	67

Tablo 4.21: Credit Card Fraud veri kümesi Borderline-SMOTE yöntemi performansı.	68
Tablo 4.22: Credit Card Fraud veri kümesi FuzzyCSampling yöntemi performansı.	68
Tablo 4.23: Credit Card Fraud veri kümesi FuzzyCSampling+Borderline-SMOTE performansı. ...	69
Tablo 4.24: Fraud veri kümesi k-ortalamalar kümeleme-FuzzyCSampling performansı.	70
Tablo 4.25: Fraud veri kümesi k-ortalamalar kümeleme-FuzzyCSampling+Borderline-SMOTE...	70
Tablo 4.26: Pima veri kümesi ile ilgili çalışmalarla model performansının karşılaştırılması.	71
Tablo 4.27: Fraud veri kümesi ile ilgili çalışmalarla model performansının karşılaştırılması.	71



SİMGE VE KISALTMA LİSTESİ

Kısaltmalar	Açıklama
ACC	: Doğruluk (<i>Accuracy</i>)
AUC	: Alıcı İşlem Karakteristik Eğrisi Altında Kalan Alan
BI	: İş Zekası (<i>Business Intelligence</i>)
BP	: Geri Yayılım (<i>Back Propagation</i>)
K-NN	: K-en Yakın Komşu (<i>K-nearest Neighbors</i>)
KCN	: Anahtar Sözcüklerin Birliktelik Ağı (<i>Keywords Co-occurrence Network</i>)
FN	: Yanlış Negatif (<i>False Negative</i>)
FP	: Yanlış Pozitif (<i>False Positive</i>)
GBS	: Guillain-Barre Sendromu (<i>Guillain-Barre Syndrome</i>)
GM-NSC	: Geometrik Ortalama En Yakın Küçülen Merkez
IoT	: Nesnelerin İnterneti (<i>Internet of Things</i>)
IR	: Dengesizlik Derecesi/Oranı (<i>Imbalanced Ratio</i>)
MCP	: Çoklu Ülke Yayınları (<i>Multiple Country Publications</i>)
MTS	: Mahalanobis-Taguchi Sistemi (<i>Mahalanobis-Taguchi System</i>)
PCA	: Temel Bileşenler Analizi (<i>Principal Component Analysis</i>)
PSO	: Parçacık Sürü Optimizasyonu (<i>Particle Swarm Optimization</i>)
QDA	: Kuadratik Diskriminant Analizi (<i>Quadratic Discriminant Analysis</i>)
ROC-CURVE	: Alıcı İşletim Karakteristik Eğrisi (<i>Receiver Operating Characteristic</i>)
SCP	: Tek Ülke Yayınları (<i>Single Country Publications</i>)
SDC	: Smote With Different Costs
SMOTE	: Sentetik Azınlık Aşırı Örnekleme Tekniği
SVM	: Destek Vektör Makineleri (<i>Support Vector Machines</i>)
TN	: Doğru Negatif (<i>True Negative</i>)
TP	: Doğru Pozitif (<i>True Positive</i>)
TNR	: Doğru Negatif oranı (<i>True Negative Rate</i>)
TPR	: Doğru Pozitif oranı (<i>True Positive Rate</i>)
UCI	: Kaliforniya Irvine Üniversitesi Makine Öğrenimi Platformu

ÖZET

DOKTORA TEZİ

DENGESİZ VERİ KÜMELERİYLE SINIFLANDIRMA İÇİN KÜMELEMeye DAYALI YENİ BİR HİBRİT METODOLOJİ

Abdullah MARAŞ

İstanbul Üniversitesi

Fen Bilimleri Enstitüsü

Enformatik Anabilim Dalı

Danışman: Doç. Dr. Çiğdem EROL

Makine öğrenmesi modellerinde istenilen performans değerlerine ulaşamamasının bir nedeni de dengesiz veri kümelerinin kullanılmasıdır. Dengesiz veri kümeleri hedef değişkenlerin dağılımlarının eşit olmadığı veri kümelerine verilen addır. Makine öğrenmesi model kurulumu sırasında kullanılan algoritmalar, veri kümesindeki hedef değişkenlerin dağılımının eşit olduğu varsayımı altında daha iyi performans göstermektedirler. Bu nedenle makine öğrenmesi modellerini oluştururken probleme yönelik sunulan aşağı örneklem, yukarı örneklem veya hibrit yaklaşım çözümlerinden biri uygulanmaktadır. Bu tez kapsamında öncelikle dengesiz veri kümeleri problemi ve çözümleri ile ilgili bir bibliyometrik analiz gerçekleştirilmiş olup sonrasında problem çözümü için FuzzyCSampling olarak adlandırılan hibrit bir yaklaşım geliştirilmiştir. FuzzyCSampling yaklaşımı kümeleme ve veri örneklem yöntemlerinin birlikte kullanarak hedef değişkenlerin eşitsiz dağılımlarının oranının azalmasını sağlamıştır. Önerilen yaklaşım farklı dengesizlik oranına ve büyüklüklere sahip olan *Pima Indians Diyabet*, *KDD Cup 1999* ve *Credit Card Fraud Detection* veri kümeleriyle model oluşturulması için kullanılıp model performanslarındaki değişim incelenmiştir. Model değerlendirme ölçütleri olarak doğruluk, duyarlılık, kesinlik, F-ölçütü ve AUC kullanılmıştır. Model performans değerlendirmesinde veri kümesi %70 eğitim %30 test ayrılarak kullanılmıştır. Ayrıca modeller rastgele orman algoritması, k-en yakın komşu ve destek vektör makineleri kullanılarak Python dilinde gerçekleştirilmiştir. FuzzyCSampling yönteminin başarısının karşılaştırılması amacıyla

yaygın kullanılan çözümler olan rastgele aşağı örnekleme, SMOTE, Borderline-SMOTE ve herhangi bir çözüm kullanılmayan durum için modeller kurulmuştur.

Pima Indians Diyabet veri kümesi ile oluşturulan modeller kıyaslandığında, k-en yakın komşu algoritması ve FuzzyCSampling yaklaşımı ile oluşturulan modelin performansının (doğruluk:0,885; AUC: 0,872 ve F-ölçütü:0,846) diğer modellerden daha iyi olduğu gözlemlenmiştir. *KDD Cup 1999* veri kümesi ile oluşturulan modellerin arasında da rastgele orman algoritması ve FuzzyCSampling yaklaşımının uygulanması durumunda diğer modellerden daha iyi performans sonuçları (doğruluk:0,999; kesinlik: 0,993 ve F-ölçütü:0,978) elde edildiği gözlemlenmiştir. *Credit Card Fraud Detection* veri kümesiyle oluşturulan modeller karşılaştırıldığında FuzzyCSampling yaklaşımının rastgele orman algoritması birlikte uygulanmasının performansı (AUC:0,941; kesinlik: 0,970 ve F-ölçütü:0,932) iyileştirdiği gözlemlenmiştir. Bu tez kapsamında dengesiz veri kümeleri problemine çözüm olarak sunulan FuzzyCSampling yaklaşımının ikili sınıflandırma problemlerinde iyi bir alternatif çözüm olduğu görülmüştür.

Ocak 2023, 102 sayfa.

Anahtar kelimeler: İkili sınıflandırma, dengesiz veri kümeleri, makine öğrenmesi, bulanık c-ortalama kümeleme, bibliyometrik analiz.

SUMMARY

Ph.D. THESIS

A NOVEL HYBRID METHODOLOGY BASED ON CLUSTERING FOR CLASSIFICATION WITH IMBALANCE DATASETS

Abdullah MARAŞ

İstanbul University

Institute of Graduate Studies in Sciences

Department of Informatics

Supervisor: Assoc. Prof. Dr. Çiğdem EROL

The usage of imbalanced datasets would be another reason that prevents machine learning models from producing the desired performance results. Datasets that have an uneven distribution of the target variables are known as imbalanced datasets. The performance of algorithms used during building a machine learning model is better when the distribution of the target variables in the dataset is equal. For this reason, when building machine learning models, solutions for imbalanced datasets such as undersampling, oversampling, or hybrid approaches are applied to the problem. Within the scope of this thesis, a bibliometric analysis of the imbalanced datasets problem and its solutions was performed first, followed by the proposal of a new hybrid approach based on clustering for the imbalanced dataset problem. The FuzzyCSampling approach has decreased the rate of uneven distribution of the target variables by combining data sampling and clustering techniques. The proposed approach is used to create models with Pima Indians Diabetes, KDD Cup 1999 and Credit Card Fraud Detection datasets which have different imbalanced ratios and sizes, and the change in model performances is examined. The results were analyzed using the evaluation metrics accuracy, recall, precision, F1-score, and AUC. The dataset was divided into 70% training and 30% testing for the model performance evaluation. Moreover, support vector machines, k-nearest neighbor, and the random forest technique were used to build the models in Python. Models were created for the widely used solutions random undersampling, SMOTE, Borderline-SMOTE, and the absence of any solution in order to compare the effectiveness of the FuzzyCSampling approach.

In a comparison of the models developed using the Pima Indians Diabetes dataset, it was found that the model developed using the k-nearest neighbor technique and the FuzzyCSampling approach performed (accuracy:0,885; AUC: 0,872 and F1-score:0,846) better than the other models. Among the models created with the KDD Cup 1999 dataset, it was also observed that better performance results (accuracy:0,999; precision: 0,993 and F1-score:0,978) were obtained when the random forest algorithm and the FuzzyCSampling approach were applied. When the models built with the Credit Card Fraud Detection dataset were compared, it is observed that utilizing the FuzzyCSampling approach in conjunction with the random forest algorithm improves the performance (AUC:0,941; precision: 0,970 and F1-score:0,932). The FuzzyCSampling strategy, which is put forth in this thesis as a solution for the issue of imbalanced datasets, was found to be a good alternative solution for binary classification problems.

January 2023, 102 pages.

Keywords: Binary classification, imbalanced datasets, machine learning, fuzzy c-means, bibliometric analysis.

1. GİRİŞ

Teknolojideki son gelişmeler veri çağıının kapılarını açıp, iş yapış şekillerini değiştirmeye başlamıştır. Donanım teknolojisindeki ilerlemeler, cihaz işleme kapasitelerindeki ve bunun sonucunda depolanan ham veri hacmindeki artış, makine öğrenimi algoritmalarının daha etkin bir şekilde kullanılmasını sağlamıştır. Şirketler, rakiplerinin önünde kalabilmek için makine öğrenimi algoritmalarını kendi işlerine uyarlama yarışına girmiştir. Nesnelerin İnterneti (IoT), sosyal medya, e-ticaret, çevrimiçi bankacılık ve yoğun veri üreten teknolojik ürünlerin kullanımının artması, yalnızca depolanan verilerin genişlemesini değil, bununla birlikte veri biliminde de büyük ilerlemeleri beraberinde getirmektedir. Makine öğreniminin endüstriler arasında yaygın olarak kullanılması, başarılı modellerin oluşturulmasına engel teşkil edebilecek gerçek sorunlara çözümlerin ortaya çıkmasını da hızlandırmaktadır. İstatistik temellerinden başlayarak verilerden içgörü kazanmaya yönelik yaklaşımlar, makine öğreniminden derin öğrenmeye kadar çeşitli çalışma alanlarının gelişimine yol açmaktadır.

Yapay zeka alanında en çok çalışılan konulardan biri, geniş bir uygulama yelpazesi içeren sınıflandırma problemleridir. Bir danışmanlı öğrenme yaklaşımı olan sınıflandırma problemleri, önceden tanımlanmış hedef değişkenlerden birini tahmin etmeyi amaçlar [1]. Sınıflandırmanın ana amacı, girdi değişkenleri ile hedef değişkenler arasındaki ilişkiyi aramaktır [2]. Dolandırıcılık tespiti [3], kredi skorlama [4], tıbbi tanı koyma [5–7], müşteri kayıp tahmini [8] ve duygu analizi [9] sınıflandırma problemlerine konu olan bazı ana örneklerdir. İkili sınıflandırma (*binary classification*), çok sınıflı sınıflandırma (*multiclass classification*) ve çok etiketli sınıflandırma (*multi-label classification*) olmak üzere üç tür sınıflandırma vardır. İkili sınıflandırma, bir veri kümesindeki öğelerin iki gruptan birine atama problemine verilen addır [10]. Çok sınıflı sınıflandırmada ise veri kümesindeki her bir öğenin ikiden fazla hedef sınıftan birine atanması hedeflenir [11]. Çok etiketli sınıflandırma, ikili ve çok sınıflı sınıflandırmadan farklı olarak, her veri noktası için bir veya daha fazla sınıf etiketinin yansıtıldığı problemleri modellemek için kullanılır [12]. Rastgele orman (*random forest*), karar ağaçları (*decision trees*), destek vektör makineleri (*support vector machines*), sinir ağları (*neural networks*), lojistik regresyon (*logistic regression*) ve naive bayes gibi çok sayıda algoritma, sınıflandırma problemlerini çözmek için uygulanmıştır [13].

Bu algoritmalar kullanılarak geliştirilen modeller, veri kümesinin hedef değişkenin dengeli bir dağılımına sahip olması durumunda başarılı sonuçlar vermektedir. Bununla birlikte, dengesiz veri kümesi olarak adlandırılan hedef değişkenlerin dağılımlarının eşit olmadığı koşullarda, modellerin performansının nispeten daha düşük olduğu da gözlenmektedir [14]. Gerçek dünyadaki makine öğrenimi uygulamalarında dengesiz veri kümeleriyle çalışmak kaçınılmaz olabilmektedir. Dolandırıcılık tespiti [15], istenmeyen e-posta tespiti [16], hastalık teşhisi [17] ve metin sınıflandırması [18], dengesiz veri kümeleriyle sıklıkla karşılaşılan problem türleridir.

Dengesiz veri kümelerinde hedef değişkenlerden biri çoğunlukla diğer hedef değişkenden çok daha fazladır [19]. Modeller çoğunluk hedef değişken örneklerini tahmin etmeye meyilli olduğundan, dengesiz veri kümeleri durumunda yüksek bir doğruluk puanı elde edilebilir [20]. Bununla birlikte, model değerlendirme ölçütlerinden olan doğruluğun yüksek olması her zaman iyi bir modelin göstergesi değildir. Yüksek doğruluk değeri çoğunluk hedef değişkeninin iyi tahmin edilmesinden kaynaklanırken, dengesiz veri kümesi problemlerinde tahmin edilmesi daha önemli ve zor olan azınlık sınıflarının tahmin edilmesinde aynı ölçüde başarı göstermeyebilir. Örneğin, bir dolandırıcılık tespit problemi için daha az dolandırıcılık örneği olmasına rağmen dolandırıcılık işlemlerinin doğru tahmin edilmesi önemlidir [3]. Dengesiz veri kümelerinin model üzerindeki olumsuz etkileri yadsınamaz bir gerçektir. Konunun önemi ve konuya olan ilgi, AAAI 2000 [21] ve ICML 2003 [22] konferanslarından sonra artış trendine girmiştir. Konferanslardaki çalıştaylarda, problem tanımından çözümlere ve dengesiz veri kümelerinin değerlendirilmesine kadar çeşitli konular tartışılmıştır. Dengesiz veri kümeleri, günlük uygulamalarda her zaman farklı derecelerde hedef değişken dengesizliğine sahiptir. Bir sınıflandırma sorununun dengesizlik derecesi/oranı (*imbalanced ratio*) literatürde IR olarak gösterilir ve problemin zorluğunu ifade etmede kullanılmaktadır. Dengesizlik oranı, çoğunluk hedef değişkeninin (*majority target variable*) azınlık hedef değişkenine (*minority target variable*) oranı olarak tanımlanmaktadır [19, 23–25]. Gerçek dünya uygulamalarında, dengesizlik oranı 100:1, 1000:1 kadar veya daha büyük olabilir [19]. Dengesizlik oranının makine öğrenmesi algoritmalarının performansı üzerinde büyük etkisi olduğundan dolayı bu problemin çözümünde kullanılmak üzere farklı yaklaşımlar geliştirilmiştir.

Dengesiz veri kümelerinin nedeni içsel (*intrinsic*) veya dışsal (*extrinsic*) olabilir [24]. İçsel dengesiz veri kümeleri sorunları, sorunun doğası gereği ortaya çıkar. Tıbbi sınıflandırma

problemlerinin çoğu, içsel dengesiz veri kümeleri problemlerine örnek gösterilebilir. Bu tür veri kümelerinde, her zaman diğer değişkenlerden daha az örneği olan bir hedef değişken grubu yer almaktadır [26]. Aksine dışsal kaynaklı nedenlerle oluşan dengesiz veri kümeleri, yetersiz veri toplama süresi veya veri depolama sırasındaki hataların sonucu olarak ortaya çıkabilir. Dengeli bir veri kümesi problemi, belirtilen sebeplerden dolayı dengesiz bir veri kümesi problemine dönüşebilir. Gerçek zamanlı akan verileriyle sınıflandırma yaygın bir dışsal kaynaklı veri kümeleri sorunudur [27].

Dengesiz veri kümelerinin model performansı üzerindeki etkilerinin yanı sıra, model performansında sorun yaratabilecek diğer faktörler hakkında da araştırmalar mevcuttur [19, 28–30]. Hedef sınıflar içinde oluşan alt kümeler (*small disjuncts*) [29, 30], hedef sınıf değişkenlerinin bölgelerinin çakışması (*class overlapping*) [31] ve veri kümelerindeki [32, 33] gürültülü örnekler, model performansını engelleyen diğer faktörler arasındadır.

Farklı endüstrilerdeki veri kümelerinin farklı doğası ve dengesiz veri kümelerinin son yıllarda çok çalışılan bir konu olduğu gerçeği göz önüne alındığında, dengesiz veri kümeleriyle başa çıkmak için araştırmacılar tarafından birçok strateji geliştirilmiştir. Önerilen yaklaşımlar veri ön işleme, topluluk teknikleri ve maliyete duyarlı öğrenme olmak üzere üç ana başlıkta gruplanabilir [34]. Veri seviyesi çözümleri olarak da bilinen veri ön işleme çözümleri, hedef sınıfların dağılımını dengelemeye çalışır. Bu grupta yer alan çözümlerin temel avantajı seçilen sınıflandırıcıdan etkilenmiyor olmasıdır [34]. Aşağı örnekleme (*undersampling*) ve yukarı örnekleme (*oversampling*), bu alana ilgi gösterenler tarafından kullanılan en yaygın iki veri ön işleme metodolojisidir. Aşağı örnekleme temelli teknikler, veri kümesinden örnekleri kaldırarak veri kümelerini dengelemeye çalışırken, yukarı örnekleme temelli yöntemler, mevcut örneklerin kopyalanması veya mevcut örneklerden yeni örneklerin oluşturulması yoluyla veri kümelerini dengelemeye çalışır. Bununla birlikte, hem aşağı örnekleme hem de yukarı örnekleme tabanlı yöntemlerin kendi zorlukları vardır. Bu nedenle araştırmacılar hem aşağı örnekleme hem de yukarı örnekleme tekniklerini birlikte kullanan hibrit çözümler önermişlerdir [35]. Topluluk teknikleri, dengesiz veri kümeleriyle başa çıkmak için maliyete duyarlı öğrenme veya veri ön işleme yaklaşımlarından yararlanır [34, 36]. Topluluk yöntemleri, model performansını artırmak için birkaç sınıflandırıcının birleşimi olan topluluk sınıflandırıcılarından da yararlanabilir. Dengesiz veri kümelerini ele alan son çözüm grubu, yanlış sınıflandırma maliyetlerini değerlendirerek model performansını artıran maliyete duyarlı öğrenmedir [37]. Alanda

alışan uzmanlar ve akademisyenler iin ounlukla dengesiz veri kmelerindeki azınlık sınıfı iyi tahminlemek daha onemlidir. Sonu olarak, azınlık sınıfı, yanlış sınıflandırma maliyetinin ve cezaların belirlenmesinde kritik oneme sahip olmaktadır.

Son yıllarda, araştırmacılar, model performansını iyileştirmek, aşığı rneklemeye ve yukarı rneklemeye yaklaşımlarının olumsuz taraflarını azaltmak iin stratejileri birleştirmeye ynelim gstermektedir. Dengesiz veri kmeleriyle alışma talebi arttıka, bu alanda onerilen zmlerin sayısı da artmaktadır. Bu alışmada, bilgi kaybını azaltarak model performanslarını artırmak iin bulanık c-ortalama kmeleme (*fuzzy c-means*) ve veri n ileme yaklaşımlarını birleştiren bir topluluk yaklaşımı sunulmuştur.

Bu tez alışması Genel Kısımlar, Malzeme ve Yntem, Bulgular, Tartışma ve Sonu olmak zere 4 blm ile devam etmektedir. Genel Kısımlarda ilk nce dengesiz veri problemiyle ilgili alışmaların zetini gstermek iin bir bibliyometrik analiz gerekleştirilmişt ve sonuları yorumlanmıştır. Problemin tanımı ve yaklaşımlar iin bilgiler zetlendikten sonra, tez kapsamında onerilen yaklaşımın kavranması iin makine ğrenmesi konusunda bilgiler paylaşılmıştır. Malzeme ve Yntem blmnde ise onerilen yntemin test edilmesinde kullanılacak veri kmelerine yer verilmiştir. onerilen yntemin blok diyagramı ve aıklamaları da kullanılan veri kmelerinin ayrıntılarının ardından bu blmde paylaşılmıştır. onerilen yaklaşıma dayalı kurulan modellerin sonuları Bulgular blmnde paylaşılmıştır. onerilen yaklaşımın deęerlendirilmesi ve mevcut yaklaşımlar ile karşılaştırmalı tartışılması Tartışma ve Sonu blmnde sunulmuştur.

1.1. ALIŐMANIN AMACI

Bu tez alışmasındaki ama; makine ğrenmesi algoritmalarında performans azalmasına sebebiyet veren dengesiz veri kmesi problemi iin kmelemeye dayalı, aşığı rneklemeye (*undersampling*) ve yukarı rneklemeye (*oversampling*) yaklaşımlarını kullanan hibrit yeni bir yaklaşım sunmaktır. Dengesiz veri kmeleriyle model kurulumunda kullanılan yaklaşımlar her veri kmesi iin aynı başarılı sonucu vermemektedir. Bu alışma kapsamında nceki onerilen metotlardan da faydalanılarak alternatif bir zm nerisine yer verilmiştir. Araştırma sonucunda onerilen yntemin farklı endstrelere ait veri kmeleri iin oluşturulan makine ğrenmesi modelleri iin performans geliştirmelerine bakılmıştır.

1.2. ÇALIŞMANIN KAPSAMI VE ÖNEMİ

Önerilen yöntem ile problem doğasından kaynaklı veya veri toplama sorunlarından dolayı oluşan makine öğrenmesi modellerinde düşük performans çıktılarının önüne geçilmesi hedeflenmiştir. Kümelemeye dayalı önerilen yeni hibrit yaklaşım ile aşağı örnekleme yöntemlerinde meydana gelebilecek bilgi kaybının önüne geçilmek hedeflenmiştir. Aynı zamanda kullanılan kümeleme yöntemi ile sınır bölgesinde yer alan örneklerin sınıflandırma başarısının arttırılması amaçlanmıştır. Önerilen yöntemin yanı sıra dengesiz veri kümeleri üzerine ilk bibliyometrik analiz gerçekleştirilerek problemin tanımından en çok kullanılan yöntemlere kadar birçok konuda alanın durumu özetlenilmiştir.

1.3. ÇALIŞMANIN SINIRLILIKLARI

1. Önerilen yöntem sadece ikili sınıflandırma problemi için geliştirilmiştir. Çok sınıflı sınıflandırma problemleri için genelleştirme gerçekleştirilmemiştir.
2. Önerilen yöntemde kullanılan kümeleme yaklaşımı için küme sayısı kullanıcı tarafından belirlenmelidir.
3. Önerilen yöntem belirlenen üç veri kümesi üzerine denenmiştir. Daha fazla veri kümesi üzerinde aynı yaklaşım denenerek genelleştirme çalışmaları devam edilmelidir.
4. Özgün bir veri kümesi üzerinde önerilen yaklaşım denenmemiştir. Veriler dengesiz veri kümeleri konusunda literatürde kullanılmış olan veri kümelerinden seçilmiştir.

2. GENEL KISIMLAR

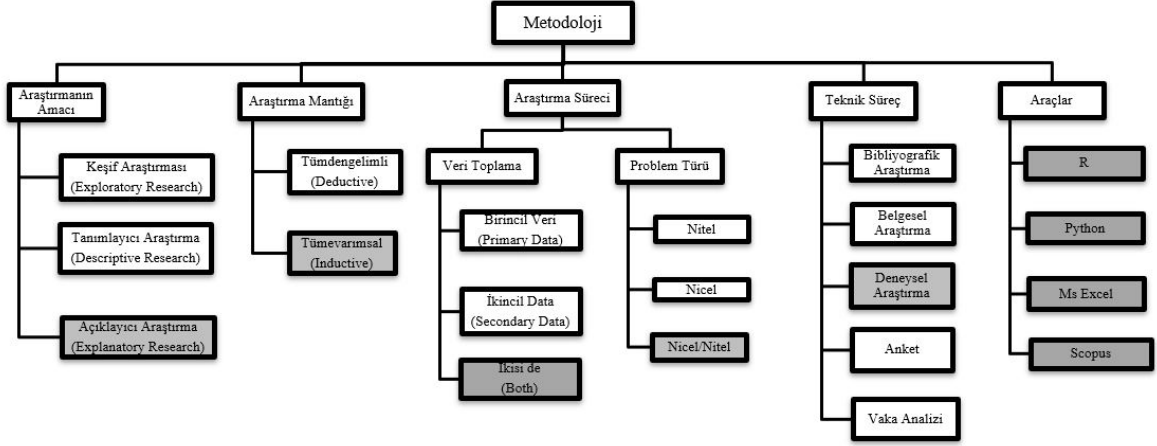
Bu bölümde dengesiz veri kümeleriyle kurulan makine öğrenmesi modellerine getirilen çözüm önerileri bibliyometrik analiz ile sunulmuştur. Bibliyometrik analiz metodolojisi ve kullanılan teknolojiler bölüm 2.1.'de yer almaktadır. Önerilen yöntemin test edilmesi için kurulacak makine öğrenmesi modelinin temelleri, literatür incelemesinden sonra bölüm 2.2.'de verilmiştir. Önerilen yöntem ve seçilen değerlendirme parametreleri ise Malzeme ve Yöntem başlığı altında sunulmuştur.

2.1. BİBLİYOMETRİK ANALİZ

Makine öğrenimi ile ilgili araştırma yayınlarının hızla artması, alanın araştırmacılar tarafından takip edilmesini zorlaştırmaktadır. Bibliyometrik analiz, bir araştırma alanını gözden geçirmek ve yükselen trendleri keşfetmek için nicel ve istatistiksel analizleri kullanmayı hedefler [38]. Bibliyometrik analiz ile yayınların literatürdeki etkinliği konusunda bilgi sahibi olunurken, alandaki gelişmelerin takip edilmesini de kolaylaştırmaktadır.

Bibliyometrik analiz sonuçlarına girmeden önce kullanılan metodolojiyi anlamak önemlidir. Bu çalışmada kullanılan metodolojik çerçeve Şekil 2.1'de sunulmuştur. Çalışmada kullanılan bileşenler gölgeli kutularla temsil edilmiştir. Çalışmamız, bir durumu açıklamak için veri toplama tekniği olarak tanımlanan açıklayıcı araştırma örneğidir. Araştırma mantığımız, endüktif olarak da bilinen bibliyometrik analiz metodu aşağıdan yukarıya stratejiyi benimsemektedir. Araştırmada Şekil 2.1'de birincil ve ikincil veriler olarak ifade edilen, yayınların hem bağlamını hem de meta verileri incelenmektedir. Bibliyometrik analiz veri kümesi, *imbalanced dataset*, *imbalanced big dataset*, *oversampling*, *smote*, *unbalanced big dataset*, *unbalanced dataset* ve *undersampling* anahtar kelimelerini içeren ve 1957-2021 yılları arasında yayınlanan makalelerin Scopus'tan [41] toplanan verilerinden oluşmaktadır. Eksik değerler ve tekrarlar giderildikten sonra geriye 16255 yayına ait üst veri kalmıştır. Ön işleme tabi tutulan veriler, Aria ve diğerleri [42] tarafından geliştirilen R Bibliometrix paketi sürüm 3.1.4 kullanılarak analiz edilmiştir. Bibliometrix paketi ortak alıntı veya işbirliği

yapan yazarları, yayınlarda geçen ortak anahtar kelimelerin analizini kolaylaştırmakla kalmaz, aynı zamanda görselleştirmeleri de kolaylaştırır [42].



Şekil 2.1: Bibliyometrik analiz metodolojik çerçeve [39].

Bu bölümde bibliyometrik analiz sonucu elde edilen bulgulara yer verilmiştir. 16255 makaleye ait verinin nicel bir analizinin yanı sıra alanın tarihsel gelişimi ve eğilimine genel bir bakış sunulmuştur.

2.1.1. Yıllık Bilimsel Üretim

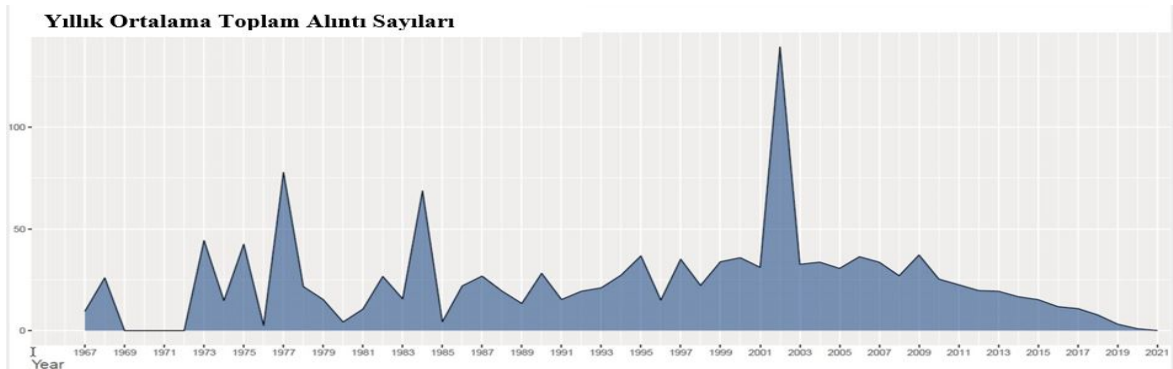
Önceden işlenmiş veriler, konunun yıllar içinde çektiği ilginin gelişimini göstermek için analiz edilmiştir. Şekil 2.2 1957 ve 2021 yılları arasındaki yıllık yayın sayısını göstermektedir. Yapay zeka araştırmalarının ilk aşamalarında dengesiz veri kümelerine ilgi çok nadir görülmüştür. Özellikle 2003 yılından sonra konuya olan küresel ilginin arttığı gözlemlenmektedir. Son yıllarda, yayın sayılarında görülen dramatik artış, Şekil 2.2’de belirgin olarak gözlemlenmektedir.



Şekil 2.2: 1957-2021 yılları arası dengesiz veri kümeleri üzerine Scopus yayın sayıları [40].

2.1.2. Ortalama Toplam Alıntı

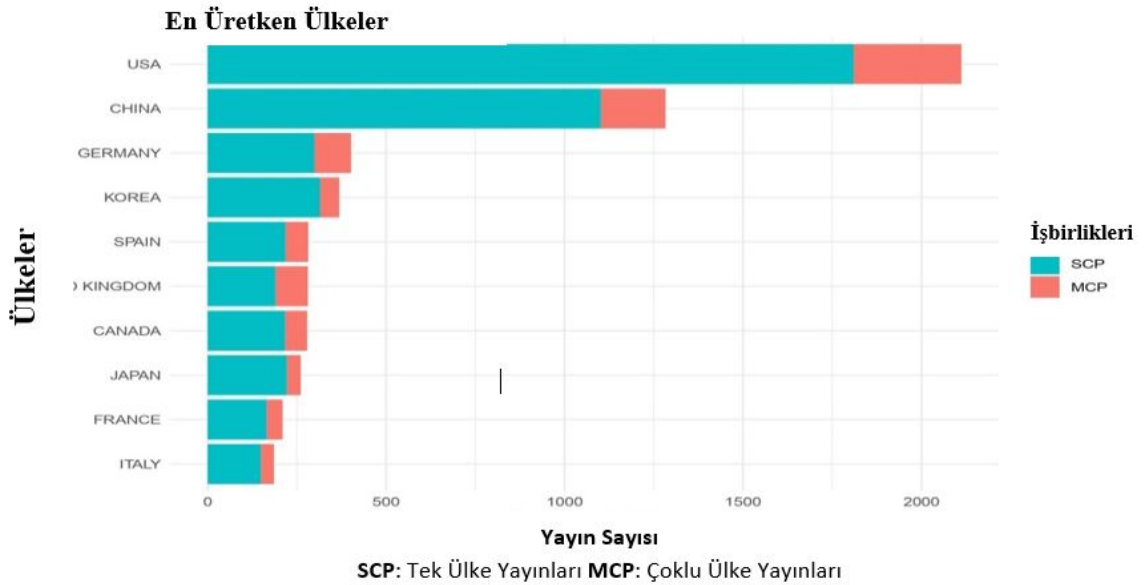
Şekil 2.2’de gösterildiği gibi dengesiz veri kümelerine açık bir ilgi olmasına rağmen, Şekil 2.3 ortalama alıntılarının azaldığını ve zaman içinde konuya olan ilginin değişkenlik gösterdiğini işaret etmektedir. En fazla ortalama atıf 2002 yılında görülmektedir. Bu yılda yapılan yayınların kapsamlı bir incelemesinin ardından, yayınlardaki artışın sebeplerinden birinin, araştırmacılar tarafından iyi bilinen ve sıklıkla kullanılan Sentetik Azınlık Aşırı Örnekleme Tekniğinin (*synthetic minority oversampling technique*, SMOTE) tekniğinin Chawla ve diğerleri tarafından [43] 2002 yılında geliştirilmesi olduğu gözlemlenmiştir. SMOTE yaklaşımı günümüzde dengesiz veri kümeleri probleminin çözümünde hala aktif olarak araştırmacılar tarafından kullanılmaktadır. SMOTE yaklaşımının duyurulmasından sonra SMOTE çözümünü merkez alan bir çok hibrit yaklaşım da sunulmuştur.



Şekil 2.3: Yıllık ortalama toplam atıf sayısı [40].

2.1.3. Ülkelerin Dengesiz Veri Kümeleri Probleminde Çalışmaları

Şekil 2.4 dengesiz veri kümeleri problemi hakkında en fazla yayın çıkaran ülkeleri göstermektedir. Araştırmacıların kendi alanlarına katkıda bulunan ülkeleri tanımaları alanın gelişimini takip etmek ve işbirlikleri kurmak için önemlidir. Şekil 2.4'e göre, makalelerin çoğunluğu Amerika Birleşik Devletleri'ndeki yazarlar tarafından yayınlanmaktadır, bunu sırasıyla Çin ve Almanya izlemektedir. Şekil 2.4'ten, farklı ülkelerdeki yazarların işbirliklerinden ortaya çıkan yayınlardan daha fazla tek ülke yayınının olduğu da gözlemlenmektedir. Şekilde tek ülke yayınları SCP, farklı ülkelerden yazarların işbirlikleri ise MCP kısaltmasıyla gösterilmiştir. MCP, her ülke için farklı bir ülkeden en az bir ortak yazarın bulunduğu belge sayısını gösterir. Amerika Birleşik Devletleri bu alanda en fazla ülke işbirliğine sahip olan ülke olarak görülmektedir. Ancak, yüksek düzeyde uluslararası işbirliğine sahip ülkelerin yüzdelere (MCP/ (MCP+SCP)) baktığımızda, %32 ile Birleşik Krallık ilk sırada yer alırken, onu Almanya (%26) ve İspanya (%23) sırasıyla izlemektedir.

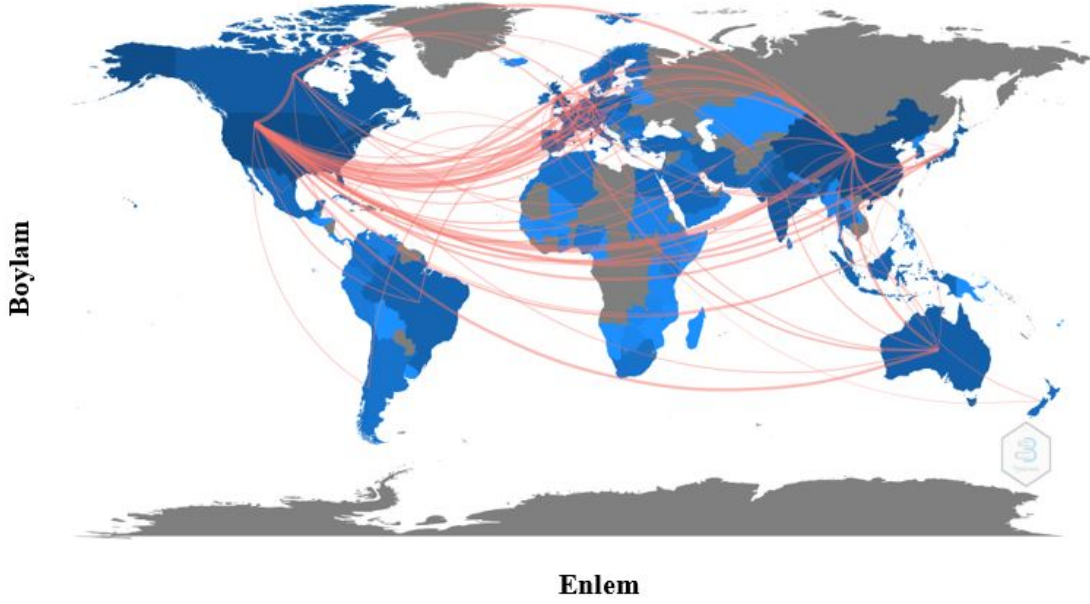


Şekil 2.4: Ülkelere göre dengesiz veri kümesi problemi yayın miktarları [40].

Ülkelerin işbirliği ve ülkelere göre araştırma yaygınlığı Şekil 2.5'te gösterilmektedir. Şekil 2.5'te görülebileceği gibi, Amerika Birleşik Devletleri ve Çin'in yanı sıra Amerika Birleşik Devletleri ve Avrupa, özellikle Almanya arasında güçlü işbirlikleri olduğu görülmektedir. Hong Kong, Birleşik Krallık ve Kanadalı yazarların, Çin'deki yazarlar için güçlü ortaklar

olduğu görünüyor.

Ülke İş Birliği Haritası



Şekil 2.5: Dünya haritası üzerinde ülkelerin işbirliği [40].

2.1.4. En Çok Atıf Alan Yayınlar

Bibliometrix paketinin web arayüzünden elde edilen Tablo 2.1, dengesiz veri kümeleri sorunu alanında en çok atıf alan 20 yayını göstermektedir. Toplam atıflar (*Total citations*), her bir makaleye kaç kez atıf yapıldığını gösterirken, yerel atıflar (*local citations*), bu koleksiyondaki bir yazarın (veya bir belgenin) yine bu koleksiyondaki diğer yazarlar tarafından kaç kez alıntılandığını gösterir. Bölüm 2.2.2’de bahsedildiği gibi, Chawla ve diğ. [43] yayınladıkları çalışma dengesiz veri kümelerine çözüm arayan araştırmacılar tarafından çok dikkat çekip, şu anda da alandaki en çok başvurulan makalelerden biridir. Çalışmada, azınlık verilerinin sentetik örnekleri yaratılarak azınlık sınıf örneklem sayısının artırılması hedeflenmiştir [43]. Borderline-SMOTE makalesi, SMOTE yaklaşımını temel alarak yeni bir teknik öneren ve en yüksek atıf alan ikinci makaledir [44]. Han ve diğ. [44], sadece sınır çizgisi etrafındaki azınlık örneklerini çoğaltan ve borderline-SMOTE1 ve borderline-SMOTE2 olarak adlandırdıkları iki yeni yukarı örnekleme yaklaşımı geliştirmişlerdir. SMOTE yaklaşımı ile karşılaştırıldığında, borderline yönteminin model performans değerlendirme yöntemlerinden olan doğru pozitif oranı ve F-ölçütü açısından daha iyi sonuçlar ürettiği gözlemlenmiştir [44]. En çok atıf

alan üçüncü yayın, bir hibrit yaklaşım olan ve Chawla ve diğ. [22] tarafından geliştirilen SMOOTEBoost yaklaşımının makalesidir. SMOOTEBoost yaklaşımı SMOTE ve boosting yaklaşımlarının birlikte kullanılmasına dayanmaktadır. Yeni önerilen SMOOTEBoost yaklaşımı ile yüksek ve orta derecede dengesiz veri kümelerinde azınlık sınıfını ve F-ölçütünü tahmin etmede daha iyi başarı elde edilmiştir [45]. Galar ve diğ. [46] dengesiz veri kümeleri üzerine bir literatür taraması yayınlamışlardır. Araştırmaya göre, bagging veya boosting tekniklerinin rastgele aşağı örnekleme ile birlikte kullanılmasının model performansını arttırdığı ifade edilmiştir. Aynı zamanda dengesiz veri kümelerinde kullanılan topluluk tabanlı (*ensemble*) yaklaşımlarının veri ön işleme çözümlerinden daha başarılı olduğu söylenmiştir. Aşağı örnekleme, dengesiz veri kümeleriyle başa çıkmak için kullanılan ana tekniklerden biridir. Liu ve diğ. [47] aşağı örnekleme tekniğinin zayıflığına dikkat çekip, bununla başa çıkmak için iki yeni yaklaşım önermişlerdir. Yeni önerilen yöntemler EasyEnsemble ve BalanceCascade, aşağı örnekleme tekniğinde göz ardı edilen çoğunluk sınıfı örneklerini de dikkate alır. Çalışmaya göre bu yöntemler sadece ROC eğrisi altında kalan alanı, F-ölçütünü ve G-mean değerlerini iyileştirmekle kalmayıp, aynı zamanda eğitim süresini de hemen hemen uzatmamaktadır. Dengesiz veri kümeleri makine öğrenmesi modellerinin performansının azalmasındaki tek sebep değildir [48]. Lopez ve diğ. düşük performanli makine öğrenmesi modellerinin gelişmesinde, dengesiz veri kümeleriyle birlikte *small disjuncts* (veri kümesi içinde alt grupların olması), *overlapping* veya *class separability* (hedef sınıfların kesişen bölgelerinin olması veya sınıfların ayırt edilebilirliğinin az olması), gürültülü veriler (*noisy data*), *borderline examples* (sınırdaki örnekler) ve veri kümeleri kayması gibi dengesiz veri kümelerinin alt problemlerinin de bulunması olduğunu vurgulamışlardır [48]. Ayrıca dengesiz veri kümeleri için sunulan çözümleri karşılaştırmak için karşılaştırmalı bir araştırma gerçekleştirmişlerdir. Destek Vektör Makineleri (*Support Vector Machines* veya kısa adıyla SVM), sınıflandırma problemlerinde sıklıkla kullanılan algoritmalarından biridir. Ancak dengesiz veri kümelerinden olumsuz etkilenmektedir. Akbani ve diğ. [49], çalışmalarında dengesiz veri kümeleri durumunda SVM'nin performans düşüşünü tartışmış olup, aşağı örnekleme tekniğinin de dezavantajını ele alarak hibrit bir yöntem geliştirmişlerdir. Chawla ve diğ.'nin [43] önerdiği SMOTE yaklaşımı ile Veropoulos ve diğ. [50] önerdiği farklı hata maliyeti hesaplama algoritmalarını birlikte kullanan SDC (*SMOTE with Different Costs*) yöntemini önermişlerdir. Başka bir hibrit yöntem olan RUSBoost, Seiffert ve diğ. [51] tarafından araştırılmıştır. RUSBoost, rastgele yetersiz örnekleme ve AdaBoost kombinasyonunu

içeren bir SMOTEBoost varyasyonudur. Hem SMOTEBoost hem de RUSBoost teknikleri AdaBoost algoritmasına dayansa da örnekleme yöntemlerinin farklılaşmasının RUSBoost'u SMOTEBoost'tan daha hızlı yaptığı belirtilmektedir [51]. Zhou ve diğ. [52], maliyete duyarlı öğrenme yöntemlerinin de dengesiz veri kümeleri sorununu çözmede etkili olup olmadığı ve bunun tersinin de geçerli olup olmadığı sorularını cevaplamaya çalışmışlardır. 21 UCI veri kümesi üzerindeki araştırmaları, maliyete duyarlı sinir ağlarının daha yüksek derecede sınıf dengesizliği varlığında model kurmanın daha zor olduğunu göstermektedir [52]. Estabrooks ve Japkowicz [53], aşağı örnekleme ve yukarı örnekleme yaklaşımlarının etkinliğini araştırmış ve daha iyi sonuçlara ulaşmak için hibrit bir şema sunmuştur. SVM ve veri örnekleme yöntemlerinin kombinasyonu hakkında ayrıntılı bir çalışma Tang ve diğ. [54] tarafından yapılmıştır. Tang ve diğ. dört yaklaşım önerip, bu yaklaşımları son teknoloji çözümlerle karşılaştırmışlardır. Önerilen yeni yaklaşımların sonuçlarına göre, aşağı örnekleme için SVM'nin kendisini doğrudan kullanmayı amaçlayan Granüler Destek Vektör Makineleri – Tekrarlayan Undersampling (*Granular Support Vector Machines – Repetitive Undersampling*, GSVM-RU)'in, dört yaklaşım arasında daha iyi sonuçlar verdiğini aktarmışlardır [54]. SMOTE yaklaşımı, çoğunluk sınıfı örneklerinden bağımsız olarak en yakın komşulardan yeni örnekler sentezleyerek azınlık sınıfına ait örnek sayısını arttırır. Safe-Level-SMOTE, güvenli seviyeyi (*safe level*) en yakın k komşudaki pozitif örnek sayısı olarak tanımlar ve tüm sentetik örnekleri güvenli seviyede üretmeyi amaçlar [55]. Bunkhumpornpat ve diğ. [55] önerdikleri yeni yaklaşım SMOTE ve Borderline-SMOTE ile karşılaştırıldığında daha iyi F-ölçütü değeri elde etmişlerdir. Wang ve Yao [56], dengesiz veri kümelerinde çeşitliliğin etkisini araştırmış ve daha fazla çeşitliliğin azınlık sınıfı için daha iyi duyarlılık değerine, ancak çoğunluk sınıfları için daha kötü duyarlılık değerlerine neden olduğunu belirtmişlerdir. Tıbbi veri kümeleri, sorunun doğası gereği her zaman dengesizliğe eğilimlidirler. Tıbbi teşhiste dengesiz veri kümelerinin etkileri üzerine yapılan araştırmalar, dengesizlik durumunda model performansının düştüğünü göstermektedir [57]. Mazurowski ve diğ. [57] geri yayılım (BP, *backpropagation*) ve parçacık sürü optimizasyonunu (PSO, *particle swarm optimization*) sinir ağlarını uygular ve BP'nin dengesiz veri kümeleriyle her zaman PSO'dan daha iyi sonuçlar verdiğini bildirmişlerdir. RUSBoost tabanlı EUSBoost çeşitliliğe daha fazla odaklanmak için, Galar ve diğ. [58] tarafından önerilmiştir. EUSBoost, tüm azınlık sınıfı örneklerini ve seçilmiş çoğunluk sınıfı örneklerini içeren ve her sınıflandırıcıyı çoğunluk sınıfının farklı alt kümeleri ile eğiterek çeşitliliği sağlayan bir veri kümesi elde edilmesini sağlar. SMOTE tekniğinin yayınlanması, dengesiz

veri kümeleri literatürü üzerinde büyük etki yaratmıştır ve zamanla farklı varyasyonlarının ortaya çıkmasına da yol açmıştır. Fernandez ve diğ. [59] sadece SMOTE ve uygulamalarına dayalı çalışmaları analiz etmekle kalmıyor, aynı zamanda büyük veri üzerinde SMOTE uygulamaları için olası zorlukları da tanımlamışlardır. Gün geçtikçe artan veri hacmi, dengesiz veri kümeleri için mevcut yöntemleri kullanırken performans sorunlarına neden olmaktadır. Triguero ve diğ. [60], performans sorunlarının üstesinden gelmek için aşağı örnekleme yaklaşımı için bir MapReduce şeması önermişlerdir. Sáez ve diğ. [61] ayrıca dengesiz veri kümelerinde gürültülü ve sınırdaki örneklerin öneminin altını çizerek ve dezavantajlarını hafifletmek için SMOTE'un bir varyasyonunu önermişlerdir. Gürültülü örnekleri filtrelemek ve sınıf sınırlarında yer alan örnekleri temizlemek için *Iterative-Partitioning Filter* (IPF) yöntemini önermişlerdir. Biyomedikal ve biyoinformatik, yüksek boyutluluk ve dengesiz veri kümeleri ile sıklıkla mücadele eden diğer alanlardır. Blagus ve Lusa [62], g-means değerini maksimize etmeyi hedefleyen Geometrik Ortalama En Yakın Küçülen Merkez (GM-NSC, *Geometric Mean Nearest Shrunk Centroid*) yöntemini önermişlerdir. Kredi puanlamasında dengesiz veri kümeleri sorunu Brown ve Mues [63] tarafından incelenmiştir. Araştırmalarına göre, rastgele ormanlar (*random forest*) ve (*Placeholder1*) gradient boosting algoritmaları daha iyi sonuçlar verirken, Kuadratik Diskriminant analizi (QDA, *Quadratic discriminant*) ve C4.5 gibi tekniklerin daha kötü sonuçlar verdiklerini aktarmışlardır.

2.1.5. En Sık Kullanılan Anahtar Kelimeler

Yayın anahtar kelimelerinin analizi, alan hakkında ilk fikir edinmek için önemli bir adımdır. Tablo 2.2, yazarlar tarafından en çok kullanılan anahtar kelimeleri göstermektedir. Anahtar kelimelere dayanarak, yazarların veri ön işleme yöntemlerinden birini kullanarak veri kümelerini dengelemeye odaklandıkları görülmektedir. Ayrıca veri ön işleme yöntemleri arasında yazarların en fazla ilgisini yukarı örnekleme ve onu temel alan yöntemler çekmektedir. Örnekleme yöntemlerinin ardından yazarlar, en çok rastgele orman, destek vektör makinesi ve topluluk modelleri algoritmalarına odaklanmışlardır. Tablo 2.2'den görüntü sınıflandırma probleminin de dengesiz veri kümelerinden muzdarip olduğu ve yazarların dikkatini çektiği görülmektedir.

Şekil 2.6'da yer alan kelime bulutu, *keywords plus* sözcük öbeklerine dayalı olarak zaman içinde en sık kullanılan kelimeleri göstermektedir. *Keywords plus* bir makalenin referanslarının başlıklarında sıklıkla yer alan, ancak mutlaka makalenin başlığında veya

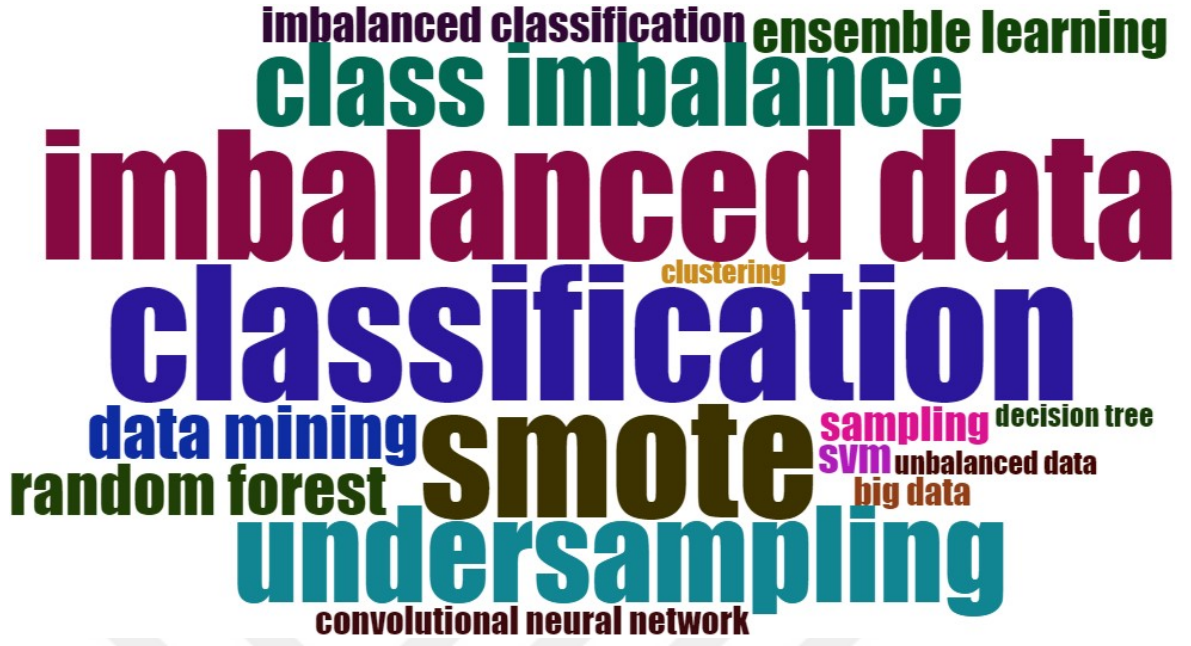
Tablo 2.1: En çok atıf alan 20 makalenin özet bilgileri [40].

Yayın	DOI	Yerel Atıf Sayısı	Toplam Atıf Sayısı
CHAWLA NV, 2002, J ARTIF INTELL RES	10.1613/jair.953	2915	7540
HAN H, 2005, LECT NOTES COMPUT SCI	10.1007/11538059_91	756	1131
CHAWLA NV, 2003, LECT NOTES ARTIF INTELL	10.1007/978-3-540-39804-2_12	504	834
GALAR M, 2012, IEEE TRANS SYST MAN CYBERN PT C APPL REV	10.1109/TSMCC.2011.2161285	435	1145
LIU XY, 2009, IEEE TRANS SYST MAN CYBERN PART B CYBERN	10.1109/TSMCB.2008.2007853	387	915
LPEZ V, 2013, INF SCI	10.1016/j.ins.2013.07.007	336	704
AKBANI R, 2004, LECT NOTES ARTIF INTELL	10.1007/978-3-540-30115-8_7	329	713
SEIFFERT C, 2010, IEEE TRANS SYST MAN CYBERN PT A SYST HUMANS	10.1109/TSMCA.2009.2029559	320	734
ZHOU ZH, 2006, IEEE TRANS KNOWL DATA ENG	10.1109/TKDE.2006.17	293	732
ESTABROOKS A, 2004, COMPUT INTELL	10.1111/j.0824-7935.2004.t01-1-00228.x	249	556
TANG Y, 2009, IEEE TRANS SYST MAN CYBERN PART B CYBERN	10.1109/TSMCB.2008.2002909	215	498
BUNKHUMPORNPAT C, 2009, LECT NOTES COMPUT SCI	10.1007/978-3-642-01307-2_43	213	324
WANG S, 2009, IEEE SYMP COMPUT INTELL DATA MIN , CIDM - PROC	10.1109/CIDM.2009.4938667	142	254
MAZUROWSKI MA, 2008, NEURAL NETW	10.1016/j.neunet.2007.12.031	141	419
GALAR M, 2013, PATTERN RECOGN	10.1016/j.patcog.2013.05.006	119	198
FERNNDEZ A, 2018, J ARTIF INTELL RES	10.1613/jair.1.11192	106	163
GARCA S, 2009, EVOL COMPUT	10.1162/evco.2009.17.3.275	104	218
SEZ JA, 2015, INF SCI	10.1016/j.ins.2014.08.051	102	183
BLAGUS R, 2013, BMC BIOINFORM	10.1186/1471-2105-14-106	90	191
BROWN I, 2012, EXPERT SYS APPL	10.1016/j.eswa.2011.09.033	87	272

Tablo 2.2: Yayınlarda kullanılan en popüler yazar anahtar kelimeleri [40].

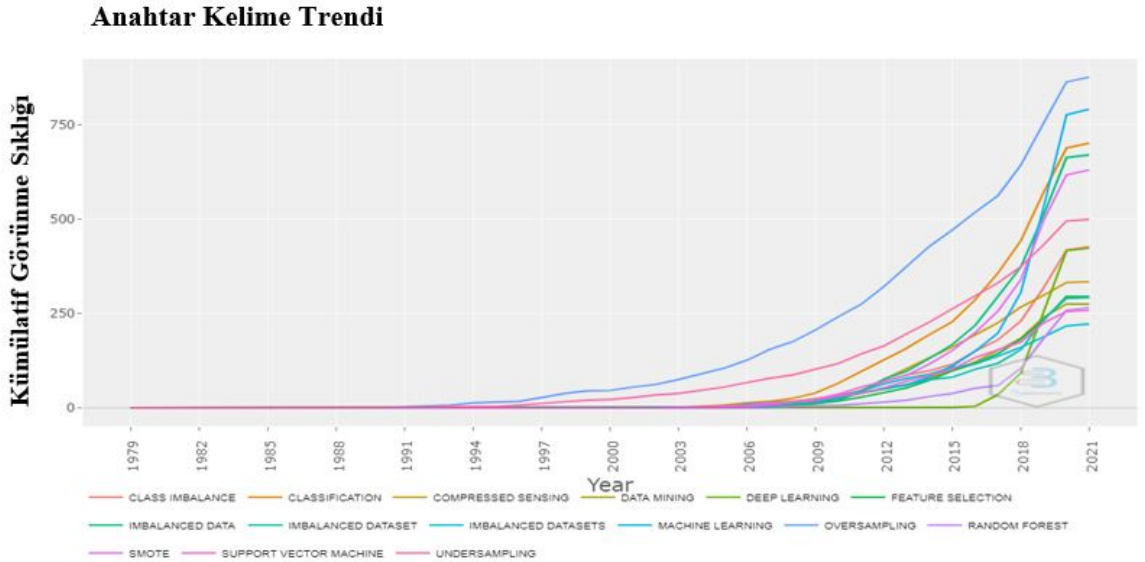
Words	Occurrences	Words	Occurrences	Words	Occurrences	Words	Occurrences
oversampling	877	feature selection	295	sampling	180	under-sampling	99
machine learning	791	imbalanced dataset	294	svm	178	parallel imaging	94
classification	701	data mining	276	convolutional neural network	147	convolutional neural networks	93
imbalanced data	672	random forest	265	big data	144	image reconstruction	93
smote	633	support vector machine	258	clustering	120	boosting	92
undersampling	502	imbalanced datasets	222	unbalanced data	120	neural networks	89
class imbalance	428	ensemble learning	219	decision tree	119	logistic regression	84
deep learning	423	imbalanced classification	184	over-sampling	109	adaboost	83
compressed sensing	334	imbalanced learning	184	support vector machines	99	anomaly detection	83

yazar anahtar kelimeleri olarak görünmeyen terimler olarak tanımlanır [42]. Kelime bulutunda, *smote*, *random forest* ve *undersampling* kelimelerinin öne çıktığı görülmektedir. Büyük veri için dengesiz veri kümeleri de yazarların odaklandığı konular arasında görülmektedir.



Şekil 2.6: Dengesiz veri kümelerinde kullanılan anahtar kelimelerin sıklığı [40].

Şekil 2.7, gelecekteki araştırmalar için daha iyi konuların seçilmesine yardımcı olabilecek yazarın anahtar kelimelerinin zaman içindeki eğilimini göstermektedir. Şekle 2.7'ye göre, *oversampling*, *machine learning*, *classification*, *imbalanced data* ve *SMOTE*, diğer anahtar kelimelere kıyasla daha popüler hale gelmeye başlayan konulardır.



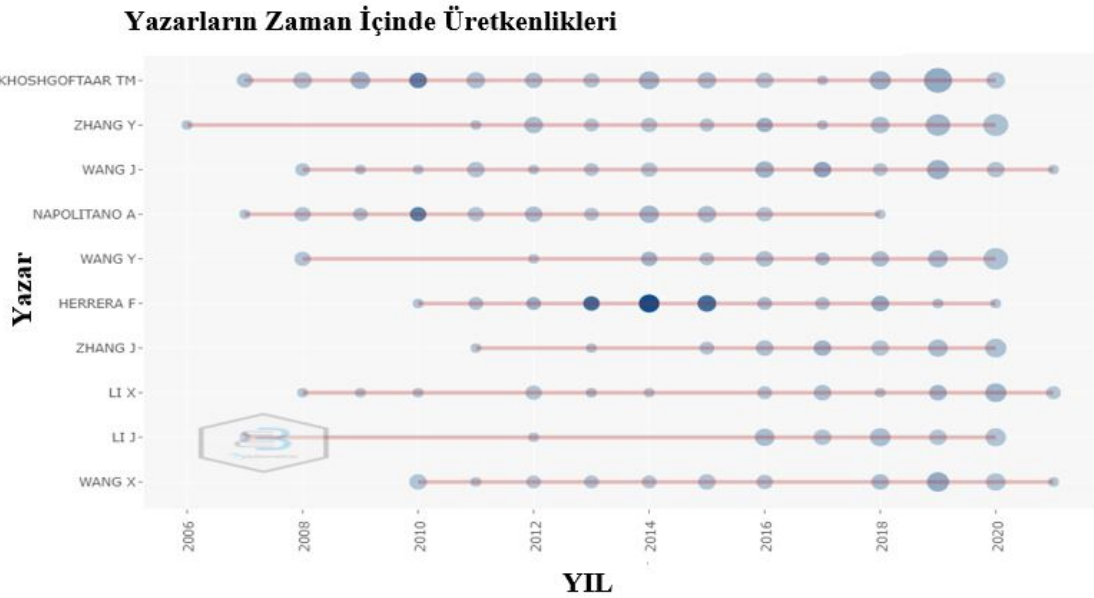
Şekil 2.7: Zamansal kümülatif kelime dinamiği [40].

Güncel yaklaşımlara takip edebilmek için alanında en çok atıf alan yazarları bilmek kadar alanında aktif araştırmacıları takip etmek de önemlidir. Şekil 2.9, yazarların zaman içindeki üretkenliğini göstermektedir. Dairelerin boyutu yayın sayısını gösterirken, dairelerin renkleri yıllık toplam atıf sayılarını vurgular. Şekil 2.9'a göre, Taghi M. Khoshgoftaar ve Yong Zhang

bu alandaki en aktif araştırmacılarıdır.

Tablo 2.3: En çok atıf alan yazarlar [40].

Yazar	Toplam Atıf Sayısı	Yazar	Toplam Atıf Sayısı
CHAWLA N V	5099	HE H	2546
HERRERA F	3957	KEGELMEYER W P	2437
HALL L O	3360	WANG Y	2428
BOWYER K W	3191	KHOSHGOFTAAR T M	1921
JAPKOWICZ N	2911	ZHANG Y	1893



Şekil 2.9: Yazarların zaman içindeki üretkenlik grafiği [40].

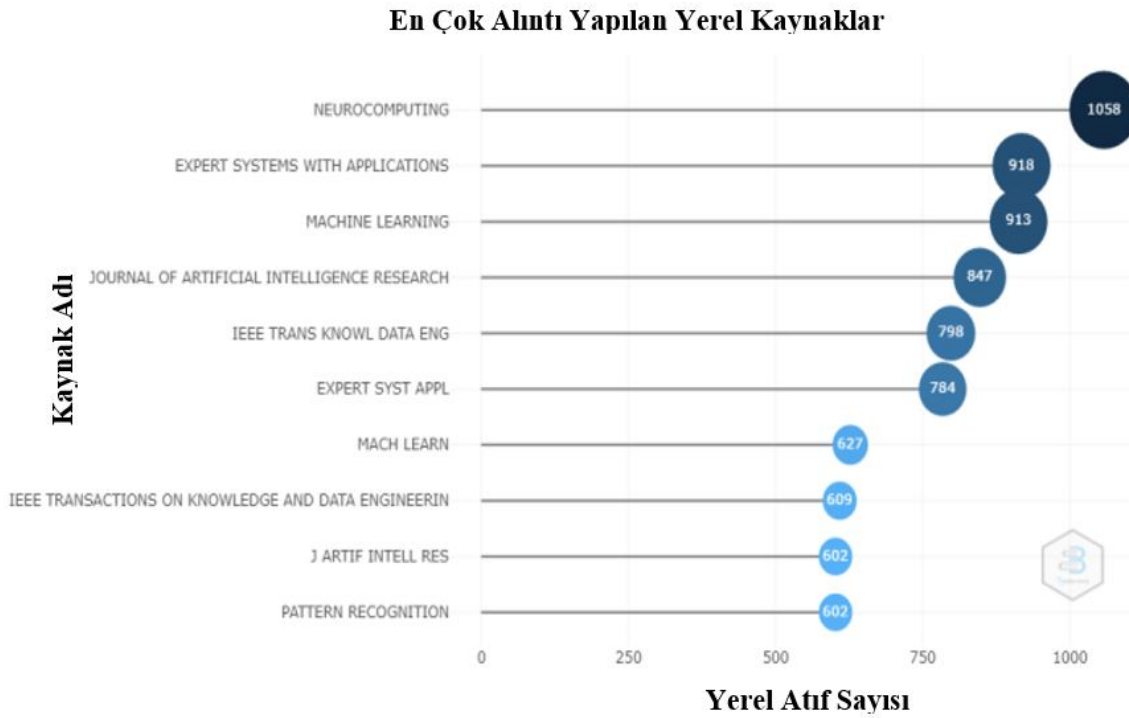
Garfield [62], bir bibliyografik koleksiyondan en alakalı doğrudan alıntıların zamansal bir ağ haritasını göstermek için bir grafik olarak *historiographic map* (tarihsel harita)’i tanıtmıştır. Şekil 2.10 sahadaki ilerlemenin yönünü gösteren tarihsel haritayı göstermektedir.

Şekil 2.10’a göre, dengesiz veri kümeleri üzerine bir dizi araştırma 2011 yılında Fernandez-Navarro ve diğ. [65] tarafından başlatılmıştır ve çalışmaların Lopez ve diğ. [48] tarafından 2013 yılında devam ettiği gözlemlenmektedir. Fernandez-Navarro ve diğ. [65] dengesiz veri kümelerinde çok sınıflı sınıflandırma problemi için dinamik bir yukarı örnekleme (*dynamic oversampling*) yaklaşımını önermişlerdir. Kovacs [66], 2019 yılında SMOTE yöntemini temel alan 85 yaklaşımın karşılaştırmalı bir çalışmasını yayınlamıştır. Kovacs’ın yayınına ek olarak Torres-Vasquez ve diğ. [67] bir bağıklık sistemi bozukluğu

Şekil 2.10: Yazarların yayınlarının tarihsel ilerleyişi [40].

2.1.8. En Çok Alıntı Yapılan Kaynaklar

Çalışmaları yayınlamak için ideal dergiyi seçmek yazarlar için zaman alan bir işlem olabilmektedir. Şekil 2.11, veri kümesini oluşturan (yerel) kaynaklar arasından alandaki en alakalı olanları göstermektedir. Şekle göre neurocomputing alanındaki mevcut temel gelişmeleri tanımlayan **Neurocomputing**, en çok alıntı yapılan kaynaktır. En çok atıfta bulunulan ikinci kaynak, dünya çapında iş dünyası, devlet ve üniversitelerde uzman ve akıllı sistemler hakkında bilgi alışverişine adanmış uluslararası bir dergi olan **Expert Systems with Applications**'dır. Geniş bir yelpazede araştırma makaleleri yayınlayan ve çeşitli öğrenme konularına uygulanan öğrenme yaklaşımlarını konu alan **Machine Learning** üçüncü sırada yer almaktadır.

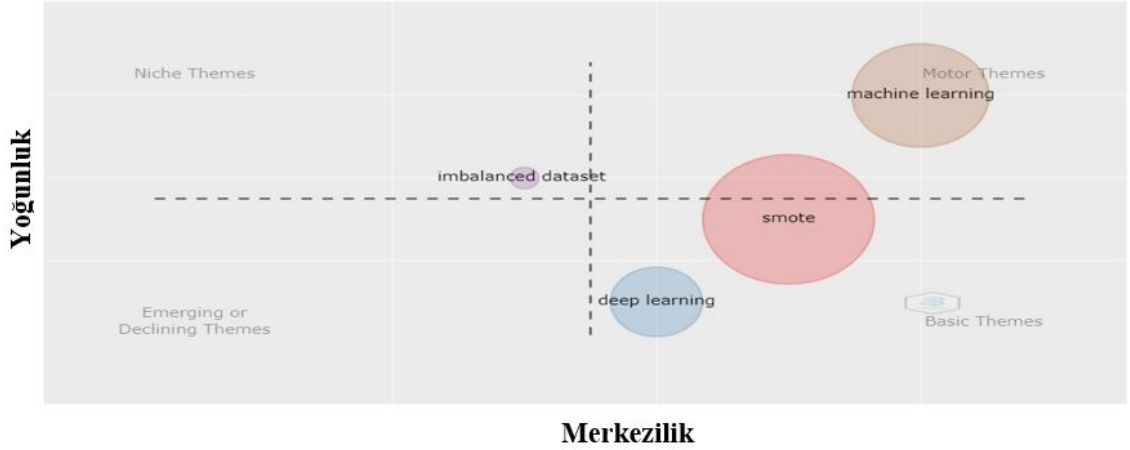


Şekil 2.11: Yayın yapılan kaynakların alınan atıf sayıları grafiği [40].

2.1.9. Tematik Harita

Tematik harita (*thematic map*), bir grup anahtar kelimeyi ve bunların tema (*theme*) olarak kabul edilen ilişkilerini gösterir [74]. Tematik haritanın yoğunluk (*density*) ve merkezilik (*centrality*) olarak adlandırılan iki boyutu vardır. Merkezilik yatay ekseninde yer alır ve diğer

temalarla ilişkileri gösterirken yoğunluk dikey ekseninde yer alır ve küme içi bağlantıları gösterir. Tematik harita, kadrantlarına göre analiz edilebilmektedir. Şekil 2.12’de yer alan tematik haritaya göre önemli araştırma kümeleri sağ alt kadranda yer almaktadır. Sol üst kadranda özelleşilmiş konu kümelerini gösterir. *Motor Themes* alanı dengesiz veri kümelerinin konu olduğu çalışma alanını ve en çok kullanılan çözüm önerisini göstermiştir.



Şekil 2.12: Tematik harita [40].

2.2. DENGESİZ VERİ KÜMELERİNE İLİŞKİN YAPILAN TÜRKİYE’DEKİ TEZLER

Dengesiz veri kümeleri ile yapılan çalışmalarda en çok atıf alan yayınlar, güncel yaklaşımlar ve yazarların konuya ilgileri bibliyometrik analiz ile incelenirken, bu bölümde konuyla ilgili Türkiye’de yapılan tezlere yer verilmiştir. Yayınlanan tezler, daha önce önerilen yöntemlerin farklı konulara ilişkin dengesiz veri kümelerinde denenmesi ve yeni metodolojilerin önerildiği yayınlardan oluşmaktadır. İncelenen tezler ulusal tez merkezinden *imbalanced data* anahtar kelimesiyle yapılan arama sonucu seçilmiştir. Yapılan arama sonucunda 11 adet yüksek lisans tezi ve 2 adet doktora tezi olmak üzere konu ile ilgili toplam 13 adet teze ulaşılmıştır. Öztürk [75] önerdiği yöntemde, dengesiz veri kümeleri ile oluşturulan SVM modellerindeki performans probleminin önüne geçmek için azınlık ve çoğunluk sınıfından meydana gelen hataları ayrı ayrı dikkate almayı hedeflemiştir. Ayhan [76] ise hem ikili sınıflandırma hem de çok sınıflı sınıflandırmada dengesiz veri kümelerinin performansı olumsuz etkilediğini vurgulayıp, hem aşağı örnekleme hem de yukarı örnekleme yöntemlerini kullanan yeni bir yaklaşım önermiştir. Tez kapsamında Mahalanobis Taguchi Sistem (MTS) yöntemini inceleyip, çok sınıflı sınıflandırma probleminde önerdiği çözüm

için temel olarak kullanmıştır. Önerilen yöntemlerin sekiz farklı veri kümesinde başarılı sonuçlar verdiğini ifade etmiştir. Aydın [77] bir bilgisayarlı görü problemi olan trafik işaretlerini tanımlama projesinde ortaya çıkan dengesiz veri küme sorununa öznitelik çıkarımı ve sınıflandırma adımlarından oluşan yeni bir yaklaşım sunmuştur. Gümüştas [78] yayınladığı tezde veri kümelerinde yer alan eksik verilerin nasıl doldurulması gerektiğini araştırırken, dengesiz veri problemi için ise farklı topluluk tabanlı algoritmaları kullanarak performanslarını karşılaştırmıştır. Al-Shamaa [79] doktora tezinde, dengesiz veri problemine çözüm olarak Hellinger mesafesi alt örnekleme modelini kullanarak çoğunluk sınıfa ait örneklem sayısını azaltmayı önermiştir. Sınır bölgelerindeki örnekleri azaltarak, azınlık ve çoğunluk sınıf ayrışımını arttırmak için de ANOVA özellik seçimi yöntemini kullanmışlardır. Sağlam [80] yüksek lisans tezinde dengesiz veri kümeleme için geliştirilen veri ön işleme yöntemlerinin zayıf yönlerini ele alıp, SMOTE ve *boosting* yöntemlerini hibrit kullanan yeni bir yaklaşım önermiştir. Dikdere [81] tezinde yapay sinir ağlarını kullanarak azınlık sınıfa ait örnek sayısını ve yine yapay sinir ağlarına dayalı altı farklı yöntem daha sunmuştur. Oğul [82] duygu analizi problemine, farklı veri örnekleme yaklaşımlarını ve sınıflandırma algoritmalarını uygulayak model performansına etkilerini yorumlamıştır. Mulla [83] yayınladığı tezde yüksek boyutlu veri kümelerinde dengesiz veri kümelerinin performansa etkisini araştırmıştır. Dengesiz veri kümelerinin olumsuz etkilerini gidermek için temel bileşen analizi ve SMOTE yukarı örnekleme yöntemlerini hibrit kullanan bir çözüm sunmuşlardır. Öztürk [84] tezinde kredi risk değerlendirme problemine görülen dengesiz veri kümeleri için farklı veri işleme tekniklerini, topluluk algoritmalarıyla birlikte deneyerek performans artışını yakalamayı hedeflemiştir. Tez neticesince oversampling yönteminin ağaç tabanlı sınıflandırma algoritmasıyla birlikte kullandığı durumlarda model performansında iyileşme gözlemlemiştir. Öztürk [85] farklı dengeleme yaklaşımlarını kullanarak karar ağacı modelleri neticesinde, dengeleme yöntemlerinin hepsinin model performansını arttırdığını gözlemlemiştir. Aynı zamanda dengeleme yöntemlerini kullanırken azınlık ve çoğunluk sınıfları arasında oluşan yeni dengesizlik oranlarının da model performansına etkisini gözlemlemiştir. Arık [86] SMOTE yukarı örnekleme yöntemini kullanarak azınlık sınıfının örnek sayısını arttıran ve *Near Miss* aşağı örnekleme yöntemini kullanarak çoğunluk sınıfı örnek sayısını azaltan ROGONG-IDS modelini önermişlerdir. Önerilen yaklaşım Gradyan arttırma tabanlı üç farklı sınıflandırma algoritmasıyla kullanılarak, yöntemin literatürdeki en yüksek F-ölçütü değerini verdiği ifade edilmiştir. Pir [87] ise SVM tabanlı aşağı örnekleme

yöntemi ve *safe level smote* yukarı örnekleme yöntemini birlikte kullanarak yeni bir veri ön işleme yaklaşımı önermişlerdir. Önerilen yöntem sekiz farklı veri kümesinde denenerek olumlu sonuçlar alındığı ifade edilmiştir.

2.3. MAKİNE ÖĞRENMESİ

Günümüzde birçok problemin çözümü probleme ilişkin verilerin işlenmesine ve veriler ile çalışmayı öncelik haline getirmeye dayanmaktadır. Tartışmasız bir şekilde verilerin herhangi bir başarılı problem çözümünün temelinde olduğu inkar edilemez ve bu konuda öne çıkan çözümlerin öne geçmelerini sağlayacak şeyin verilere daha derinlemesine bakmanın olmasıdır. Verilerin günümüz problemleri için önemi arttıkça, alanda her gün yeni terimler ortaya çıkmaktadır. Bu bölümde alanda sıklıkla geçen terimler özetlenip makine öğrenmesine giriş yapılmıştır.

Veri analizi, çeşitli türlerde verileri içeren bir veri kümesinin her bir parçasının birbirleriyle olan ilişkilerini tanımlar. Veri analiziyle tüm verilerle uğraşmak yerine, veri kümesini daha kolay sindirilebilir parçalara ayırıp veriden anlam çıkarmak hedeflenir. Bununla birlikte, dikkat edilmesi gereken önemli bir nokta veri analiziyle geçen yaz bir mağazada satışların nasıl azaldığını açıklamak için bir analiz kullanmak gibi geçmişte zaten olmuş şeyler üzerinde analizler yapılmaktadır. Kısaca veri analizini bir şeyin nasıl ve neden olduğunu açıklamak için kullanırız. Veri analitiği ile ise geçmiş olayları açıklamak yerine, gelecekteki olası olaylar araştırılır. Analitik, temel olarak, bir analizde elde edilen bileşen parçalarına mantıksal ve hesaplamalı akıl yürütmenin uygulanmasıdır. Analitik ile veride bulunan örüntüler tespit edilip, gelecek tahminlemesi için kullanılması hedeflenir. Veri analitiği de iki alt başlık altında incelenmektedir. Nitel analitik ile işlemlerinizi planlamak için sezgilerinizi ve deneyiminizi analizle birlikte kullanmak hedeflenir. Nicel analitik ise analizlerden elde edilen sayılara formüller ve algoritmaların uygulanması olarak tanımlanmaktadır.

İş zekası (*business intelligence*, BI), geçmiş iş verilerini analiz etme ve raporlama sürecine verilen addır [88]. İş zekası kapsamında raporlar ve panolar hazırlandıktan sonra stratejik ve taktiksel iş kararları vermek için kullanılabilirler. Kısaca iş zekası, iş verilerini kullanarak geçmiş olayları açıklamayı amaçlar. İş zekası aynı zamanda tahmine dayalı analitiğin ilk adımıdır. Makinelerin, açıkça programlanmadan sonuçları tahmin etme yeteneği ise literatürde makine öğrenimi (*machine learning*) olarak kabul edilir. Başka bir deyişle

makine öğrenimi, makinelerin verileri almasına ve bu verileri kendi başlarına tahminler yapmak, örüntüleri analiz etmek ve önerilerde bulunmak için kullanmasına izin veren algoritmalar oluşturmak ve uygulamak olarak tanımlanabilir. Makine öğreniminin veriden ayrı düşünülmesi mümkün değildir. Kartal [89] doktora çalışmasında Mitchell'in [90] makine öğrenmesi tanımlamasından faydalanarak makine öğrenmesini aşağıdaki şekilde ifade etmiştir.

"Eğer bir bilgisayar programının G görevlerinde P ile ölçülen performansı deneyim D ile artıyorsa, o bilgisayar programının bazı G görevlerinin sınıflarına ve performans ölçüsü P 'ye göre deneyim D 'den öğrendiği söylenmektedir."

Yapay zeka ise bilgisayarla insan davranışlarını simüle eden sistemler olarak adlandırılabilir. Yapay zeka farklı yorumlara açık olan genel bir terimdir. Yapay zekaya makine öğrenmesi yaklaşımları ile daha da yaklaşılmıştır. Kısaca makine öğrenmesi yapay zeka sistemlerinin oluşturulması için kullandığımız araçlardan bir tanesidir. Derin öğrenme ise yapay sinir ağları adı verilen beynin yapısından ve işlevinden ilham alan algoritmalarla ilgili makine öğreniminin bir alt alanıdır. Derin öğrenme algoritmalarının başarımı veri miktarının artmasıyla arttığı için son yıllarda çalışmalarda daha fazla yer aldığı görülmektedir. Son yıllarda adından daha çok söz ettiren bir diğer konu nesnelerin interneti ise birbiriyle ilişkili bilgi işlem cihazları, mekanik ve dijital makineler, nesneler, hayvanlar veya insanlardan oluşan bir sistemin birbiriyle veri iletişimde bulunmasının sağlanması şeklinde özetlenebilir. Farklı kaynaklardan gelen verinin iletişim halinde kullanılması makine öğrenmesinin gelişimini etkileyen faktörler arasında yer almaktadır.

Günümüzde veri terimini kullandığımızda aklımıza ilk gelen geleneksel veri olarak kabul edilebilen, sayısal veya metin değerleri içeren tablolar biçimindeki verilerdir. Tablolar şeklinde belirli bir formatta tutulan veriler bir bilgisayardan yönetilebilir, veritabanlarında yapılandırılır ve saklanırlar. Büyük veri ise veri boyutlarının büyüdüğü, işlenmesinin zorlaştığı ve saklama için gereken donanım altyapısının arttığı durumlar için ayrılmış bir terimdir. Büyük veri yapılandırılmış, yarı yapılandırılmış veya yapılandırılmamış olabilmektedir.

Makine öğrenmesi, bir bilgisayarın verilere mümkün olan en iyi şekilde uyan bir model bulmak için kullandığı bir algoritma oluşturmak olarak tanımlanır [91]. Makine öğrenmesinin geleneksel yöntemlerden farkı kurulacak modelin kurallarının açıkça kullanıcı

tarafından belirtilmemesinden gelmektedir. Bunun yerine makineye kendi kendine veriden nasıl öğreneceği konusunda talimatlar veren algoritmalar sağlanmaktadır. Başka bir tanımlama ile makine öğrenmesi bir deneme yanılma süreci olarak ifade edilebilir. Ancak makine öğrenmesini ayırt eden özellik, birbirini izleyen her denemenin en az bir önceki kadar iyi olmasıdır. Teknik açıdan bakılırsa makine öğrenmesi veri, model, amaç fonksiyonu ve optimizasyon algoritması şeklinde dört başlık altında toplanabilir.

Çözülme istenen veri dayalı probleme göre farklı makine öğrenimi yaklaşımları bulunmaktadır. Danışmanlı veya denetimli öğrenme (*supervised learning*), danışmansız veya denetimsiz öğrenme (*unsupervised learning*), yarı-danışmanlı veya yarı-denetimli öğrenme (*semi-supervised learning*), transdüktif akıl yürütme (*transductive inference*), çevrimiçi öğrenme (*online learning*), pekiştirmeli öğrenme (*reinforcement learning*) ve aktif öğrenme (*active learning*) başlıca makine öğrenmesi türleridir [89, 92–95]. Danışmanlı ve danışmansız öğrenme, literatürde aynı zamanda eğitilmiş ve eğitimsiz öğrenme olarak da anılmaktadır. Gelecek bölümlerde danışmanlı öğrenme, danışmansız öğrenme, yarı-danışmanlı öğrenme ve pekiştirmeli öğrenme türlerini genel hatlarıyla ele alınıp tez kapsamında danışmanlı öğrenme ve danışmansız öğrenme yöntemleri uygulanarak bir model geliştirilecektir.

2.3.1. Danışmanlı Öğrenme

Danışmanlı öğrenme, girdi değişkenleri ile hedef değişken arasında bir örüntüyü öğrenmek için bir model kullanmayı içeren problem türünü tanımlamaktadır [96]. Modeller, girdi ve çıktılardan oluşan eğitim veri kümeleri kullanılarak eğitilir. Eğitim sonrası kurulan modeller yalnızca girdilerin sağlandığı test kümelerinde tahminler yapmak için kullanılır. Elde edilen sonuçlar önceden bilinen test kümesi hedef değişkenlerle karşılaştırılır ve modelin performansını ölçmek için kullanılır. Kurulan model başarımı eğitim kümesinin veri ön işleme çalışmalarına göre değişmektedir.

Danışmanlı öğrenme problemlerinin iki ana türü vardır: bunlar, bir sınıf etiketini tahmin etmeyi içeren sınıflandırma (*classification*) ve sayısal bir değeri tahmin etmeyi içeren regresyon (*regression*) olarak adlandırılmaktadır [97]. Sınıflandırma, bir sınıf etiketinin tahmin edilmesini içeren danışmanlı öğrenme problemidir. Bilimsel çalışmalarda en çok karşılaşılan iki sınıflandırma türü ikili sınıflandırma ve çok sınıflı sınıflandırma

problemleridir. İkili sınıflandırmada, iki adet hedef sınıftan birinin tahminlenmesi hedeflenir. En çok kullanılan alanlardan biri de hastalıkların tespitinde rol almasıdır. Örneğin hasta ve sağlıklı bireylerin tansiyon, ağırlık, boy gibi belirli özelliklerine bakılarak kişinin diyabete sahip olup olmadığını belirlemede ikili sınıflandırmadan sıklıkla faydalanılmaktadır [98]. Görüntü işleme ya da bilgisayarlı görme hastalık tespitinde kullanılan diğer önemli yöntemdir. Örneğin beyinden alınan manyetik rezonans görüntüleri kullanarak bireyde alzheimer hastalığının olup olmadığı tespit edilebilmektedir [99]. Hastalık tespitinde hedef sınıf değişkenlerimiz hastalık olduğunu belirten "var" ve hastalık olmadığını belirten "yok" şeklinde iki adettir. Çok sınıflı sınıflandırma problemine ise metin madenciliğinde sıklıkla karşılmaktadır [100]. Regresyon ise sayısal bir etiketi tahmin etmeyi içeren danışmanlı öğrenme problemidir. Regresyon modelleri, bir bölgede yer alan evlerin fiyatlarının tespit edilmesi, ticari bir işletmenin gelecek satışlarının tahmin edilmesi gibi cevabı yoğun şekilde aranan problemleri çözmek için kullanılmaktadır.

Hem sınıflandırma hem de regresyon problemlerinde bir veya daha fazla girdi değişkeni olabilmektedir. Aynı zamanda girdi değişkenleri, sayısal veya kategorik herhangi bir veri türü olabilmektedir. Bazı makine öğrenmesi algoritmaları, danışmanlı makine öğrenimi sorunları için tasarlandıkları için "danışmanlı" makine öğrenimi algoritmaları olarak da tanımlanır. Danışmanlı öğrenme yöntemi adıyla anılan popüler algoritmalarından bazıları destek vektör makineleri, karar ağaçları, naive bayes ve k-en yakın komşu algoritmalarıdır. Algoritmalara danışmanlı öğrenme algoritmaları da denilmesinin sebebi, verilen örnek verilerle tahminler yaparak öğrenmeleridir. Bazı algoritmalar, sınıflandırma (lojistik regresyon gibi) veya regresyon (doğrusal regresyon gibi) için özel olarak tasarlanmıştır. Bazıları ise küçük değişikliklerle (yapay sinir ağları, karar ağaçları gibi) her iki tür problem için kullanılabilir. Model performanslarını ölçmek için regresyon ve sınıflandırma için farklı değerlendirme metrikleri yer almaktadır.

2.3.2. Danışmansız Öğrenme

Danışmansız öğrenme, verilerdeki ilişkileri tanımlamak veya çıkarmak için bir model kullanmayı içeren problemlerin yer aldığı gruba verilen genel addır [101]. Danışmanlı öğrenmeye kıyasla danışmansız öğrenme, çıktılar veya hedef değişkenler olmadan yalnızca girdi verileri üzerinde çalışır. Veri kümesinde yer alan birbirine benzeyen örnek noktaların aynı grup içinde yer alması ve birbirine bezemeyen örnek noktaların birbirinden uzak

gruplar içinde yer alması temel hedeftir. Bu nedenle, danışmanlı öğrenmede olduğu gibi, danışmansız öğrenmede modeli düzelten bir öğretmen yoktur. Kullanılan algoritmalar hedef değişken olmadan veri kümesindeki örnekleri en iyi ayırt edecek grupları bulmayı hedefler.

Danışmanlı öğrenmede olduğu gibi danışmansız öğrenme için de karşılaşılan farklı problemler ve bu problemler için önerilen farklı çözümler yer almaktadır. Danışmansız öğrenme önerilen çözümlere göre iki alt grupta incelenebilir. Bu iki grup, verilerde grupları bulmayı içeren kümeleme (*clustering*) ve verilerin dağılımını özetlemeyi içeren yoğunluk kestirimi (*density estimation*)'dir. Kümeleme, veri kümesi içerisinde birbirinden farklı grupları bulmayı içeren danışmansız öğrenme problemidir. Kümeleme algoritmasına yaygın kullanılan bir örnek olarak k-Ortalamlar (*k-Means*) verilebilir [102]. Burada k ile ifade edilen verilerde keşfedilecek küme sayısıdır. Kümeleme yaklaşımlarının sıklıkla çözüm aradığı sorulardan biri müşteri segmentasyonudur [103]. İşletmeler müşteri segmentasyonu ile müşterileri demografik özellikleri ve satın alma alışkanlıklarına göre gruplara ayırarak, her müşteri grubuna uygun satış stratejileri geliştirmeyi hedeflerler. Yoğunluk kestirimi ise veri dağılımının özetlenmesini içeren danışmansız öğrenme problemi olarak adlandırılır. Yoğunluk kestirimi yaklaşımına bir örnek olarak problem uzayındaki yeni noktaların dağılımını tahmin etmek için yakından ilişkili veri örneklerinden oluşan küçük grupların kullanılmasını içeren çekirdek yoğunluk kestirimi verilebilir (*kernel density estimation*).

Veri kümeleme yöntemleri kullanırken görselleştirme ve boyut indirgeme (*dimensionality reduction*) tekniklerinden de faydalanılmaktadır. Görselleştirme, veri grafikleri oluşturmayı içeren danışmansız öğrenme yaklaşımıdır. Veri kümemizde örnekleri temsil eden çok fazla değişken bulunabilmektedir. Değişken miktarının çok fazla olduğu durumlarda görselleştirme yöntemleri yetersiz kalmaktadır. Bu nedenle veri indirgeme yöntemleriyle verilerin daha az değişkenle temsil edilmesi hedeflenir. Görselleştirme tekniğine bir örnek olarak veri kümesindeki her bir değişken çifti için bir dağılım grafiği oluşturan serpilme diyagramı (*scatter plot*) verilebilir. Veri indirgeme yöntemlerinin en popülerleri ise bir veri kümesinin özdeğerler ve özvektörler cinsinden doğrusal bağımlılıklar kaldırılarak özetlenmesini içeren Temel Bileşen Analizi (*principal component analysis*, PCA)'dir [134].

2.3.3. Yarı Danışmanlı Öğrenme

Danışmansız ve danışmanlı öğrenme problemleri arasındaki farklar keskin çizgilerle ayrılmamış olup her çalışma alanından yararlanan birçok hibrit yaklaşım da vardır. Yarı danışmanlı öğrenme, eğitim verilerinin çok az etiketli örnek ve çok sayıda etiketsiz örnek içerdiği problemlerde kullanılan öğrenme türüdür [105]. Günlük hayatta etiketli veriye ulaşmak her zaman mümkün olmamakta veya etiketleme maliyeti fazla olabilmektedir. Bu nedenle yarı danışmanlı bir öğrenme modelinin amacı, danışmanlı öğrenmede olduğu gibi yalnızca etiketlenmiş verileri değil, mevcut tüm verileri etkili bir şekilde kullanmaktır. Etiketlenmemiş verilerin etkili bir şekilde kullanılması, kümeleme ve yoğunluk kestirimi gibi danışmansız öğrenme yöntemlerinin kullanılmasını veya bunlardan ilham alınmasını gerektirebilir. Danışmansız öğrenme yöntemleriyle veri kümesi içerisinde gruplar veya modeller belirlendikten sonra, danışmanlı yöntemler ile etiketlenmemiş örnekleri etiketlemek veya daha sonra tahmin için kullanılacak etiketlenmemiş temsillere etiket uygulamak için kullanılabilir. Etiketsiz veri kümelerindeki örnekleri etiketlemek için gereken masraf veya hesaplama maliyeti göz önüne alındığında, birçok gerçek dünya danışmanlı öğrenme probleminin yarı danışmanlı öğrenme problemleri ile çözülmeye çalışılması uygulamada oldukça yaygındır. Örneğin resimlerin belirli bir kategoriye ait olup olmadığının sınıflandırılması etiketlenmiş örneklerden oluşan bir veri kümesi gerektirir. Resim sınıflandırma projelerine danışmanlı öğrenme yaklaşımları altında çözüm aranmasına rağmen, resimlerin etiketleme maliyetinden dolayı yarı danışmanlı öğrenme yöntemleriyle sonuç bulmaya çalışılabilmektedir [106]. Bilgisayarla görme (*computer vision*), doğal dil işleme (*natural language processing*) ve otomatik konuşma tanıma (*automatic speech recognition*) alanlarındaki birçok sorun bu kategoriye girmektedir. Yarı danışmanlı öğrenme algoritmalarının model performansına katkısı sadece etiketli veriler ile kurulmuş bir modelin performansı ile karşılaştırılabilir.

2.3.4. Pekiştirmeli Öğrenme

Pekiştirmeli öğrenme, bir temsilcinin (*agent*) bir ortamda çalıştığı ve geri bildirim kullanarak çalışmayı öğrenmesi gereken bir problem sınıfını tanımlar [107]. Pekiştirmeli öğrenme, sayısal bir ödül parametresini en üst düzeye çıkarmak için ne yapılacağını,

durumları eylemlerle nasıl eşleştirileceğini öğrenmeyi hedefler. En basit anlamda bir ödül ceza mekanizması işletilerek zaman içinde modelin performansının artmasının beklendiği söylenebilir. Algoritmaya açık bir şekilde hangi eylemleri yapması gerektiği söylenmez. Bunun yerine hangi eylemlerin en çok ödülü verdiğini deneyerek keşfetmesi gerekir. Pekiştirmeli öğrenmede politika (*policy*), ödül (*reward signal*), durum değeri (*value function*) ve çevre modeli (*environment*) olmak üzere dört ana öge bulunmaktadır. Öğrenme tanımlamalar çerçevesinde gerçekleşir. Pekiştirmeli öğrenmede modelin bazı hedeflere göre ceza ve ödül almasından kaynaklı danışmanlı öğrenmeye benzediği söylenebilir. Pekiştirmeli öğrenme problemine bir örnek, temsilcinin yüksek puan alma hedefinin olduğu ve oyunda hamleler yapabildiği ve ceza ya da ödül açısından geri bildirim aldığı bir oyun oynamak verilebilir. Satranç oyunu oynayan bir model pekiştirmeli öğrenme yaklaşımı ile geliştirilebilir. Yine Google'ın pekiştirmeli öğrenme kullanarak geliştirdiği AlphaGo [108] başarılı sonuçlar vererek gerçek go oyuncularını bile yenmeyi başarmıştır. Q Öğrenme (*Q learning*), derin Q öğrenme (*deep Q learning*) ve geçici fark algoritmaları (*temporal difference algorithms*) pekiştirmeli öğrenmede yaygın kullanılan yöntemlere örnektir.

2.4. MAKİNE ÖĞRENMESİ YÖNTEMLERİ

Bu bölümde tez çalışması kapsamında önerilen yöntemin altyapısını oluşturan ve önerilen yöntemin test edilmesi için kurulacak olan modellerde kullanılan algoritmaların detaylarına yer verilmiştir. Malzeme ve yöntem bölümünde adı geçen algoritmaların kullanım alanlarına yer verilmiştir. Önerilen yöntemin performans ölçümleri için yaygın kullanılan üç farklı sınıflandırma algoritmasıyla modeller kurulup, önerilen yöntemin dengesiz veri kümeleri üzerinde performansları ölçümlenmiştir. Model performans değerlendirme ölçülerine sonraki bölümde yer verilmiştir.

2.4.1. K-En Yakın Komşu Algoritması

K-en yakın komşu algoritması (k-NN), sınıflandırma ve regresyon problemleri için kullanılabilen, veri kümesindeki örnekler arası uzaklığa dayalı danışmanlı bir makine öğrenme algoritmasıdır [109]. Uygulama kolaylığından dolayı araştırmacılar ve sektör çalışanları tarafından yoğunlukla kullanılan ve araştırma projelerinde ilk denenilen algoritmalarındandır. K-en yakın komşu algoritması parametrik değildir, yani model kurulumu

sırasında kullanılan veri kümesi için bir varsayımda bulunmamaktadır. K-en yakın komşu algoritması, eğitim verilerinden ayırt edici bir işlev öğrenmediği ve bunun yerine eğitim veri kümesindeki örnekleri ezberlediği için tembel algoritma (*lazy algorithm*) olarak da adlandırılır. Örneğin, lojistik regresyon algoritması eğitim süresi boyunca model ağırlıklarını (parametrelerini) öğrenip, test verisiyle tahminleme yaparken bu bilgiden faydalanır. Lojistik regresyonun aksine k-en yakın komşu algoritması eğitim boyunca sadece veri kümesindeki örnekleri ezberlemektedir.

K-en yakın komşu algoritması kullanıldığı modelde sırasıyla aşağıdaki adımları uygular [109–111]. Veri kümesinde kategori A ve kategori B olarak adlandırılan iki kategori olduğunu varsayalım. Bir sınıflama problemi örneği olarak yeni bir veri noktasının hangi kategoride olduğu tahminlemek amaçtır. En basit örnek olarak k-en yakın komşu algoritması yeni gelen noktayı sınıflandırmak için kullanılabilir.

Öncelikle uzaklık hesabı yapılırken kullanılacak ve uzaklık hesabı yapılacak komşu noktaların sayısını belirten k değerine karar verilir. Eğitim veri kümesindeki tüm örnekler ile yeni (tahmin edilmesi istenilen) örnekler arasındaki mesafe hesaplanır. Hesaplanan mesafeler artan şekilde yeniden sıralanır. Sıralanan listeden ilk k adet mesafe seçilir. Çözülmek istenen problem türüne göre mesafelerle ilişkili sınıfların modu/ortalaması alınır. Sınıflandırma problemleri için mod, regresyon problemleri için ortalama kullanılır.

K-en yakın komşu algoritmasının başarısı kullanılan uzaklık ölçütüne doğrudan bağlıdır. Belirli bir veri kümesi için en uygun mesafe ölçümlerini seçmek başarılı bir model kurmak için esastır. Aşağıda k-en yakın komşu algoritması kapsamında kullanılabilen çeşitli mesafe ölçümlerine yer verilmiştir [110]:

Minkowski Uzaklığı (Minkowski Distance): Minkowski mesafesi, mesafelerin bir uzunluğa sahip vektörler biçiminde olduğu ve uzunluğun negatif olamayacağı durumlarda hesaplanır. Minkowski uzaklığı Öklid uzayında yer almaktadır. Öklidyen ve Manhattan mesafelerinin daha genelleştirilmiş bir hali olarak düşünülmektedir. **Manhattan Uzaklığı (Manhattan Distance):** İki nokta arasındaki mesafe, kartezyen koordinatlarının mutlak farklarının toplamıdır. **Öklid Uzaklığı (Euclidean Distance):** Öklid uzayında iki nokta arasındaki gerçek düz çizgi mesafesinin bir ölçüsüdür. Öklid uzaklığı aynı zamanda k-en yakın komşu algoritmasında en sık başvuru uzaklık ölçütüdür. **Kosinüs Uzaklığı (Cosine Distance):** İki vektör arasındaki benzerliği hesaplamak için kullanılır. Kosinüs

fonksiyonu kullanılarak yapılan ölçüm yönü ölçer ve iki vektör arasındaki açıyı hesaplar. **Jaccard Uzaklığı (Jaccard Distance):** Tüm veri kümesinde dağıtılan bir özneliği karşılaştırmasından dolayı kosinüs uzaklığı ile benzerlik gösterir. Jaccard uzaklığı, iki farklı veri kümesine bakarak ve her iki veri kümesinde de değerin bire eşit olduğu örnekleri bulur.

K-en yakın komşu algoritması ile kurulacak modelin performansını etkileyen en önemli parametre komşu örnek sayısını belirten **k** değeridir. K sayısı seçilirken k değeri kullanılmamalıdır. K değerinin küçük seçilmesi aşırı öğrenmeye (*overfitting*) sebep olabilmektedir. Model eğitim aşamasında iyi performans gösterirken, aşırı öğrenme sonucu olarak aynı model test aşamasında kötü performans gösterebilmektedir. Yüksek bir k değeri seçmek ise eksik öğrenmeye (*underfitting*) neden olabilmektedir. Eksik öğrenme durumunda model hem eğitim hem de test aşamasında düşük performans gösterebilmektedir. İkili sınıflandırma problemlerinde k değeri seçilirken çift değerlerden uzak durulması gerekmektedir. Böyle bir durumda sınıflar eşit olarak dağıldığında model veri noktasını bir sınıfa atayamamaktadır. K değerine ağırlıklı olarak alandaki uzmanların bilgisine göre karar verilir. Alan uzmanının olmadığı veya k değerine karar verilemediği durumlarda dirsek yöntemine (*elbow method*) başvurulmaktadır.

K-en yakın komşu algoritmasının en büyük avantajı anlaşılmasının ve kullanılmasının kolay olmasından gelmektedir. Bunun dışında özellikle küçük boyutlu veri kümeleri için iyi sonuçlar verdiği gözlemlenmiştir. K-en yakın komşu algoritması eğitim aşamasında öğrenmediğinden dolayı eklenen yeni veri noktaları model performansını da etkilemektedir.

Ancak k değerinin belirlenmesinin zor olması k-en yakın komşu algoritmasının bilinen en büyük dezavantajları arasında yer almaktadır. Algoritma, farklı ölçekte girdi değişkenlerinin olduğu veri kümelerinde de kötü sonuçlar vermektedir. Bunun altında yatan en büyük neden uzaklık ölçütlerine bağlı çalışmasıdır. Veri boyutunun büyüdüğü durumlarda algoritmanın çalışma süresinin uzamasından dolayı tercih edilmemektedir. K-en yakın komşu algoritması aynı zamanda eksik verilere ve aykırı noktalara (*outliers*) karşı hassas olmasıyla da bilinmektedir.

2.4.2. Destek Vektör Makineleri

Destek Vektör Makinesi (*support vector machines*, SVM), hem sınıflandırma hem de regresyon problemleri için kullanılabilen ve sıklıkla tercih edilen danışmanlı makine

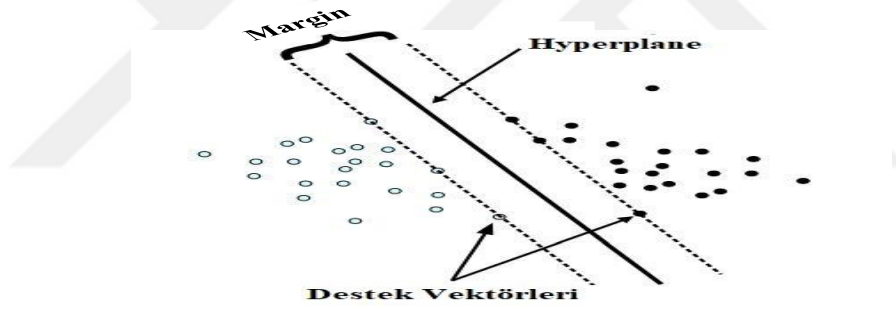
öğrenimi algoritmasıdır. Kapsamlı bir literatür taraması sonucunda daha çok sınıflandırma problemleri için kullanıldığı görülmektedir. SVM algoritmasında, her veri ögesini n (her bir değişkeni ifade eden girdi özniteliklerinin sayısı) boyutlu uzayda bir nokta olarak çizeriz ve her bir özelliğin değeri belirli bir koordinatın değerine eşit olur. Daha sonra sınıflandırma problemlerinde iki sınıfı birbirinden çok iyi ayıran aşırıdüzlem (*hyperplane*) bularak sınıflandırma gerçekleştirilir. 1990'larda geliştirilen destek vektör makinelerinin bulunduğu dönemdeki popülerliği günümüzde de devam etmektedir.

SVM'in küçük ve karmaşık veri kümeleriyle daha iyi sonuçlar ürettiği gözlemlenmiştir. Destek vektör makinelerinin özellikle büyük veriler ile çalışma süreleri diğer algoritmalara göre daha uzun olabildiği için öncelikle lojistik regresyon gibi daha hızlı sonuç üreten algoritmalar ile model kurulumu denenmektedir. Herhangi bir çekirdek yöntemi (*kernel trick*) kullanılmadan kurulacak SVM tabanlı modelin başarımı lojistik regresyon ile kurulan modelle benzer sonuçlar vermektedir. Bu nedenle öncelikle daha hızlı ve basit modeller kurmayı sağlayan algoritmaların kullanılması tercih edilmektedir.

Destek vektör makineleri kullandıkları çekirdek yöntemine göre farklı isimler almaktadırlar. SVM'de kullanılan çekirdek yöntemlerinin başarısı, kullanılan verinin doğrusal ayrılıp ayrılmadığına doğrudan bağlıdır [112]. Veriler doğrusal olarak ayrılabilir olduğunda, doğrusal SVM'yi (*linear SVM*) kullanımı tercih edilir. Verilerin doğrusal olarak ayrılması, veri noktalarının tek bir düz çizgi kullanılarak iki sınıfa ayrılabilceği anlamına gelmektedir. Verilerin doğrusal olarak ayrılamadığı durumlarda doğrusal olmayan destek vektör makineleri (*non-linear SVM*) tercih edilir.

İkili sınıflandırma problemi için geliştirilen SVM'in [113] çalışma prensibinde kullanılan temel terimler Şekil 2.13'te gösterilmektedir. Şekil 2.13'te gösterilen destek vektörlerindeki noktalar aşırıdüzleme en yakın noktalardır. Bu veri noktaları yardımıyla bir ayırma çizgisi tanımlanarak ikili sınıflandırmada iki sınıf birbirinden ayırt edilmektedir. Aşırıdüzlem ile aşırıdüzleme en yakın gözlemler arasındaki mesafe ise *margin* olarak adlandırılır. SVM'de büyük margin, iyi bir margin olarak kabul edilir. *Margin* de hard ve soft margin olmak üzere ikiye ayrılmaktadır. Destek vektör makinelerinde oluşturulan marginler her zaman Şekil 2.13'teki gibi olmamaktadır. Bazen veri kümesindeki bazı örnekler *margin* olarak belirlenen aralıkta yer alabilmektedir. Bu tarz marginler *soft margin* olarak adlandırılırken, sınıfları birbirinden tamamen ayıran marginler *hard margin* olarak adlandırılır.

Destek vektör makineleri, yalnızca destek vektörleri ile temsil edilir. Bu nedenle, *margin* tanımlanırken diğer gözlemler de göz önünden bulundurulduğundan destek vektör makineleri ifade edilirken diğer noktaların tanımlamaya doğrudan etkisi olmaz. Şekil 2.13'teki gibi bir veri kümesindeki iki sınıfı ayırmayı amaçlayan bir problemimiz mevcut ise bu noktaları sınıflandırmak için birçok karar sınırimız olabilir, ancak asıl soru hangisinin en iyi olduğu ve bunu nasıl bulacağımızdır. İki boyutlu bir uzayda bu sınıfları ayıran karar sınırimız bir doğru ile ifade edilirken çok boyutlu uzayda bir aşırıdüzlem ile ifade edilmektedir. Destek vektör makinelerinde en iyi aşırıdüzlem, her iki sınıftan da maksimum uzaklığa sahip olan düzlem olarak belirtilir. En iyi aşırıdüzlem, sınıfları en iyi şekilde birbirinden ayıran farklı aşırıdüzlemler bularak tespit edilir. Daha sonra bulunan aşırıdüzlemlerden, veri noktalarından en uzak olanı veya maksimum *margin*e sahip olanı seçilir. Destek vektör makinelerinin bu kadar popüler olmasının arkasında yatan en büyük neden, doğrusal olmayan verilerle de model kurulmasını sağlayan çekirdek fonksiyonlarıdır.

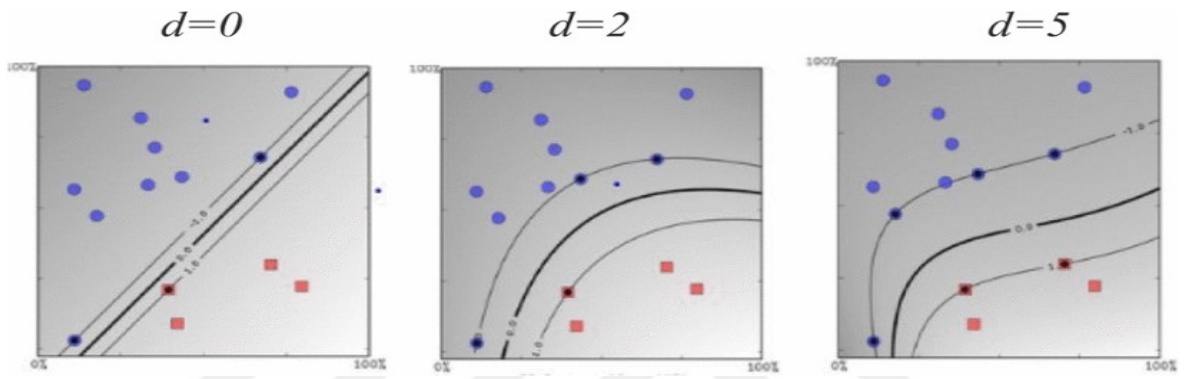


Şekil 2.13: Destek vektör makineleri doğrusal aşırıdüzlem örneği [114].

Verileri bir doğru veya bir aşırıdüzlem ile ayıramadığımız durumlarda veri noktalarını net bir şekilde bölen bir karar sınırı (*decision boundary*) bulmamızı sağlayacak bazı ikinci dereceden fonksiyonları kullanarak, bu daha düşük boyutlu uzayı daha yüksek boyutlu bir uzaya dönüştürmek amaçlanır. Bu fonksiyonlar çekirdek fonksiyonları olarak anılmaktadır. Çalışmlarda kullanılan en yaygın çekirdek fonksiyonları polinomial çekirdek fonksiyonu (*polynomial kernel*), sigmoid çekirdek fonksiyonu (*sigmoid kernel*) ve radyal tabanlı fonksiyon (*radial basis function kernel*)'dur [110, 115].

Polinomial çekirdek fonksiyonu, orijinal değişkenlerin polinomları üzerinden bir özellik uzayında eğitim örneklerinin benzerliğini temsil eder [115]. Polinomial çekirdek fonksiyonu, hem girdi örneklerinin verilen özelliklerine hem de bunların etkileşim özellikleri

olarak adlandırılan kombinasyonlarına bakar. Şekil 2.14'te farklı derecelerden polinomial fonksiyonların grafiklerine yer verilmiştir. Sigmoid çekirdek fonksiyonu yapay sinir ağlarında da sıklıkla kullanılan diğer bir fonksiyondur. Özellik vektörlerinin sayısının yüksek olduğu veya 2 boyutta doğrusal olmayan karar sınırının olduğu problemlerde, sigmoid çekirdek fonksiyonu radyal tabanlı fonksiyon kadar iyi olabilmektedir [115]. Radyal tabanlı fonksiyon ise iki yer işareti arasındaki öklid mesafesini serbest bir parametre bir benzerlik fonksiyonudur. Radyal tabanlı fonksiyon destek vektör makinelerinde en yaygın olarak kullanılan çekirdek fonksiyonudur.



Şekil 2.14: Farklı derecelerden polinomial çekirdek fonksiyonları [115].

Destek vektör makinelerinde modelin başarısı seçilen çekirdek fonksiyonuna bağlı olduğundan doğru çekirdek fonksiyonunu seçmek önemlidir. Çekirdek fonksiyonunun seçimi ise kullanılan veri kümesi ile ilişkilendirilmektedir. Eğer veri kümesi doğrusal olarak ayrılabilir ise, kullanımı çok kolay olduğu ve karmaşıklığı diğer çekirdek fonksiyonlarına kıyasla çok daha düşük olduğu için doğrusal çekirdek fonksiyonu seçilmelidir. Doğrusal çekirdek fonksiyonlarının eğitim süresi de kısa olduğundan öncelikle doğrusal çekirdek fonksiyonuyla çalışmalara başlanılmalıdır.

2.4.3. Rastgele Orman Algoritması

Rastgele Orman algoritması (*random forest*) hem regresyon hem de sınıflandırma problemleri için sıklıkla kullanılan bir diğer danışmanlı öğrenme algoritmasıdır. Rastgele orman algoritması varsayılan parametreler ile de başarılı sonuçlar verebilmektedir. Bir başka deyişle hiperparametre ayarlamasına (*hyperparameter tuning*) ihtiyaç duyulmadan da bazı problemlerin çözümünde kullanılabilir. Rastgele Orman algoritması, farklı örnekler üzerinde bir dizi karar ağaçları (*decision trees*) oluşturur.

Rastgele Orman algoritması danışmanlı öğrenme ailesinin bir alt grubu olan topluluk öğrenme (*ensemble learning*) yaklaşımları altında yer alır [116]. Topluluk öğrenme yaklaşımları birden fazla zayıf öğrenicinin sonuçlarını kullanarak güçlü bir model oluşturmayı hedefler. Topluluk öğrenme yaklaşımları torbalama (*bagging*) ve arttırma (*boosting*) denilen iki yöntemden biri ile geliştirilebilmektedir [117].

Bir veri kümesini kullanarak farklı modeller kurup ve sonrasında model sonuçlarını birleştirmek model performansını arttırmada her zaman etkili bir yöntem olmayabilmektedir. Bunun yerine, *bootstrapping* (önyükleme) adı verilen bir yöntem kullanarak, modellere girdi sağlayan veri kümelerinde farklılık yaratılması hedeflenir. *Bootstrapping* yöntemiyle orijinal veri kümesinin alt kümeleri oluşturulur ve her bir alt veri kümesi ile farklı bir model kurulur. Oluşturulan veri kümelerindeki eleman sayısı birbirine eşit ve ana veri kümesinin eleman sayısından küçüktür. Torbalama yöntemi ile aşırı öğrenme (*overfitting*) probleminin etkisi azaltılır. Torbalama yöntemi aynı zamanda yüksek boyutlu veri kümeleriyle çalışmaya olanak sağlar ve model performansını artırır.

Arttırma tekniği, her modelin bir önceki modelin hatalarını düzeltmeye çalıştığı sıralı bir süreçtir. Her model kendinden önceki modelin hatalarından öğrenerek model performansının iyileşmesi beklenmektedir. XGBoost günümüzde arttırma yaklaşımına dayanan en popüler algoritmalarındandır [118]. Farklı sektörlerde birçok problemin çözümü yüksek performansı nedeniyle XGBoost ile gerçekleştirilmektedir.

Rastgele orman algoritması torbalama prensibiyle çalışır. Rastgele orman algoritması, bir ağaca bağlı kalmak yerine her ağaçtan tahmin alır ve tahminlerin çoğunluk oylarına dayanarak nihai çıktıyı tahmin eder. Rastgele orman algoritması temelde bir zayıf öğrenici olan karar ağaçlarını kullanır. Karar ağaçları da hem sınıflandırma hem de regresyon problemleri için kullanılabilen bir danışmanlı öğrenme algoritmasıdır [119]. Karar ağaçları, bir dizi özellik tabanlı bölmeden kaynaklanan tahminleri göstermek için ağaç yapısı gibi bir akış şeması kullanır. Karar ağaçlarının ilk bölmesi kök (*root node*) olarak adlandırılır. Kökte yer alan koşula göre alt kırımlar oluşur. Bu alt kırımlar düğüm (*node*) olarak adlandırılır. Karar ağaçlarının son kırımlarında ise yapraklar (*leaf nodes*) bulunur. Bu bölüm aynı zamanda çözmek istenilen problemin sonucunun yer aldığı kısımdır. Karar ağaçlarında düğüm sayısı arttıkça modelin karmaşıklığı da artmaktadır. Karar ağaçlarında kök ve düğümler belirlenirken veri kümesini en iyi derecede açıklayacak değişkenler kullanılır.

Daha fazla bölünecek bir özellik seçmek için, bu bölünmenin ne kadar saf olacağını bilmemiz gerekir. Yeni bölünmenin ölçülmesinde kullanılan yaygın değerlerden biri gini değeridir (*gini index*). Gini değeri temelde rastgele seçilmiş belirli bir özelliğin yanlış sınıflandırılma olasılığını hesaplar. Gini değerine alternatif olarak kullanılabilen diğer bir değer ise entropidir (*information entropy*). Entropi de hangi özelliğin bir sınıf hakkında maksimum bilgi verdiğini belirlemek için kullanılmaktadır [120]. Entropi karar ağacının kök düğümlerinden yaprak düğümlerine kadar olan entropi düzeyini düşürmeyi amaçlar. Gini değeri ile entropi kullanımı arasında küçük bir fark vardır. Gini değeri daha fazla örnek olan hedef değişkeni ayırıştırma eğilimindedir. Entropi ise her iki sınıfı da iyi tahminlemeyi hedefleyen daha dengeli bir model elde edilmesini sağlamaktadır.

Karar ağaçları ve rastgele orman algoritması arasındaki temel fark, rastgele orman algoritması tahmin yapmak için orijinal veri kümesinin bir alt kümesini kullanan torbalama yöntemini kullanarak karar ağaçlarında oluşabilecek aşırı öğrenme probleminin üstesinden gelmeye yardımcı olmasıdır. Rastgele orman algoritması aynı zamanda karar ağaçlarının fazla büyümesinden ortaya çıkacak yüksek varyans probleminin çözülmesinde de yardımcı olmaktadır. Yüksek varyansın olduğu durumlarda model eğitim veri kümesinde iyi performans gösterirken, yeni bir veri noktası tahmin edildiğinde model performansı başarısız olmaktadır. Bu nedenle başarılı bir model kurulumu için yüksek varyans probleminden kaçınılmak istenmektedir. Karar ağaçları ile oluşturulan model, modele girdi olarak verilen değişkenler kullanılarak formüle edilebilmektedir. Rastgele orman algoritması ise birçok karar ağacının onucunu kullanarak tahmin oluşturduğu için girdi değişkenleri ile formüle edilememektedir. Rastgele orman algoritması veri kümelerinde yer alan eksik veri problemiyle de başarılı tahmin sonuçları verebilmektedir.

2.4.4. K-ortalamlar Kümeleme Yöntemi

Şu ana kadar anlatılan k-en yakın komşu, destek vektör makineleri ve rastgele orman algoritmaları makine öğrenmesi altında, danışmanlı öğrenme grubunda yer alan algoritmalar. Bu algoritmalar önerilen dengesiz veri problemi çözümü ile birlikte kullanılarak modeller oluşturulup, önerilen yöntemin performansı ölçümlenecektir. K-ortalamlar (*k-means*) ise önerilen yöntemde kullanılan bulanık c-ortalama (*fuzzy c-means*) kümeleme yöntemiyle karşılaştırma amacıyla kullanılacaktır. En temel tanımıyla kümeleme, verilerdeki kalıpları tanımlama ve gruplara bölme işlemi olarak adlandırılan bir

danışmansız öğrenme yaklaşımıdır [121].

K-ortalamlar kümeleme yöntemi, her yinelemede yerel maksimumları bulmayı amaçlayan bir kümeleme algoritmasıdır. Algoritma 5 aşamada özetlenebilir [122]. Öncelikle istenilen k küme sayısına karar verilir. Her veri noktası rastgele bir kümeye atanır. Her bir küme için o kümeye atanan verilerden faydalanılarak küme merkezleri hesaplanır. Belirlenen küme merkezlerine göre veri kümesindeki her nokta kendisine en yakın küme merkezine yeniden atanır. Her örnek yeni veri kümelerine atandıktan sonra küme merkezleri tekrardan hesaplanır. Hiçbir iyileştirme mümkün olmayana kadar son iki adım tekrarlanır. Ardışık iki tekrar için iki küme arasında veri noktalarında daha fazla değişme olmayacağı zaman algoritma sonlanır.

Kümeleme yöntemlerinde başarımın ölçülmesi için oluşturulan kümelerin bazı özellikleri taşıması beklenir. Bir veri kümesi ile oluşturulan kümelemede, her bir kümedeki tüm veri noktaları birbirine benzer olmalıdır ve bu özellik *inertia* metriğiyle değerlendirilir. *Inertia* bir küme içindeki tüm noktaların o kümenin merkezine olan uzaklıklarının toplamını hesaplar. *Inertia* değeri ne kadar düşükse, kümeleme sonucu da o kadar iyidir. Her küme içerisindeki noktaların birbirine yakın olması gerektiği kadar, farklı kümelerde yer alan noktaların birbirine uzaklıkları fazla olmalıdır.

2.4.5. Bulanık C-ortalama Kümeleme Yöntemi

Bulanık C-ortalama (*Fuzzy C-means*) yöntemi k-ortalamlar yöntemi gibi bir danışmansız öğrenme yöntemidir [123]. Bulanık C-ortalama yönteminde yer alan c parametresi k-ortalamlar kümeleme yöntemindeki k sayısına benzerdir ve küme merkezlerini ifade etmektedir. Bir kümeleme algoritmasında, bir veri noktasının bir kümeye ait olma olasılığı yalnızca bir veya sıfır değerini alabiliyorsa bu kümeleme *hard clustering* olarak adlandırılır [124]. *Hard clustering* yöntemlerinde küme sınırları net bir şekilde belirgindir. *Hard clustering* yöntemlerinin aksine soft clustering grubuna giren kümeleme yaklaşımları, bir kümeye ait bir veri noktasının olasılığı sıfır ile bir arasında herhangi bir değer alabilir. Bu nedenle bir kümenin sınırı bulanık bir sınır olarak ifade edilir. Bir veri noktası küme merkezine ne kadar yakınsa, o küme merkezine üyeliği de o kadar fazladır ve her veri noktasının kümelere olan üyelik oranları toplamı bire eşit olmalıdır.

Bulanık c-ortalama algoritması, çözümün bulanıklık derecesine bağlıdır. Büyük bulanıklık

dereceleri sınıfları bulanıklaştırır ve tüm öğeler tüm kümelere ait olma eğilimindedir. Optimizasyon probleminin çözümleri bulanıklık derecesi parametresine bağlıdır. Bulanık kümelemede, geleneksel k-ortalamlar yönteminde olduğu gibi tamamen tek bir kümeye ait olmak yerine, her noktanın her bir kümeye ait olma olasılığı vardır. Bununla birlikte bulanık c-ortalama yöntemi, daha fazla hesaplamada bulunduğu için k-ortalamlar kümeleme yönteminden daha yavaş çalışma eğiliminde olacaktır. Bulanık c-ortalama yönteminde her nokta, her küme ile birlikte değerlendirilir ve her değerlendirmede daha fazla işlem yer almaktadır. K-ortalamlar kümeleme yönteminde ise sadece bir uzaklık metriği hesaplanmaktadır.

2.5. MODEL PERFORMANS ÖLÇÜLERİ

Bu bölümde model performansı ölçülmesinde kullanılan farklı ölçütlerin tanımlamalarına yer verilecektir. Değerlendirme ölçütleri bir olasılık eşik değerine bağımlı olup olmadıklarına göre iki grupta incelenebilir. Bir olasılık eşik değerinden bağımsız hesaplanan ölçütler Kesinlik-Duyarlılık Eğisi (*Precision-Recall*) ve alıcı işletim karakteristik eğrisi (*Receiver Operating Characteristic Curve*, ROC Curve) ile örneklendirilebilir. Bir olasılıksal eşik değerine bağımlı ölçütler ise doğruluk (*accuracy*) ve karmaşıklık matrisinden (*confusion matrix*) elde edilen değerler ile örneklendirilir. Üzerinde çalışılan problem türüne ve hedeflere göre bu değerlendirme ölçütlerinden bir veya birkaçı bir arada kullanılarak kurulan modellerin performansları değerlendirilebilir. Her bir değerlendirme ölçütünün kendi içinde avantaj ve dezavantajları olmasından dolayı uygulamada çoğu zaman birden fazla değerlendirme ölçütüne bakılarak model başarısına karar verilir.

2.5.1. Karmaşıklık Matrisi

Karmaşıklık matrisi, bir sınıflandırma modelinin performansını ölçümlemek için kullanılan bir yöntemdir. Karmaşıklık matrisi sınıflandırma modelinin neyi doğru yaptığı ve ne tür hatalar yaptığı konusunda bilgi verir. Karmaşıklık matrisi, gerçek pozitiflerin, yanlış pozitiflerin ve doğruların sayısını içeren bir matristir. Karmaşıklık matrisinden faydalanılarak model performansının değerlendirilmesinde kullanılan diğer ölçütler hesaplanmaktadır. Tez kapsamında önerilen çözümler ikili sınıflandırma problemine yönelik

olduğundan dolayı değerlendirme ölçütleri de ikili sınıflandırma problemi için anlatılacaktır. Ancak aynı değerlendirme ölçütlerini çok sınıflı sınıflandırma problemleri için de uyarlamak mümkündür.

Karmaşıklık matrisi, doğru ve yanlış sınıflandırılan örnek sayısının görselleştirilmesidir. Tek başına bir değerlendirme ölçütü olarak kullanılmaz. Tablo 2.4'te karmaşıklık matrisinin gösterimi yer almaktadır. Örneğin, bu gösterimde aslında negatif olan gözlemlere karşı negatif veya pozitif olarak tahmin edilen gözlemleri ve pozitif olan gözlemlere karşı negatif veya pozitif tahmine sahip olanları görebiliriz. Gözlemin gerçek sınıfı pozitifse ve pozitif olarak tahmin edildiyse, bu doğru pozitif (*true positive*, TP) olarak adlandırılır. Aynı şekilde gözlemin gerçek sınıfı negatifse ve negatif olarak tahmin edildiyse, bu doğru negatif (*true negative*, TN) olarak adlandırılır. Yani karmaşıklık matrisinde köşegende doğru tanımlanmış gözlemler görünür [125, 126].

Ancak gözlem negatifse ve pozitif olarak tahmin ediliyorsa, o zaman yanlış pozitif (*false positive*, FP) olarak adlandırılır. Eğer gözlem pozitifse ancak model negatif olduğunu söylüyorsa, o zaman yanlış negatif (*false negative*, FN) olarak adlandırılır.

Tablo 2.4: Karmaşıklık matrisi.

	Tahmin Pozitif	Tahmin Negatif
Gerçek Pozitif	Doğru Pozitif (TP)	Yanlış Negatif (FN)
Gerçek Negatif	Yanlış Pozitif (FP)	Doğru Negatif (TN)

2.5.2. Doğruluk

Doğruluk (*accuracy*, ACC), doğru tahminlerin yüzdesini veya oranını temsil eden ve karmaşıklık matrisinden elde edilen bir model değerlendirme ölçütüdür. Doğruluk, doğru tahmin sayısının toplam tahmin sayısına bölünmesiyle elde edilir. İkili sınıflandırma problemi için bu formül, gerçek pozitiflerin toplam sayısı ve gerçek negatiflerin toplam sayısının toplam tahmin sayısına bölünmesi olarak da ifade edilebilir. Yüksek doğruluk oranları model performansının da yüksek olduğu anlamına gelir. (**Doğruluk (ACC)**= (TP + TN)/(TP + TN + FP +FN))

Doğruluk makine öğrenmesi modellerinin değerlendirilmesinde en sık kullanılan ölçütlerden biri olmasına rağmen, dengesiz veri küme probleminin gözlemlendiği durumlarda en iyi

değerlendirme yöntemi olarak görülmemektedir. Dengesiz veri kümeleri için, farklı hedef sınıfların örneklerinin doğru sınıflandırılmış sayıları arasında ayırım yapmadığından dolayı doğruluk model performans değerlendirmesi için tercih edilmez. Hedef sınıflarının örnek sayılarının dengesiz olduğu durumlarda, azınlık hedef sınıfının az örnek sayısına sahip olmasından dolayı genel doğruluk değeri üzerinde de çok az etkisi olduğu düşünülmektedir.

2.5.3. Duyarlılık

Makine öğrenmesi modellerinin performanslarının ölçülmesinde doğruluk metriğiyle birlikte kullanılan diğer bir metrik ise duyarlılık ya da diğer adıyla hassasiyettir (*recall, sensitivity, true positive rate/TPR*). Duyarlılık, model performans değerlendirmesinde dengesiz veri kümeleri için daha uygun bir metriktir. Çünkü duyarlılık bir veri kümesindeki sınıfların her biri ve özellikle de azınlık sınıfı için modelin ne durumda olduğunu biraz daha iyi gösterir. Duyarlılık, toplam pozitif örneklerden, model tarafından kaç tanesinin doğru olarak tahmin edildiğini gösterir. Duyarlılık, gerçek pozitiflerin toplam sayısını gerçek pozitiflerin ve yanlış negatiflerin toplamına böldüğümüz formülle hesaplanır. Doğruluk metriğinde olduğu gibi yüksek duyarlılık değerleri daha iyi modelleri temsil etmektedir. (**Duyarlılık**= $TP/(TP+FN)$)

2.5.4. Belirleyicilik

Duyarlılık bize veri kümemizdeki azınlık sınıfı hakkında bilgi verirken, belirleyicilik (*specificity, true negative rate/TNR*) ise aynı hesaplamanın çoğunluk sınıfı için gerçekleştirilene verilen addır. Kısaca belirleyicilik, gerçek negatif sınıftaki örneklerin ne kadar iyi tahmin edildiğinin bir göstergesidir. Model tarafından doğru bir şekilde negatif olarak sınıflandırılan toplam gözlemlerin, yanlış pozitiflerin ve doğru negatiflerin sayısının toplamına bölünmesi ile hesaplanır. (**Belirleyicilik**= $TN/(TN+FP)$)

2.5.5. Kesinlik

Kesinlik (*precision, positive predictive value*), bir veri kümesinde doğru sınıflandırılmış pozitif örneklerin, doğru veya yanlış pozitif olarak sınıflandırılmış toplam örnek sayısına oranı olarak tanımlanır. Model değerlendirmesi sonucu elde edilen yüksek kesinlik değerleri, aynı zamanda model başarısının yüksek olduğunu gösterir. Dengesiz veri kümelerinde

genellikle azınlık sınıfının tahmini daha önemli olduğu için, duyarlılık ve kesinlik model performans değerlendirmesinde tercih edilen iki ana metriktir. (**Kesinlik**= TP/(TP+FP))

Kesinlik makine öğrenmesi modelinin pozitif sınıfı tahminleme kabiliyeti hakkında bilgi verirken, duyarlılık pozitif tahminlerin gerçeği ne kadar yansıttığını göstermektedir. Kesinlik hesaplanırken veri kümesinde yer alan hem pozitif hem de negatif sınıf örnekleri göz önüne alınmaktadır. Ancak duyarlılık hesaplanırken sadece pozitif sınıf örnekleri kullanılır ve negatif sınıf örnekleri göz ardı edilir. Duyarlılık doğru tahmin edilen pozitif sınıf örneklerini göz önüne alırken, modelin yaptığı yanlışları göz önüne almamaktadır.

Oluşturulan bir makine öğrenmesi modelinin performansı ölçümlenirken hem duyarlılık hem de kesinliğin yüksek değerler alması beklenir. Kesinlik ve duyarlılık sıfır ve bir arasında değerler alırlar. Sonuç bire ne kadar yakınsa modelin başarısı da o kadar yüksektir. Hem duyarlılık hem de kesinlik olasılık eşik değerine bağlı hesaplanan ölçütlerdir. Olasılık eşik değeri, bir örneğin azınlık sınıfına ait olduğu düşünülen olasılık değeridir. Kesinlik ve duyarlılık değerleri, sınıf çıktısını belirlemek için kullanılan eşik değere bağlı olarak değişmektedir.

2.5.6. F-ölçütü

F-ölçütü (*F1-score*, *F-measure*), duyarlılık ve kesinliğin harmonik ortalaması alınarak hesaplanan bir model performans değerlendirme ölçütüdür. Matematiksel olarak, kesinlik ve duyarlılık çarpımının iki katı alınarak, kesinlik ve duyarlılık toplamına bölünmesiyle elde edilir. F-ölçütünü en üst düzeye çıkarmak, kesinlik veya duyarlılık değerinin optimize edildiği anlamına gelir. F-ölçüsünün değerleri de sıfır ile bir arasında değişir ve bir alabileceği en iyi değerdir. F-ölçütünün en iyi değere getirilmesi duyarlılık ve kesinlik arasında en iyi dengenin sağlandığı anlamına da gelmektedir.

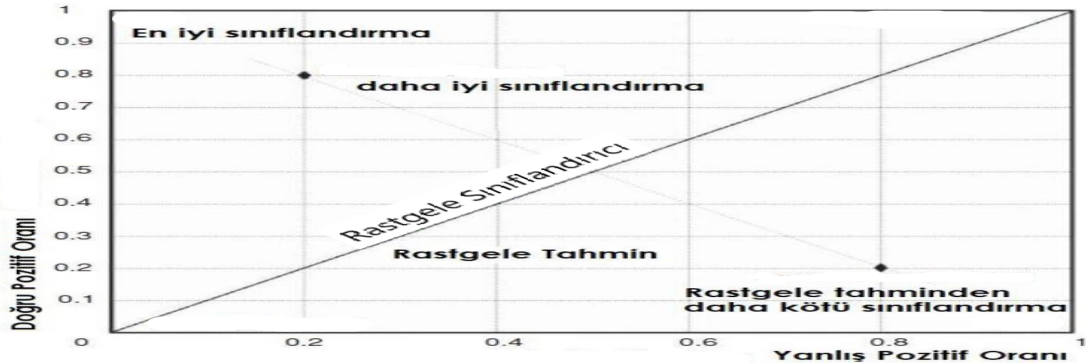
$$\mathbf{F\text{-}ölçütü} = \frac{2 \times \text{Duyarlilik} \times \text{Kesinlik}}{\text{Duyarlilik} + \text{Kesinlik}} \quad (2.1)$$

2.5.7. Alıcı İşlem Karakteristik Eğrileri

Alıcı işlem karakteristik eğrisi (*receiving operating characteristic*, ROC Curve) doğru pozitif oranını y ekseninde ve yanlış pozitif oranını x ekseninde gösteren bir eğriyi temsil eder.

Eğrideki her nokta, bir örneğin pozitif sınıfa ait olup olmadığını belirlemek için bir eşik olarak kullandığımız bir olasılık eşiğini temsil eder. Alıcı işlem karakteristik eğrisi ile belirli bir örnek için doğru pozitif oranı ile yanlış pozitif oranı arasındaki ödünleşim bulunur. Eğri üzerinde hareket edilerek farklı bir olasılık eşiği seçilirse, farklı bir doğru pozitif oranı ve yanlış pozitif oranı elde edilir.

Alıcı işlem karakteristik eğrisi, bir sınıflandırıcının pozitif örnekleri negatif örneklerden ne kadar iyi ayırabildiğini değerlendirir. Bununla birlikte, farklı olasılık eşiklerine göre hesaplama yaptığı için, pozitif ve negatif sınıfları ayıran eşiğin ne olduğunu belirlemeye yardımcı olur. Daha yüksek derecede yanlış pozitifleri kabul etmek isteniliyorsa, daha büyük doğruluklar elde edebilmek için daha küçük eşikler seçebilir. Ancak yanlış pozitiflerin yüksek sayıda olması istenmiyorsa, o zaman yanlış pozitif oranını en aza indirmek için daha yüksek olasılık eşik değerlerini seçeriz. Bu durumda doğru pozitif oranı da daha düşük olmaktadır. Şekil 2.15'te örnek bir alıcı işlem karakteristik eğrisi temsil edilmektedir. Alıcı işlem karakteristik eğrisi aslında üç boyutlu bir grafik olduğu düşünülebilir. Grafikte doğru pozitif oranı, yanlış pozitif oranı ve üçüncü boyut olarak farklı olasılık eşikleri temsil edilir. Grafikteki olasılık eşik değeri sağa doğru azalmaktadır. Örneğin, grafiğin solunda bir gibi çok yüksek bir olasılık eşik değeri mevcuttur. Ve grafiğin sağına doğru ilerledikçe, sıfıra ulaşana kadar olasılık eşik değerleri giderek küçülmektedir. Olasılık eşik değeri sıfır seçildiği durumda, bütün örneklerin pozitif sınıfa ait olduğu tahminlenmektedir. Bu durumda duyarlılık bir değerine eşit olacaktır. Eğer olasılık eşik değeri bir ise veya bire yaklaşırsa, o zaman duyarlılık sıfıra yaklaşır. Çünkü, veri kümesindeki çok az örnek doğru bir şekilde sınıflandırılacaktır.



Şekil 2.15: Alıcı işlem karakteristik eğrisi [110, 127].

2.5.8. Eğri Altında Kalan Alan

Alıcı işlem karakteristik eğrisi altında kalan alan (AUC), olası tüm sınıflandırma eşiklerinde modelimizin performansının başka bir ölçüsünü sağlar. Eğri altında kalan alan ne kadar yüksek ise, model her iki sınıfı da o kadar iyi tahmin eder. Mükemmel bir modelde eğri altında kalan alanın bire eşit olması beklenir. Aksine, model performansı kötü ise eğri altında kalan alanın sıfıra yaklaştığı gözlemlenir. Rastgele model, her bir olasılık eşiği için aynı sayıda doğru pozitif ve yanlış pozitif verir. Bu durumda hesaplanacak eğri altındaki alan 0,5 olacaktır

2.6. DENGESİZ VERİ KÜMELERİ ÇÖZÜMLERİ

Bu bölümde önerilen yöntemin karşılaştırılması için temel alınacak ve literatürde yaygın şekilde kullanılan çözümler incelenecektir. Hem yukarı örnekleme (*oversampling*) hem de aşağı örnekleme (*undersampling*) çözümlerine yer verilmiştir. Aşağı örnekleme yöntemlerinden rastgele aşağı örnekleme ve en yaygın kullanımı olan yukarı örnekleme çözümlerinden SMOTE ve Borderline-SMOTE ele alınmıştır.

2.6.1. Rastgele Aşağı Örnekleme

Aşağı örnekleme, çoğunluk sınıfından gözlem sayısını azaltma sürecini ifade eder. Yani tipik bir aşağı örnekleme senaryosunda, çoğunluk sınıfından gelen gözlem sayısı azalan bir veri kümesi elde ederiz. Çoğunluk sınıfının veri kümesindeki varlığının ne kadar azaltılacağı ile ilgili farklı yaklaşımlar bulunmaktadır. Aşağı örnekleme yöntemleri de iki temel grup altında incelenebilir. Birinci grupta yer alan aşağı örnekleme yaklaşımları, çoğunluk sınıfını azınlık sınıfının aynı sayıda gözlemine indirgeyen yöntemlerdir. Rastgele aşağı örnekleme bu grupta yer alan yaklaşımlar içerisinde yer almaktadır. İkinci grupta yer alan aşağı örnekleme yaklaşımları ise çoğunluk sınıfını farklı koşullara bağlı olarak indirgemektedir.

Rastgele aşağı örneklemede çoğunluk sınıfından gözlemler, belirli bir kriter sağlanana kadar rastgele alınır [128]. Rastgele aşağı örnekleme basit bir teknik olup, veri kümesi hakkında bir varsayımda bulunmaz. Kısaca, biri çoğunluk sınıfı ve diğeri azınlık sınıfı olmak üzere iki sınıflı bir hedef varsa, rastgele aşağı örnekleme, azınlık sınıfından aynı

sayıda gözleme ulaşana kadar çoğunluk sınıfından rastgele gözlemler seçecektir. Hedef sınıflarının yeni oranları bire eşit oluncaya kadar veri indirgeme işlemi yapılacağı gibi isteğe bağlı olarak farklı dengeleme oranları da kullanılabilir. Rastgele aşağı örnekleme yönteminin önemli dezavantajlarından biri, model performansına olumlu yönde katkı sağlayacak örneklerin de rastgele olarak veri kümesinden atılma riski olmasıdır. Ancak yine de hedef sınıflarının eşit şekilde algoritmaya girdi olarak sağlanması model performansını artırıcı yönde etki edebilmektedir.

2.6.2. Sentetik Azınlık Yukarı Örnekleme

Yukarı örnekleme, azınlık sınıfından veya birden fazla varsa azınlık sınıflarından örnek sayısını artırma işlemine verilen addır [129]. Yukarı örnekleme ile genel olarak azınlık sınıfı ile çoğunluk sınıfı örnek sayıları arasındaki oran bir oluncaya kadar azınlık sınıfı örnek sayısı artırılır. Yukarı örnekleme yaklaşımları da iki grup altında incelenebilir. İlk grupta yer alan yaklaşımlar azınlık sınıfından rastgele örnekler çıkararak hedef sınıflardaki örnek sayılarını eşitlemeyi hedefler. Bu durumda son veri kümesinde sahip olunan tüm gözlemler mevcut gözlemlerle aynı olur. Bu grupta yer alan yaklaşıma örnek olarak rastgele yukarı örnekleme (*random oversampling*) verilebilir. Alternatif olarak ikinci grupta yer alan yaklaşımlar, azınlık sınıfındaki örnek sayısını yeni örnekler üreterek arttırmaktadır. Bu grupta yer alan yaklaşımlar, azınlık sınıfında yer alan örnekleri kullanarak yeni örnekler üretmeyi hedefler. Bu durumda sentezlenen yeni örnekler mevcut örneklerden farklı olacağı için modelin öğrenme sürecine de katkı sağlayacaktır. Bu grupta önde gelen yaklaşımlar sentetik azınlık yukarı örnekleme (*synthetic minority oversampling technique*, SMOTE) ve onun varyasyonlarıdır.

SMOTE enterpolasyon yoluyla azınlık sınıfı örneklerinden yeni gözlemler yaratır [43]. Enterpolasyon, bilinen veri noktaları aralığında yeni veri noktaları oluşturulan bir tahmin türüdür. SMOTE ile sentetik veri üretimi gerçekleştirildiği için, rastgele yukarı örneklemede gözlemlenen veri tekrarı probleminin önüne geçer. SMOTE sadece azınlık sınıfı örnekleri ile çalışır. Bu nedenle ilk adım olarak veri sentezlenmesinde kullanılacak azınlık sınıfı örneklerini filtreler. İkinci aşama olarak sadece azınlık sınıfı örneklerini kullanarak bir k-en yakın komşu algoritması modeli oluşturur. Daha sonra her bir veri noktası için o noktaya en yakın komşu noktaları tespit eder. Son aşama olarak mevcut nokta ve yakın komşu noktalar arasındaki uzaklık değerleri de göz önüne alınarak yeni örnekler sentezlenir.

2.6.3. Borderline-SMOTE

Borderline-SMOTE, ikili sınıflandırma veya çok sınıflı sınıflandırmada azınlık sınıfı örnekleri ile çoğunluk sınıfı örnekleri arasında bulunan sınır bölgesi örneklerini kullanarak yeni sentetik veriler oluşturan sentetik azınlık yukarı örnekleme yönteminin bir varyasyonudur [44]. Sadece sınırdaki örneklerden veri sentezlenmesinin arkasında yatan sebep sınırdaki örneklerin sınıflandırılmasının daha zor olduğu hipotezinden kaynaklanmaktadır. Bu nedenle, algoritmanın sınırdaki örneklerden daha fazla öğrenebilmesi adına bu bölgedeki örnek sayısının artırılması hedeflenir.

Borderline-SMOTE yöntemi, hem çoğunluk hem de azınlık gözlemlerini içeren veri kümesini kullanarak bir k-en yakın komşu algoritması modeli eğitir. Sonraki aşamada azınlık sınıftan gelen, tüm komşuları karşı sınıfa ait gözlemleri bulur ve yok sayar. Bu örnekler gürültü olarak varsayımda bulunduğu ve sınıflandırma problemi için modele fazladan bilgi sağlamadıkları düşüncesinden dolayı bu yöntemde yok sayılırlar. Borderline-SMOTE aynı zamanda veri kümesindeki bir örneği komşularının çoğu azınlık grubuna aitse, bu örnekleri de yok saymayı temel alır. Bur örneklerin sınıflandırması kolay olduğu için veri sentezleme sürecinde bu örneklerden de faydalanılmaz. Bu yöntemde, komşularının çoğu çoğunluk sınıfına ait azınlık sınıfı örnekleri seçilerek veri sentezleme işlemi gerçekleştirilir.

Borderline-SMOTE yönteminin iki varyasyonu bulunmaktadır [44]. Birinci versiyonda sadece azınlık sınıfı örnekleri kullanılarak oluşturulan k-en yakın komşu modelinden tahmin yolu ile yeni örnekler üretilir. İkinci versiyonda ise hem sınır bölgesindeki azınlık sınıfı örnekleri hem de çoğunluk sınıfı örnekleri kullanılarak yeni örnekler sentezlenir. İkinci yöntemde örnek nokta ile komşuları arasındaki uzaklıklar birinci yöntemde göre daha kısa seçilir.

3. MALZEME VE YÖNTEM

Bu tez çalışmasının gerçekleştirilmesindeki temel amaç makine öğrenmesi modellerinde performans kayıplarına sebep olan dengesiz veri kümeleri problemine farklı bir çözüm getirmektir. Bu kapsamda önceki bölümlerde problem tanımlaması ve bu konu ile ilgili araştırmalara yer verilmiş olup bu bölümde önerilen çözümde faydanılacak yöntemler ve değerlendirme ölçütlerine yer verilmiştir. Bu bölümde çalışmada kullanılan veri kümelerine, veri kümelerinin analizlerine ve tezin ana odağı olan dengesiz veri kümeleri problemi için getirilen çözüm önerisine yer verilmiştir. Veri kümelerindeki niteliklerin tanıtılması ve model kurulumu öncesi yapılan işlemler bu bölümde paylaşılmıştır.

3.1. VERİ KÜMELERİ

Önerilen yöntemin performansının ölçülmesi için üç farklı veri kümesi üzerinde modeller kurularak farklı performans değerlendirme ölçütleri ile sonuçlar paylaşılmıştır. Elde edilen sonuçlar bulgular bölümünde paylaşılmıştır. Tartışma ve sonuç bölümünde ise önerilen yeni yaklaşımın problem çözümüne katkısı değerlendirilmiştir.

Dengesiz veri kümelerinde performansı etkileyen veri kaynaklı sorunlardan biri de dengesizlik derecesidir. Önerilen yöntemin değerlendirilmesi için farklı dengesizlik derecelerine sahip veri kümeleri seçilmiştir. Veri kümelerinin seçiminde diğer önemli etken veri kümelerinin büyüklüğüdür. Bu kısıtlar altında açık erişime sahip üç farklı veri kümesi seçilmiştir. Bir hastanın diyabete sahip olup olmadığını tespit etmek için toplanan **Pima Indians Diabetes** veri kümesi [130], çeşitli izinsiz ağ girişlerini tespit için oluşturulacak makine öğrenmesi modellerine kaynak oluşturan ve aynı zamanda bir yarışma veri kümesi olan **KDD Cup 1999** veri kümesi [131] ve kredi kartlarındaki dolandırıcılık girişimlerinin tespiti için kullanılan **Credit Card Fraud Detection** [132] veri kümeleri tercih edilmiştir. Veri ön işlemeden önce veri kümelerinin özellikleri Tablo 3.1’de yer almaktadır. KDD Cup 1999 veri kümesi için seçilen 2 hedef sınıfa ait veri kümesinin özelliklerine yer verilmiştir.

Tablo 3.1: Veri ön işleme öncesi veri kümelerinin özellikleri.

Veri Kümesi Adı	Kayıt Sayısı	Azınlık Sınıfı Büyüklüğü	Çoğunluk Sınıfı Büyüklüğü	Nitelik Sayısı	Dengesizlik Oranı
Pima Indians Diabetes Veri Kümesi	768	268	500	8	1,87
KDD Cup network intrusion detector Veri Kümesi	88831	999	87832	41	88
Credit Card Fraud Detection Veri Kümesi	284807	492	284315	29	577,9

3.1.1. Pima Indians Diyabet Veri Kümesi

Bu veri kümesi *National Institute of Diabetes and Digestive and Kidney Diseases* tarafından oluşturulmuştur [130]. Veri kümesinin amacı, veri kümesinde yer alan belirli biyolojik test ölçümlerine dayalı olarak bir kişinin diyabet hastası olup olmadığını tahmin etmektir. Veri kümesi bir ikili sınıflandırma problemine çözüm geliştirmek için toplanmıştır. Bu veri kümesi daha büyük bir veri kümesinden çeşitli kısıtlar uygulanarak elde edilmiştir. Veri kümesindeki tüm katılımcılar kadın seçilmiş olup en az 21 yaşındadırlar. Tablo 3.2’de veri kümesinde girdi ve çıktı olarak kullanılacak değişkenlerin açıklamalarına yer verilmiştir.

Tablo 3.2: Pima Indians diyabet veri kümesi değişkenleri.

Değişken Adı	Değişken Açıklaması
Pregnancies	Hamile kalma sayısı
Glucose	Oral glukoz tolerans testinde 2 saatlik plazma glukoz konsantrasyonu
BloodPressure	Diastolik kan basıncı değeri (mm Hg)
SkinThickness	Triceps deri kıvrım kalınlığı (mm)
Insulin	2 saatlik serum insülini (mu U/ml)
BMI	Vücut kitle indeksi
DiabetesPedigreeFunction	Diyabet soyağacı
Age	Yaş
Outcome	Diyabet hastalığının olup olmadığı

Pima Indians diyabet veri kümesinde yer alan değişkenlerin sekiz tanesi girdi değişkeni olarak tasarlanmışken *Outcome* değeri hedef değişken olarak belirtilmiştir. *Outcome* değişkeninin 1 olduğu durumlar hastalığın varlığını gösterirken 0 olduğu durumlar hastalığın olmadığını göstermektedir. Şekil 3.2’de biyolojik verilere ait değişkenlerin hedef değişken sınıfları ile aralarındaki ilişki ve değişkenlerin dağılımları gösterilmiştir. Şekil 3.2’de biyolojik veriler arasında güçlü bir korelasyon gözlemlenmediği için tüm değişkenler model kurulumunda girdi olarak kullanılmıştır. Aynı zamanda biyolojik verilerin hedef sınıflar için farklı değerler aldıkları görülmektedir.



Şekil 3.1: Pima Indians diyabet veri kümesi değişken dağılımları.

3.1.2. KDD Cup 1999 Veri Kümesi

Bu veri kümesi, bir askeri ağ ortamında simüle edilen çok çeşitli izinsiz giriş kayıtlarından oluşmuştur [131]. Veri kümesi 5. Uluslararası Bilgi Keşfi ve Veri Madenciliği Konferansı KDD-99 kapsamında düzenlenen 3. Uluslararası Bilgi Keşfi ve Veri Madenciliği Araçları Yarışmasında (*Third International Knowledge Discovery and Data Mining Tools Competition*) izinsiz giriş dedektörü öğrenme görevi için oluşturulmuştur. Saldırı tespit sistemleri, izinsiz girişler veya saldırılar olarak adlandırılan kötü bağlantılar ile normal bağlantılar arasında ayrım yapabilen bir tahmin modeli oluşturma problemidir. Tablo 3.3'te veri kümesinde yer alan değişkenlere ve açıklamalarına yer verilmiştir.

Tablo 3.3: KDD Cup 1999 veri kümesi değişkenleri [133].

Değişken Adı	Değişken Açıklaması
duration	Saniye cinsinden bağlantı süresi
protocol_type	Protokol türü : TCP, UDP, ICMP
service	Hedef ana bilgisayarın ağ hizmeti türü
flag	Normal veya yanlış bağlantı durumu
src_bytes	Kaynak ana bilgisayardan hedef ana bilgisayara veri baytlarının sayısı
dst_bytes	Hedef ana bilgisayardan kaynak ana bilgisayara veri baytlarının sayısı
land	Aynı ana bilgisayardan / bağlantı noktasına teslim ediliyorsa 1, değilse 0
wrong_fragment	Yanlış parça sayısı
urgent	Acil paket sayısı
hot	Sisteme duyarlı dosyalara ve dizinlere erişim
num_failed_logins	Başarısız giriş denemesi sayısı
logged_in	Giriş başarılı mı ? Yoksa başarısız mı ?
num_compromised	Güvenliği ihlal edilmiş koşulun gerçekleşme sayısı
root_shell	Root shell mi ?
su_attempted	Su root komutu mu?
num_root	Root kullanıcının erişme sayısı
num_file_creations	Dosya oluşturma işlemlerinin sayısı
num_shells	Shell komutunun kullanılma sayısı
num_access_files	Kontrol dosyasına erişim sayısı
num_outbound_cmds	Bir FTP oturumundaki giden bağlantıların sayısı
is_host_login	Giriş host listesine mi ait?
is_guest_login	Misafir girişi mi?
count	Son iki saniyede, mevcut bağlantıyla aynı hedef ana bilgisayara sahip bağlantı sayısı
srv_count	Son iki saniyede, mevcut bağlantıyla aynı hizmete sahip bağlantı sayısı
serror_rate	SYN hatası olan bağlantıların yüzdesi
srv_serror_rate	SYN hatası olan bağlantıların yüzdesi
rerror_rate	REJ hatası olan bağlantıların yüzdesi
srv_rerror_rate	REJ hatası olan bağlantıların yüzdesi
same_srv_rate	Aynı hizmete olan bağlantıların yüzdesi
diff_srv_rate	Farklı hizmetlere bağlantı yüzdesi
srv_diff_host_rate	Farklı ana bilgisayarlara bağlantı yüzdesi
dst_host_count	İlk 100 bağlantıda, mevcut bağlantıyla aynı hedef ana bilgisayara sahip bağlantı sayısı
dst_host_srv_count	İlk 100 bağlantıda, aynı hedef ana bilgisayara ve mevcut bağlantıyla aynı hizmete sahip bağlantı sayısı
dst_host_same_srv_rate	İlk 100 bağlantıda, aynı hedef ana bilgisayara ve mevcut bağlantıyla aynı hizmete sahip bağlantıların yüzdesi
dst_host_diff_srv_rate	İlk 100 bağlantıda, mevcut bağlantıyla aynı hedef ana bilgisayara ve farklı hizmetlere sahip bağlantıların yüzdesi
dst_host_same_src_port_rate	İlk 100 bağlantıda, aynı hedef ana bilgisayara ve mevcut bağlantıyla aynı kaynak bağlantı noktasına sahip bağlantıların yüzdesi
dst_host_srv_diff_host_rate	İlk 100 bağlantıda, mevcut bağlantıyla aynı hedef ana bilgisayara ve aynı hizmete sahip bağlantıların yüzdesi
dst_host_serror_rate	SYN hataları olan bağlantıların yüzdesi
dst_host_srv_serror_rate	İlk 100 bağlantıda, mevcut bağlantıyla aynı hedef ana bilgisayara ve aynı hizmete sahip bağlantıların yüzdesi, SYN hataları olan bağlantıların yüzdesi
dst_host_rerror_rate	İlk 100 bağlantı arasında, mevcut bağlantıyla aynı hedef ana bilgisayara sahip bağlantıların yüzdesi, RED hatası içeren bağlantıların yüzdesi
dst_host_srv_rerror_rate	İlk 100 bağlantıda, aynı hedef ana bilgisayara ve mevcut bağlantıyla aynı hizmete sahip bağlantıların yüzdesi RED hataları içeriyor
target	Varsa saldırı türü. Saldırı yoksa normal

KDD Cup 1999 veri kümesinde 41 adet girdi değişkeni ve 1 adet hedef değişkeni bulunmaktadır [131, 133]. Veri kümesi çok sınıflı sınıflandırma problemi için geliştirilmiş olup hedef değişken 5 farklı değerden birini alabilmektedir.

3.1.3. Credit Card Fraud Detection Veri Kümesi

Kredi kartlarında dolandırıcılık tespiti bankalar için önemli problemlerden biridir. Veri kümesi kredi kartlarında dolandırıcılık gerçekleşip gerçekleşmediğini tespit etmek için kurulacak modellerde kullanılmak üzere oluşturulmuştur. Veri kümesi, Avrupalı kart sahipleri tarafından Eylül 2013'te kredi kartlarıyla yapılan işlemlere ait bilgileri içermektedir [132]. Veri kümesinde kredi kartı sahiplerinin iki gün içerisinde yaptıkları işlemlerin kaydı tutulmaktadır. Veri kümesinde sadece bir temel bileşenler analizi (*principal component analysis*, PCA) dönüşümünün sonucu olan sayısal girdi değişkenleri yer almaktadır. Temel bileşenler analizi daha büyük veri kümelerindeki bilgi içeriklerinin özetlenmesini sağlayan istatistiksel bir yöntemdir [134]. Kişisel verilerin korunması nedeniyle veri kümesindeki değişkenlerin orijinal adları ve veri kümesi hakkında fazla detay bilgi paylaşılmamıştır.

3.2. VERİ ÖN İŞLEME

Bu bölümde çalışma kapsamında kullanılacak ve önceki bölümde genel bilgileri verilen veri kümeleri için modelleme öncesi gerçekleştirilen ön işleme ve analiz çalışmalarına yer verilmiştir. Her bir veri kümesi için hangi girdi değişkenlerinin model tasarımında kullanılacağı da bu bölümde belirtilmiştir.

3.2.1. Pima Indians Diyabet Ön İşleme

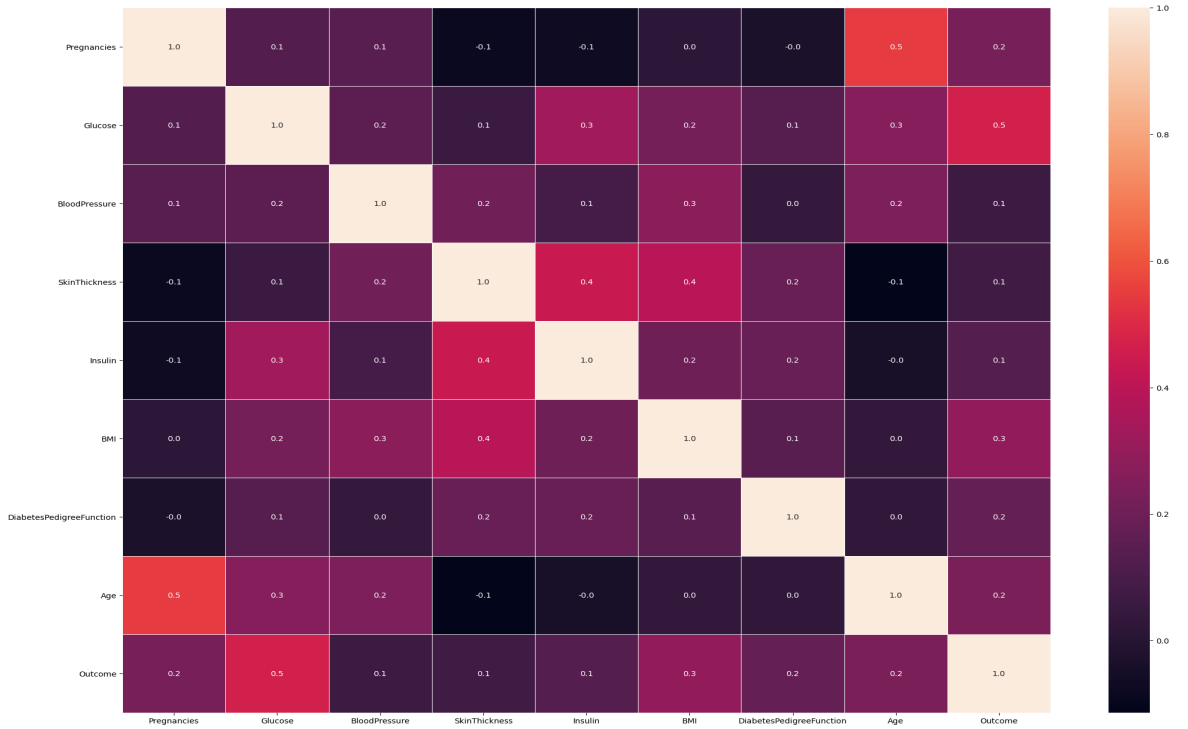
Pima Indians diyabet veri kümesinde girdi olarak kullanılacak 8 değişkenin tamamı nümeriktir. Veri kümesinde birçok değişken de eksik verinin bulunduğu tespit edilmiştir. Modelleme işleminden önce veride bulunan eksik veri probleminin çözülmesi amacıyla kullanılabilecek farklı yaklaşımlardan eksik verileri ortalama değer ile doldurma yaklaşımı tercih edilmiştir. Bu yaklaşım ile her değişken veya nitelik için hedef sınıfına göre gruplandırılıp ortalama değer hesabı yapılmıştır. İkinci aşamada var olan eksik veriler, bulunan ortalama değerler ile doldurulmuştur. Bu yöntemin seçilmesindeki ana sebep veri kümesinin küçük boyutlu olması ve daha fazla veri kaybının önüne geçilmek istenmesidir.

Eksik veri doldurma aşamasından sonra her değişken için 768 nümerik kayıt bulunmaktadır. Tablo 3.4'te veri doldurma işlemi öncesinde her bir girdi değişkeni için kaç adet eksik olmayan kayıt olduğu gösterilmiştir.

Tablo 3.4: Değişkenlerdeki eksik olmayan kayıt sayıları.

Değişken Adı	Kayıt Sayısı
Pregnancies	768
Glucose	763
BloodPressure	733
SkinThickness	541
Insulin	394
BMI	757
DiabetesPedigreeFunction	768
Age	768

Şekil 3.2'de girdi değişkenlerinin birbirleriyle karşılıklı ilişkilerini gösteren korelasyon matrisi yer almaktadır. Korelasyon matrisinde girdi değişkenleri arasında güçlü ilişkiler gözlenmemiştir. Bu nedenle tüm değişkenler modele girdi olarak verilmiştir. Elde edilen veri kümesinin dengesizlik oranı 1,87 olarak hesaplanmıştır.

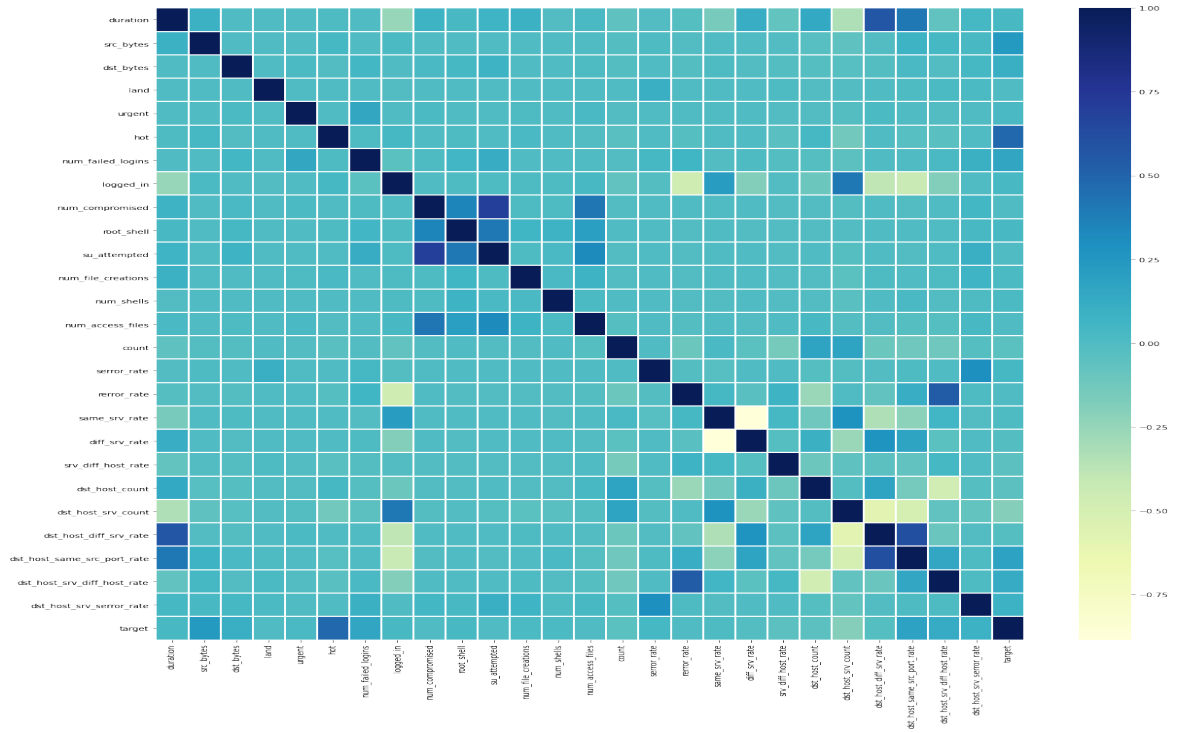


Şekil 3.2: Pima Indians diyabet veri kümesi korelasyon matrisi.

3.2.2. KDD Cup 1999 Ön İşleme

KDD Cup 1999 veri kümesi çok sınıflı sınıflandırma problemi için geliştirilmiş bir veri kümesidir. Çalışma kapsamında ikili sınıflandırma problemlerinde dengesiz veri kümelerine bir çözüm arandığı için hedef değişken olarak yeni oluşacak veri kümesindeki dengesizlik oranı da göz önüne alınarak yeni bir veri kümesi oluşturulmuştur. Yeni veri kümesi, hedef değişken "r2l" ve "normal" değerleri filtrelenerek oluşturulmuştur. Filtreleme sonucunda veri kümesinde 88831 kayıt bulunmaktadır. Aynı zamanda 41 adet girdi değişkeninin 3 tanesi metin ve 38 tanesi nümerik veri türüne sahiptir.

Veri kümesinde sabit değer aldığı gözlemlenen "wrong_fragment", "num_outbound_cmds" ve "is_host_login" değişkenleri kurulacak modelin iyileşmesinde katkı sağlamayacakları için veri kümesinden çıkartılmıştır. Korelasyon matrisi sonucunda birbirleriyle güçlü korelasyonu olan değişkenler de veriden temizlenmiştir. Yapılan temizleme işlemleri sonucunda geriye 29 girdi değişkeni kalmıştır. Geriye kalan nümerik değişkenlerin korelasyon matrisi Şekil 3.3'te gösterilmiştir.



Şekil 3.3: Veri temizleme sonrası KDD Cup 1999 veri kümesi korelasyon matrisi.

Makine öğrenmesinde kullanılan algoritmaların çoğu sayısal girdi değişkenleri ile çalıştığından dolayı veri kümesinde yer alan kategorik değişkenler *One-Hot-Encoding* yöntemiyle sayısal değerlere dönüştürülmüştür. Farklı değer aralıklarına sahip nümerik değişkenler standardizasyon işleminden geçirilerek tüm değişkenlerin aynı değer aralığında yer alması hedeflenmiştir. Standardizasyon işlemi, değişkenlerin ortalaması 0 varyansı 1 olan normal dağılım aralığına getirilmesi olarak adlandırılır [135]. Veri ön işleme çalışmaları sonucunda model kurulumunda kullanılacak veri kümesinin dengesizlik oranı 88'dir.

3.2.3. Credit Card Fraud Detection Ön İşleme

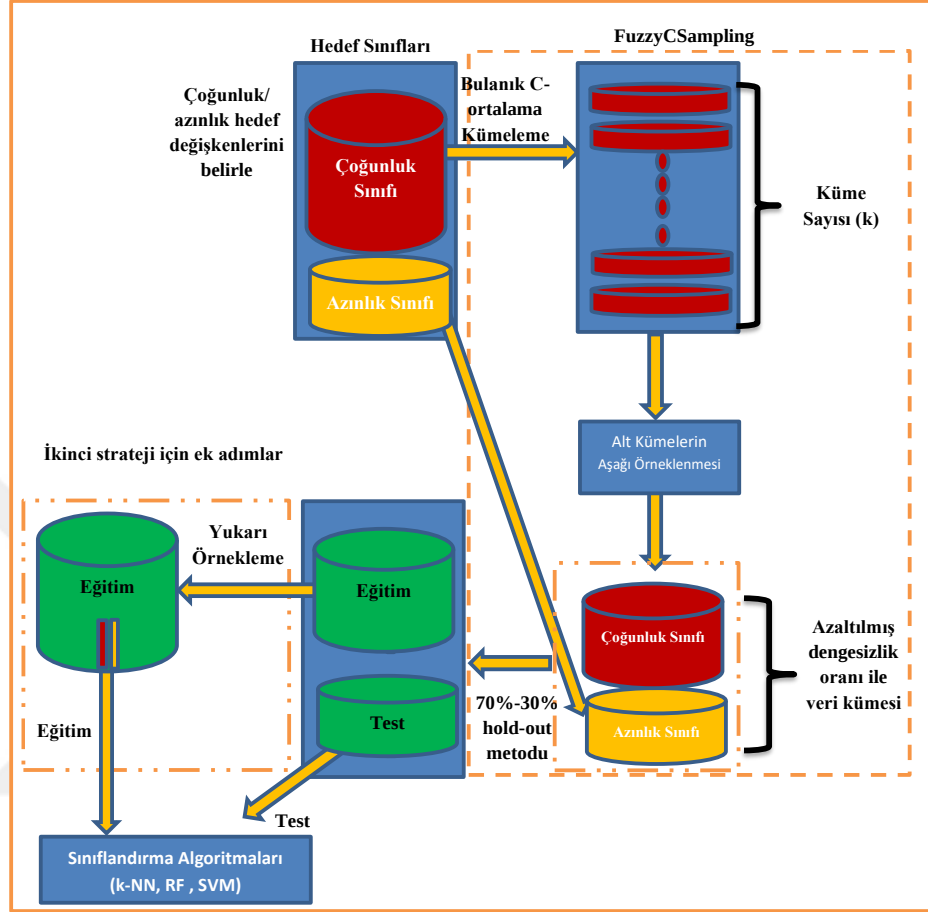
Kredi kartlarında dolandırıcılık tespiti veri kümesinde yer alan tüm değişkenler nümerik veri türündedir. 28 adet değişken temel bileşenler analizi sonucu elde edilmiştir. "*Time*" ve "*Amount*" değişkenleri ise farklı veri dağılımlarına sahiptir. "*Amount*" değişkenini diğer değişkenler ile yakın değer aralıklarına getirmek amacıyla standardizasyon işlemi uygulanmıştır. "*Time*" değişkeni ise kurulacak modele girdi olarak verilmeyeceğinden dolayı veri kümesinden çıkartılmıştır.

Veri kümesi ikili sınıflandırma problemine örnektir. Veri kümesinde yer alan "*class*" değişkeni 1 veya 0 değerini alabilmektedir. 1 değeri yapılan işlemde dolandırıcılık olduğunu gösterirken 0 değeri yapılan işlemde dolandırıcılık tespit edilmediğini göstermektedir. Veri kümesinde başlangıçta 284807 adet kayıt bulunmaktadır. Tekrar eden kayıtlar temizlendikten ve "*Time*" değişkeni atıldıktan sonra kurulacak modelde girdi olarak kullanılacak veri kümesi 275663 kayıt ve 29 değişkenden oluşmaktadır. Modelde kullanılacak veri kümesinin dengesizlik oranı 581,8'dir.

3.3. YÖNTEM

Bu bölümde dengesiz veri kümeleri problemi için önerilen yöntem anlatılmıştır. Dengesiz veri kümeleri kullanılarak oluşturulan modellerin performans problemlerinin nedenleri arasında sınır bölgesinde yer alan örneklerin sınıflandırmayı zorlaştırması ve dengesiz veri kümeleri için geliştirilen çözümlerin veri kaybına veya aşırı öğrenmeye yol açılması yer almaktadır. Tez kapsamında önerilen ve bundan sonra FuzzyCSampling olarak adlandırılacak yaklaşımlar bu iki sorunun giderilmesi hedeflenmiştir. Önerilen yöntemin blok şeması şekil 3.4'te gösterilmiştir. FuzzyCSampling dengesiz veri kümesindeki hedef

sınıfların sayısını eşitlemek için danışmanlı ve danışmansız öğrenme yöntemlerini birlikte kullanan hibrit bir yaklaşım olarak tasarlanmıştır.



Şekil 3.4: Geliştirilen FuzzyCSampling yöntemi.

FuzzyCSampling yaklaşımının ilk adımında çoğunluk ve azınlık sınıflarına karar verilir. Ardından, çoğunluk hedef sınıfı, bulanık c-ortalama kümeleme yöntemi ile k sayıda alt kümeye bölünür. Bu çalışma kapsamında oluşturulacak alt küme sayısına dirsek yöntemi (*elbow method*) ile karar verilmiştir. Bulanık c-ortalama kümelemesini kullanmanın birincil amacı, hedef sınıfların dengelenmesi sırasında bilgi kaybını azaltmaktır. Bulanık c-ortalama kümelemede bir veri noktası atandığı kümede bulunma olasılığı ile birlikte birden fazla kümede yer alabilmektedir. Belirlenecek bir olasılık eşik değeri ile en iyi kümeleme elde edilmektedir. Bulanık c-ortalama kümeleme yöntemi özellikle sınır bölgesindeki noktaların kümelere atanmasında daha etkin sonuç vermektedir. Yeni oluşturulan kümeler, azınlık hedef değişkeninden daha fazla örneğe sahiplerse aşağı örneklemeye ile örneklem büyüklüğü azınlık hedef değişken büyüklüğüne eşitlenir. Bu yaklaşımla veri kaybının en aza indirilmesi hedeflenmektedir. Sonraki aşamada aşağı örneklenmiş alt veri kümeleri ile azınlık hedef sınıfı

kümesi birleştirilerek yeni bir veri kümesi oluşturulur. Birleştirme işlemi gerçekleştirilirken azınlık sınıfı üzerinde herhangi bir işlem yapılmamaktadır.

Uygulanan yaklaşım sonrasında yeni oluşturulan veri kümesinin dengesizlik oranı azaltılmıştır. Ancak, veri kümesindeki hedef sınıflarının dağılımı hala dengesizdir. İkinci bir yaklaşım olarak, FuzzyCSampling'den elde edilen veri kümesindeki hedef değişkenlerin dengesiz dağılım problemini çözmek için veri kümesin son haline yukarı örnekleme uygulanarak sonuçlar incelenmiştir. Bulgular bölümde hem FuzzyCSampling hem de FuzzyCSampling ve yukarı örneklemenin birlikte kullanılmasıyla oluşturulan modellerin performansı karşılaştırılmıştır.



4. BULGULAR

Bu bölümde dengesiz veri küme problemine çözüm olarak sunulan FuzzyCSampling yöntemi ve ön işlemesi gerçekleştirilen 3 veri kümesi kullanılarak oluşturulan modellerin performansları ölçümlenerek önerilen yöntemin başarısı literatürde sık kullanılan çözümler ile karşılaştırılmıştır. Tablo 4.1’de ön işlemesi gerçekleştirilen veri kümelerinin büyüklükleri ve dengesizlik oranları paylaşılmıştır.

Tablo 4.1: Veri kümelerinin özellikleri.

Veri Kümesi Adı	Kayıt Sayısı	Azınlık Sınıfı Büyüklüğü	Çoğunluk Sınıfı Büyüklüğü	Nitelik Sayısı	Dengesizlik Oranı
Pima Indians Diabetes Veri Kümesi	768	268	500	8	1,87
KDD Cup network intrusion detector Veri Kümesi	88831	999	87832	29	88
Credit Card Fraud Detection Veri Kümesi	275663	473	275190	29	581,8

Veri kümeleri farklı büyüklük ve dengesizlik oranları göz önüne alınarak seçilip önerilen yaklaşımın farklı koşullar altında performansı incelenmiştir. Model değerlendirme ölçütü olarak doğruluk, eğri altında kalan alan, duyarlılık, kesinlik ve F-ölçütü hesaplanmıştır. Önerilen yöntemin başarısı her bir ölçüt için değerlendirilmiştir. Aynı zamanda model kurulumunda 3 farklı sınıflandırma algoritması kullanılarak yöntemin farklı algoritmaların performansını ne kadar iyileştirdiği yorumlanmıştır.

En küçük örnekleme sahip Pima Indians diyabet veri kümesi ile hedef sınıflarını eşitlemeden kurulan modellerin performansları Tablo 4.2’de verilmiştir. Tablo 4.2 üç farklı sınıflandırma algoritmasıyla kurulan modellerin performansını beş farklı performans değerlendirme ölçütü için sonuçlarını göstermektedir. Model kurulumunda kullanılan veri kümesinde 268 diyabet hastalığı taşıyan bireyin ve 528 diyabet hatalığı taşımayan bireyin bilgileri yer almaktadır.

Dengesiz veri kümesinin dengesizliği problemi göz önüne alınmadan kurulan modelde doğruluk ölçütü için en yüksek başarımın rastgele orman algoritması ile yakalandığı görülmektedir. K-en yakın komşu algoritması ile geliştirilen modelde doğruluk ölçütü değerlendirmesindeki başarısı ile duyarlılık ölçütü arasındaki farkın büyük olduğu gözlemlenmektedir. Destek vektör makineleri ile kurulan modelde ise tüm değerlendirme ölçütleri için model başarımı düşük çıkmıştır. Sonuçlara göre rastgele orman algoritması

bu veri kümesi için daha iyi sonuç vermektedir. Veri kümesindeki dengesizlik problemine rağmen azınlık sınıfı tahminlemede daha iyi sonuç vermiştir.

Tablo 4.2: Pima Indians dengesiz veri kümesi ile kurulan model performansı.

	K-NN	Rastgele Orman	SVM
Doğruluk	85,7	89,0	33,0
AUC	81,6	85,0	25,0
Kesinlik	82,5	87,0	2,0
Duyarlılık	70,0	77,0	2,0
F-ölçütü	75,9	82,0	2,0

Dengesiz veri kümelerinin çözümünde sık kullanılan çözümlerden olan rastgele aşağı örnekleme veri kümesine uygulanarak kurulan modellerin başarımları Tablo 4.3'te yer almaktadır. Hedef sınıf dağılımlarının eşitlendiği durumda rastgele orman algoritması ile kurulan modelde dengesiz veri kümesiyle kurulan modele göre tüm performans ölçütlerinde başarının azaldığı gözlemlenilmektedir. Destek vektör makineleri kullanılarak kurulan modelde iyileşme görülse de, genel model başarımının yeterli olmadığı gözlemlenmektedir. K-en yakın komşu algoritması ile kurulan modelde ise F-ölçütü değerlendirmesinde iyileşme olduğu gözlemlenmektedir. Aşağı örnekleme yöntemi ile azınlık sınıfı tahmini iyileşirken, çoğunluk sınıfının tahmininde performans azalması görülmektedir. Aşağı örnekleme ve üç farklı algoritma kullanılarak kurulan modeller arasında bu yöntemden en çok etkilenen algoritmanın k-en yakın komşu algoritması olduğu görülmektedir.

Tablo 4.3: Pima Indians veri kümesi rastgele aşağı örnekleme ile kurulan model performansı.

	K-NN	Rastgele Orman	SVM
Doğruluk	80,7	79,5	24,8
AUC	80,9	79,5	24,6
Kesinlik	90,4	81,8	29,0
Duyarlılık	69,5	76,8	32,9
F-ölçütü	78,6	79,2	30,8

Tablo 4.4'te ise bir yukarı örnekleme yaklaşımı olan SMOTE kullanılarak hedef sınıfların eşit sayıda örnekleme sahip olması sağlanmıştır. SMOTE yaklaşımı ve k-en yakın komşu algoritmasının kullanıldığı modelde kesinlik ölçütü dışında tüm değerlendirme ölçütleri için bir iyileşme gözlemlenmektedir. Azınlık sınıfının daha iyi tahminlendiği gözlemlenirken doğruluk ölçütünde de azalma olmadığı görülmektedir. Rastgele orman algoritması ile kurulan modelin başarısında ise somut bir artış gözlemlenmemektedir. Destek vektör

makineleri ile elde edilen modelde ise hedef sınıfların dengelenmiş olmasına rağmen model performansında önemli bir iyileşme görülmemektedir. Tüm sonuçlar ele alındığında rastgele orman algoritması ile elde edilen modellerin daha iyi sonuçlar verdiği gözlemlenmektedir. Ancak yukarı örnekleme k-en yakın komşu algoritması ile kurulan modellerin başarı artışında daha çok etki ettiği gözlemlenmektedir.

Tablo 4.4: Pima Indians veri kümesi SMOTE yöntemi ile kurulan model performansı.

	K-NN	Rastgele Orman	SVM
Doğruluk	86,5	88,3	17,0
AUC	85,4	86,0	17,0
Kesinlik	77,2	83,0	9,0
Duyarlılık	82,4	79,7	17,5
F-ölçütü	79,7	81,3	12,0

Bir diğer yukarı örnekleme yöntemi olan ve SMOTE yaklaşımı temel alınarak geliştirilen Borderline-SMOTE kullanılarak oluşturulan modellerin performansları Tablo 4.5'te yer almaktadır. Borderline-SMOTE yaklaşımıyla oluşturulan rastgele orman algoritması temelli modelin SMOTE yöntemiyle oluşturulan modelden daha iyi sonuç verdiği gözlemlenmektedir. Duyarlılık ve F-ölçütü artarken doğruluğa da negatif yönde bir etki yapmadığı görülmektedir. Borderline-SMOTE yönteminin azınlık sınıfın tahminini iyileştirdiği görülmektedir. Ancak destek vektör makineleri ile geliştirilen model performansında beş değerlendirme ölçütü için de başarılı sonuçlar elde edilmemiştir. K-en yakın komşu algoritması ile geliştirilen modelde ise azınlık sınıfının tahminlemesinin iyileştiği görülmekteyken çoğunluk sınıfının tahmin performansının azaldığı gözlemlenmektedir. Üç algoritma ile kurulan modeller göz önüne alındığında rastgele orman algoritmasının daha iyi sonuç verdiği görülmektedir.

Tablo 4.5: Pima Indians veri kümesi Borderline-SMOTE model performansı.

	K-NN	Rastgele Orman	SVM
Doğruluk	80,0	89,1	17,7
AUC	81,0	87,5	16,9
Kesinlik	64,5	82,6	7,0
Duyarlılık	83,7	83,7	14,0
F-ölçütü	72,9	83,2	10,0

Tablo 4.6'da Pima Indians veri kümesine FuzzyCSampling yaklaşımı ile uygulanarak elde edilen model sonuçlarına yer verilmiştir. FuzzyCSampling ve k-en yakın

komşu algoritmalarının birlikte kullanıldığı modelde performansının en yüksek olduğu gözlemlenmektedir. FuzzyCSampling yaklaşımı hem azınlık sınıf tahminini hem de çoğunluk sınıf tahminini iyileştirmiştir. Rastgele orman algoritması kullanılarak kurulan modelde ise performansta az da olsa azalma gözlemlenmektedir. Bu veri kümesiyle ve destek vektör makinesi algoritmasıyla oluşturulan modeller FuzzyCSampling yaklaşımıyla iyi sonuçlar vermemektedir. Dengesiz veri kümesiyle çalışırken rastgele orman algoritması daha iyi sonuç verirken önerilen FuzzyCSampling yaklaşımıyla k-en yakın komşu algoritmasıyla kurulan model öne çıkmıştır.

Tablo 4.6: Pima Indians veri kümesi FuzzyCSampling yöntemi model performansı.

	K-NN	Rastgele Orman	SVM
Doğruluk	88,5	85,6	28,7
AUC	87,2	84,2	25,0
Kesinlik	88,0	84,0	8,0
Duyarlılık	81,4	77,7	8,0
F-ölçütü	84,6	80,7	8,0

Tablo 4.7: Pima veri kümesi FuzzyCSampling+Borderline-SMOTE performansı.

	K-NN	Rastgele Orman	SVM
Doğruluk	85,1	84,6	17,7
AUC	85,6	83,4	16,7
Kesinlik	77,1	81,8	9,0
Duyarlılık	87,6	77,7	12,0
F-ölçütü	82,0	79,7	10,0

Önerilen FuzzyCSampling yaklaşımının sonunda elde edilen veri kümesinin hedef sınıfları yine eşit dağılımlı olmamaktadır. Bu nedenle veri kümesi çıktısına tekrar Borderline-SMOTE uygulanarak hedef sınıfların eşitlenerek oluşturulan model sonuçları Tablo 4.7’de paylaşılmıştır. Tablo 4.7’ye göre yaklaşım sonucu elde edilen veri kümesine yukarı örnekleme ve k-en yakın komşu algoritmasının kullanılması azınlık sınıfının tahmin performansını daha da arttırmıştır. Ancak aynı zamanda modelin çoğunluk sınıfı tahmin hatasını da arttırmıştır. Rastgele orman algoritması ile kurulan modelin performansı da yukarı örnekleme kullanılmadan oluşturulan modelin performansından daha düşüktür. Destek vektör makineleri ile geliştirilen model bu yaklaşımla da kötü sonuçlar göstermiştir. Pima Indians veri kümesinde FuzzyCSampling yaklaşımı kullanılarak oluşturulan modellerde, FuzzyCSampling yaklaşımından sonra ikinci veri çoğaltma

yaklaşımına ihtiyaç duyulmadığı görülmektedir.

Önerilen yaklaşımda bulanık c-ortalama kümeleme tekniği yerine k-ortalamlar kümeleme tekniği kullanıldığı durumda elde edilen modelin performans sonuçları Tablo 4.8’de verilmiştir. K-en yakın komşu algoritması ve k-ortalamlar kümeleme yöntemi kullanılarak oluşturulan model performansının bulanık c-ortalama kümeleme yöntemi ile oluşturulan model performansından daha düşük olduğu gözlemlenmiştir. Destek vektör makineleri kullanılarak oluşturulan tahmin modelinin performansı ise yine anlamlı bir artış göstermemiştir. Rastgele orman algoritması ile elde edilen sonuçlar ise k-ortalamlar kümeleme yöntemi kullanılarak elde edilen model performansının bulanık c-ortalama kümeleme yöntemi kullanılarak oluşturulan modelin performansından daha iyi sonuç verdiği görülmüştür.

Tablo 4.8: Pima veri kümesi k-ortalamlar kümeleme-FuzzyCSampling model performansı.

	K-NN	Rastgele Orman	SVM
Doğruluk	87,6	90,4	30,9
AUC	85,3	89,4	25,8
Kesinlik	91,0	89,6	4,0
Duyarlılık	75,3	85,1	3,0
F-ölçütü	82,4	87,3	4,0

Tablo 4.9’da k-ortalamlar kümeleme yöntemi kullanılarak uygulanan FuzzyCSampling yaklaşımı sonrasında ikinci bir yukarı örnekleme adımı uygulandığında elde edilen sonuçlara yer verilmiştir. Elde edilen sonuçlara göre ikinci yukarı örneklemenin uygulanması k-en yakın komşu algoritması ile kurulan modelde model performansını doğruluk, AUC, kesinlik ve F-ölçütü bakımından düşürmüştür. K-ortalamlar kümeleme yöntemi temelli FuzzyCSampling yaklaşımının azınlık sınıfın tahmin performansını arttırırken çoğunluk sınıfının tahmin performansını ve dolayısıyla genel model performansını düşürdüğü gözlemlenmiştir. Bu yaklaşımlar elde edilen sonuçlar yine de dengesiz veri kümesiyle oluşturulan modelden daha iyi performans göstermektedir. Rastgele orman algoritmasıyla oluşturulan modelin performansında da bir azalış görölse oransal olarak anlamlı bulunmamıştır. Destek vektör makineleriyle oluşturulan model ise yine beş performans değerlendirme kriteri için de kötü sonuçlar vermiştir.

KDD Cup 1999 veri kümesi ile herhangi bir dengesiz veri kümesi çözümü kullanmadan kurulan modelin performansı Tablo 4.10’da gösterilmiştir. KDD Cup 1999 veri kümesinin

Tablo 4.9: Pima veri kümesi k-ortalamlar kümeleme-FuzzyCSampling+ Borderline-SMOTE.

	K-NN	Rastgele Orman	SVM
Doğruluk	80,4	86,6	18,0
AUC	82,0	85,2	16,7
Kesinlik	69,9	85,3	8,0
Duyarlılık	88,4	79,0	11,1
F-ölçütü	77,8	82,0	9,0

dengeşizlik oranı 88 ve kayıt sayısı 88831'dir. Veri kümesinin hedef sınıfı 87832 normal ağ erişiminden ve 999 r2l atağından oluşmaktadır.

K-en yakın komşu algoritması ve dengeşiz veri kümesi kullanılarak oluşturulan modelin doğruluk değeri 99.8 elde edilmiştir. Bu değeri iyi bir model oluşturulduğunu düşündürse de duyarlılık ölçütünün 92.6 ve F-ölçütünün 94.7 elde edilmesi gerçekte modelin azınlık sınıfını iyi tahminleyemediğini işaret etmektedir. Rastgele orman algoritmasıyla geliştirilen modelin performansı k-en yakın komşu algoritması ve destek vektör makineleriyle geliştirilen modellerin performansında daha iyi sonuç vermiştir. Model performans değerlendirme metriklerine bakıldığında rastgele orman algoritması ile kurulan modelin azınlık sınıfının tahmin başarısını arttırdığı görülmüştür.

Destek vektör makineleri ile kurulan modelin doğruluk ölçütü değeri 98.4'tür. Bu değeri modelin başarılı olduğunu düşündürse de diğer değerlendirme metrikleri göz önüne alındığında modelin çoğunluk sınıfını iyi tahminlediğinden dolayı yüksek olduğu görülmektedir. Bu örnekte görüldüğü üzere büyük boyutlu dengeşiz veri kümelerinde doğruluk ölçütü değerlendirme kriteri olarak yanıltıcı sonuçlara yol açabilmektedir. Genel bir değerlendirme ile destek vektör makinelerinin bu veri kümesi için iyi sonuçlar üretmediği görülmüştür.

KDD Cup 1999 veri kümesi ve rastgele aşağı örnekleme yöntemi kullanılarak oluşturulan modelin beş performans değerlendirme ölçütüne göre sonuçları Tablo 4.11'de gösterilmiştir. K-en yakın komşu algoritması kullanılarak geliştirilen modelde doğruluk değeri bir azalma görülmektedir. Bununla birlikte duyarlılık değeri küçük bir iyileşme görülmektedir. Aşağı örnekleme ile kurulan modelin çoğunluk sınıfının tahminlemesini kötüleştirdiği görülmüştür.

Tablo 4.10: KDD Cup 1999 dengesiz veri kümesi ile kurulan model performansı.

	K-NN	Rastgele Orman	SVM
Doğruluk	99,8	99,0	98,4
AUC	96,3	98,1	64,1
Kesinlik	96,8	98,2	29,4
Duyarlılık	92,6	96,3	29,0
F-ölçütü	94,7	97,3	29,2

Rastgele orman algoritmasıyla geliştirilen modelin aşağı örneklenmiş veri kümesinde doğruluk ölçütü değeri 98.7'ye düşmüştür. Diğer değerlendirme ölçütleri de göz önüne alındığında k-en yakın komşu algoritması ile kurulan modelde olduğu gibi azınlık sınıfı tahminlemesi iyileşirken çoğunluk sınıfı tahminlemesi kötüleşmiştir. Rastgele orman algoritması ile geliştirilen modelin azınlık sınıfını tahminleme başarısı k-en yakın komşu algoritması ve destek vektör makineleri ile geliştirilen modellerin başarısından daha iyi olduğu görülmüştür. Aşağı örnekleme kullanılması destek vektör makineleri ile kurulan modelin azınlık sınıfı tahminleme başarısını önemli bir oranda arttırmıştır. Ancak çoğunluk sınıfının tahmin başarısı yine düşük kalmıştır.

Tablo 4.11: KDD Cup 1999 veri kümesi rastgele aşağı örnekleme ile kurulan model performansı.

	K-NN	Rastgele Orman	SVM
Doğruluk	98,7	98,7	97,7
AUC	97,7	97,8	93,2
Kesinlik	47,6	47,3	32,2
Duyarlılık	96,6	97,0	88,6
F-ölçütü	63,8	63,6	47,2

Veri kümesinin SMOTE tekniği kullanılarak yukarı örneklenmesi sonucu üç algoritmayla kurulan modellerin performansı Tablo 4.12'de gösterilmiştir. Veri miktarının yukarı örneklenmesi ile elde edilen yeni veri kümesi ve k-en yakın komşu algoritması kullanılarak oluşturulan modelin F-ölçütü ve kesinlik değerlerine göre performansın azaldığı görülmektedir. Duyarlılık ölçütü 97,6 olurken AUC değerinde de az bir iyileşme görülmüştür. SMOTE ile yukarı örneklenen veri kümesinin azınlık sınıfı tahminlemesini iyileştirdiği görülmüştür. Ancak aynı zamanda çoğunluk sınıfının tahmin performansı azalmıştır.

SMOTE yöntemi ile veri sentezlenerek ve rastgele orman algoritması kullanılarak

oluşturulan modelin performansında anlamlı sayılacak bir iyileşme gözlemlenmemektedir. Ancak rastgele orman algoritması ile oluşturulan model performansının k-en yakın komşu ve destek vektör makineleri ile oluşturulan modellerin performansından daha iyi performans gösterdiği gözlemlenmiştir. Destek vektör makineleri ile oluşturulan modelin azınlık sınıfı tahminleme başarısını artırmakla birlikte çoğunluk sınıfı başarısı düşük görülmüştür. Tüm değerlendirme ölçütleri göz önüne alındığında destek vektör makineleri ile oluşturulan modelin problem çözümünde kullanılmasının uygun olmadığı görülmüştür.

Tablo 4.12: KDD Cup 1999 veri kümesi SMOTE yöntemi ile kurulan model performansı.

	K-NN	Rastgele Orman	SVM
Doğruluk	99,7	99,9	94,2
AUC	98,7	98,1	96,4
Kesinlik	83,2	97,6	16,2
Duyarlılık	97,6	96,3	98,6
F-ölçütü	89,8	96,9	27,8

Borderline-SMOTE yöntemi kullanılarak veri kümesindeki hedef sınıfların dengesizlik oranının azaltılmasıyla ve k-en yakın komşu algoritması kullanılarak oluşturulan modelin performansı Tablo. 4.13'te yer almaktadır. K- en yakın komşu algoritmasıyla oluşturulan ve SMOTE veya Borderline-SMOTE ile veri sentezlenmiş modellerin performansı arasında önemli bir fark gözlemlenmemiştir. İki yaklaşım için de dengesiz veri kümesiyle oluşturulan modele oranla F-ölçütü ve duyarlılık değerleri açısından modellerin performansı artmıştır. Ancak çoğunluk sınıfı tahmin başarısı azalmıştır.

Rastgele orman algoritmasıyla geliştirilen model sonuçları Borderline-SMOTE yönteminde az da olsa daha iyi sonuç vermiş olsa da genelleştirilmiş bir değerlendirme yapmak için yeterli görülmemiştir. Destek vektör makineleri ile oluşturulan model performansı Borderline-SMOTE yaklaşımıyla da başarılı sonuçlar göstermemiştir. Azınlık sınıfının tahminlemesinde anlamlı sayılabilecek bir yükseliş gözlemlenirken çoğunluk sınıfı tahminlemesinin performansında düşüş gözlemlenmiştir. Borderline-SMOTE kullanılarak oluşturulan modellerin performansı içerisinde k-en yakın komşu algoritması kullanan modelin performansının daha çok arttığı gözlemlenmiştir.

Tez kapsamında önerilen FuzzyCSampling yaklaşımı kullanılarak oluşturulan modellerin performans sonuçları Tablo 4.14'te paylaşılmıştır. Önerilen yaklaşım ve k-en yakın komşu algoritmasıyla kurulan modelin azınlık sınıfı tahminlemesini iyileştirirken çoğunluk sınıfı

Tablo 4.13: KDD Cup 1999 veri kümesi Borderline-SMOTE yöntemi performansı.

	K-NN	Rastgele Orman	SVM
Doğruluk	99,7	99,9	94,5
AUC	98,8	98,3	95,2
Kesinlik	84,0	98,9	16,7
Duyarlılık	98,0	96,6	96,0
F-ölçütü	90,4	97,8	28,4

tahminlemedeki performans azalışının da diğer yaklaşımlara göre daha az olduğu görülmüştür. FuzzyCSampling yaklaşımıyla en yüksek doğruluk, duyarlılık ve F-ölçütü değerlerine ulaşılmıştır. Kesinlik ve AUC ölçütlerine göre de modelin başarılı sayılacak sonuçlar ürettiği gözlemlenmiştir.

FuzzyCSampling yaklaşımı ve rastgele orman algoritmasının kullanılmasıyla oluşturulan modelde en yüksek F-ölçütü, kesinlik ve doğruluk değerleri gözlemlenmiştir. FuzzyCSampling ile F-ölçütü sonuçlarına göre iyi performans gösteren diğer bir yaklaşım Borderline-SMOTE tekniği olmuştur. Destek vektör makineleri ile oluşturulan modelin ise azınlık sınıfı tahminlemede yeterli başarıya sahip olmadığı gözlemlenmiştir. FuzzyCSampling ve destek vektör makinesi kullanılarak oluşturulan model dengesiz veri kümesi kullanılarak oluşturulan modelden daha iyi sonuçlar vermiştir. Ancak genel model başarısının yeterli olmadığı gözlemlenmektedir.

Tablo 4.14: KDD Cup 1999 veri kümesi FuzzyCSampling yöntemi performansı.

	K-NN	Rastgele Orman	SVM
Doğruluk	99,8	99,9	98,5
AUC	97,3	98,2	73,0
Kesinlik	94,9	99,3	36,0
Duyarlılık	94,6	96,3	47,0
F-ölçütü	94,8	97,8	41,0

Tablo 4.15'te FuzzyCSampling yaklaşımı sonucunda elde edilen veri kümesine, Borderline-SMOTE yaklaşımı uygulanarak oluşturulan yeni veri kümesi ile kurulan modellerin performans ölçütleri gösterilmiştir. K-en yakın komşu algoritması kullanılarak oluşturulan modelin performansının tüm değerlendirme ölçütlerine göre FuzzyCSampling yaklaşımı ile elde edilen sonuçlardan düşük olduğu gözlemlenmiştir. İkinci bir adım uygulamanın model performansını k-en yakın komşu algoritması için arttırmadığı

gözlemlenmiştir.

FuzzyCSampling ve ikinci bir yukarı örneklemenin rastgele orman algoritmasıyla birlikte kullanılarak oluşturulan modelin performansında anlamlı sayılabilecek pozitif bir değişim görülmemiştir. Destek vektör makineleri ile kullanılması ise en yüksek duyarlılık değerini vermiştir. Ancak bu durumda diğer değerlendirme ölçütlerinin başarısında bir azalma gözlemlenmiştir.

Tablo 4.15: KDD veri kümesi FuzzyCSampling+Borderline-SMOTE yöntemi performansı.

	K-NN	Rastgele Orman	SVM
Doğruluk	99,3	99,9	94,7
AUC	98,5	98,3	95,5
Kesinlik	65,4	96,3	17,2
Duyarlılık	97,6	96,6	96,3
F-ölçütü	78,3	96,5	29,2

FuzzyCSampling yaklaşımında k-ortalamlar kümeleme yöntemi kullanılarak kurulan modellerin performans sonuçlarına Tablo 4.16'da yer verilmiştir. K-en yakın komşu algoritması ile kurulan modelin duyarlılık ölçütüne göre başarısı dengesiz veri kümesiyle geliştirilen modele göre artış göstermiştir. Ancak diğer ölçütler için kurulan modelde bir iyileşme gözlemlenmemiştir. Genel performans değerlendirmesine bakıldığında FuzzyCSampling yaklaşımında bulanık c-ortalama kümeleme yönteminin kullanılması daha iyi model performans sonuçları ürettiği gözlemlenmiştir.

Rastgele Orman algoritması kullanılarak geliştirilen modelde, model başarısında anlamlı sayılabilecek bir değişiklik göstermemiştir. Rastgele orman algoritmasıyla kurulan modelde de bulanık c-ortalama kümeleme yöntemiyle daha başarılı sonuçlar elde edilmiştir. Destek vektör makineleri ile kurulan modeller arasında ise FuzzyCSampling yaklaşımında k-ortalamlar kümeleme yönteminin kullanılması en iyi model çıktılarına üretmiştir. Ancak elde edilen model başarımları k-en yakın komşu ve rastgele orman algoritmalarıyla kurulan modellerin başarısının gerisinde kalmıştır. Üç algoritma arasında en iyi modelin rastgele orman algoritmasıyla olduğu gözlemlenmiştir.

K-ortalamlar kümeleme yöntemi kullanılarak uygulanan FuzzyCSampling yaklaşımının sonuç veri kümesine Borderline-SMOTE uygulanarak oluşturulan veri kümesiyle kurulan modellerin performansı Tablo 4.17'de verilmiştir. K-en yakın komşu algoritması ile

Tablo 4.16: KDD veri kümesi k-ortalamlar kümeleme-FuzzyCSampling yöntemi performansı.

	K-NN	Rastgele Orman	SVM
Doğruluk	99,7	99,8	99,1
AUC	97,5	98,4	88,2
Kesinlik	85,1	92,6	60,4
Duyarlılık	95,3	97,0	77,0
F-ölçütü	89,9	94,7	67,7

geliştirilen modeller arasında aşağı önekleme uygulanan model dışında tüm modeller arasında en düşük kesinlik ve F-ölçütü değerleri elde edilmiştir. Bulanık c-ortalama kümeleme kullanıldığında oluşturulan modelin performans sonuçlarının daha başarılı olduğu gözlemlenmiştir.

Tablo 4.17: KDD veri kümesi k-ortalamlar kümeleme-FuzzyCSampling +BorderlineSMOTE.

	K-NN	Rastgele Orman	SVM
Doğruluk	99,0	99,8	94,6
AUC	98,3	97,8	94,7
Kesinlik	53,9	87,8	16,7
Duyarlılık	97,6	96,0	94,6
F-ölçütü	69,5	91,7	28,4

Aynı yaklaşım rastgele orman algoritmasıyla birlikte kullanılarak model oluşturulunca model performansının yine bulanık c-ortalama kümeleme yöntemi kullanan yaklaşımdan düşük olduğu görülmüştür. Rastgele Orman algoritması ile oluşturulan tüm modeller arasında en düşük duyarlılık ve F-ölçütü değeri bu yaklaşımla elde edilmiştir. Destek vektör makineleri ile oluşturulan modelde ise azınlık sınıfı tahminlemesinde iyileşme görülürken genel anlamda model performansının azaldığı gözlemlenmiştir. Başka bir deyişle çoğunluk sınıf tahminlemesinin başarının azaldığı gözlemlenmiştir. Bu yaklaşım oluşturulan modeller arasında en yüksek AUC ve duyarlılık değeri k-en yakın komşu algoritmasıyla elde edilmiştir. Ancak tüm model değerlendirme ölçütleri göz önüne alındığına rastgele orman algoritmasıyla daha iyi model performansı elde edilmiştir.

Credit Card Fraud önerilen yöntemin başarısının testi için kullanılan en fazla kayıt sayısına ve en büyük dengesizlik oranına sahip veri kümesidir. Tablo 4.18’de veri kümesine herhangi bir işlem uygulanmadan kurulan modellerin performans değerlendirmelerine yer verilmiştir.

K-en yakın komşu algoritmasıyla kurulan modelde doğruluk ölçütü yüzde 99,0 gibi bir başarılı sonuç göstermesine rağmen modelin azınlık sınıfı tahminleme başarısının düşük kaldığı görülmüştür. Bu veri kümesi için duyarlılık 71,5 hesaplanırken F-ölçütü de 78,7 hesaplanmıştır. Modelin çoğunluk sınıfı örneklerini daha iyi öğrendiği görülmüştür.

Rastgele Orman algoritması ile geliştirilen modelde ise k-en yakın komşu algoritmasıyla geliştirilen modele göre performans değerlendirme ölçütlerinde bir bir iyileşme gözükse de duyarlılık ve F-ölçütü kriterleri cinsinden önemli bir fark oluşmadığı görülmüştür. Bu veri kümesi için de en düşük performans destek vektör makineleri ile kurulan modelden elde edilmiştir.

Tablo 4.18: Credit Card Fraud dengesiz veri kümesi model performansı.

	K-NN	Rastgele Orman	SVM
Doğruluk	99,0	99,0	99,0
AUC	85,7	86,8	66,3
Kesinlik	87,5	94,3	37,1
Duyarlılık	71,5	73,7	32,8
F-ölçütü	78,7	82,7	34,0

Credit Card Fraud veri kümesine aşağı örnekleme yaklaşımı uygulanarak kurulan modelin performans sonuçları Tablo 4.19’da verilmiştir. K-en yakın komşu algoritması kullanılarak geliştirilen modelde doğruluk ölçütü dışında tüm ölçütler cinsinden dengesiz veri kümesini kullanan modellere göre iyileşme görülmüştür. Aşağı örnekleme kullanılarak kurulan modeller arasında en başarılı sonuçlar rastgele orman algoritması ile elde edilmiştir. Ancak kesinlik ölçütüne göre en başarılı model k-en yakın komşu algoritması ile gerçekleşmiştir.

Destek vektör makinelerinin aşağı örnekleme ile birlikte kullanılması model sonuçlarını iyileştirmiştir. Özellikle azınlık sınıfı tahminlemesinde daha iyi sonuçlar elde edildiği gözlemlenmiştir. Oluşturulan tüm modeller arasında en yüksek F-ölçütü ve duyarlılık rastgele orman algoritması ile kurulan modelden elde edilmiştir. Aşağı örnekleme yaklaşımı kullanılan tüm modellerde ön işlemlerle veri kümesiyle oluşturulan modellere göre performanslarında iyileşme gözlemlenmiştir.

Credit Card Fraud veri kümesine SMOTE yukarı örnekleme tekniğinin uygulanması ile elde edilen sonuçlar ise Tablo 4.20’de verilmiştir. K-en yakın komşu algoritmasıyla kurulan modelde model performansının F-ölçütü ve kesinlik ölçütlerine göre önemli ölçüde azaldığı

Tablo 4.19: Credit Card Fraud veri kümesi rastgele aşağı örnekleme model performansı.

	K-NN	Rastgele Orman	SVM
Doğruluk	91,5	94,0	88,7
AUC	91,3	93,9	88,6
Kesinlik	96,7	96,1	91,4
Duyarlılık	85,5	91,3	84,7
F-ölçütü	90,7	93,6	87,9

görülmüştür. Modelin azınlık sınıfı tahmin başarısı artmış olsa da çoğunluk sınıfının tahmin başarısı azalmıştır.

Rastgele orman algoritmasıyla kurulan modelde ise dengesiz veri kümesiyle kurulan modele göre bir performans artışı görülmüştür. Ancak aşağı örnekleme yaklaşımı ile karşılaştırıldığında aşağı örnekleme yaklaşımının daha iyi sonuçlar ürettiği görülmüştür. Destek vektör makineleri ile kurulan modelde de SMOTE tekniği ile daha iyi duyarlılık değeri elde edildiği gözlemlenmiştir. Ancak dengesiz veri kümesi ile kurulan modele göre iyileşme söz konusu olsa da iyileşme aşağı örnekleme ile kurulan modele göre daha az olarak gerçekleşmiştir.

Tablo 4.20: Credit Card Fraud veri kümesi SMOTE yöntemi performansı.

	K-NN	Rastgele Orman	SVM
Doğruluk	99,9	99,9	98,8
AUC	89,0	88,3	89,9
Kesinlik	71,8	92,1	10,8
Duyarlılık	78,1	76,6	81,0
F-ölçütü	74,8	83,3	19,1

Credit Card Fraud veri kümesinin Borderline-SMOTE yukarı örnekleme yöntemi ile dengelenmesinden ortaya çıkan veri kümeleri ile oluşan modellerin performansları Tablo 4.21’de gösterilmiştir. K-en yakın komşu algoritması kullanılarak oluşturulan modelin başarısı SMOTE yöntemi ile elde edilen veri kümeleriyle kurulan modelin performansı ile paralellik göstermiştir. Kurulan modelin dengesiz veri kümesiyle kurulan modele göre AUC, doğruluk ve duyarlılık ölçütlerini geliştirdiği gözlemlenmiştir. Ancak kesinlik ve F-ölçütü kriterlerine göre model performansı azalmıştır.

Rastgele orman algoritması kullanılarak geliştirilen modelde ise SMOTE yaklaşımına göre az da olsa performansın kötüleştiği görülmüştür. Tüm değerlendirme ölçütleri için

dengelemez veri kümesiyle kurulan modele göre daha düşük değerler elde edilmiştir. Dengesiz veri kümesine Borderline-SMOTE yaklaşımının uygulanmasının model performansını düşürdüğü gözlemlenmiştir. Destek vektör makineleri ile oluşturulan modelde ise model başarımı F-ölçütü ve kesinlik için en düşük seviyeye ulaşmıştır. Tüm değerlendirme kriterleri göz önüne alındığında Borderline-SMOTE yaklaşımı için en iyi sonuç veren model rastgele orman algoritması ile elde edilmiştir.

Tablo 4.21: Credit Card Fraud veri kümesi Borderline-SMOTE yöntemi performansı.

	K-NN	Rastgele Orman	SVM
Doğruluk	99,9	99,9	94,0
AUC	88,6	85,7	84,9
Kesinlik	71,6	94,2	2,0
Duyarlılık	77,3	71,5	75,9
F-ölçütü	74,3	81,3	4,0

Credit Card Fraud veri kümesi ve FuzzyCSampling yaklaşımı kullanılarak oluşturulan modellerin performans sonuçları Tablo 4.22’de yer almaktadır. K-en yakın komşu algoritmasıyla elde edilen sonuçlara göre genel model performansı diğer yaklaşımlara oranla daha iyi sonuç vermiştir. Özellikle dengelemez veri kümesiyle oluşturulan model sonuçları ile karşılaştırıldığında önemli bir iyileşme olduğu görülmüştür.

Tablo 4.22: Credit Card Fraud veri kümesi FuzzyCSampling yöntemi performansı.

	K-NN	Rastgele Orman	SVM
Doğruluk	94,5	96,0	85,1
AUC	91,4	94,1	83,0
Kesinlik	98,3	97,6	74,6
Duyarlılık	83,5	89,2	77,8
F-ölçütü	90,3	93,2	76,2

Rastgele orman algoritması ile kurulan modeller incelendiğinde tüm değerlendirme kriterleri göz önüne alındığında en iyi model performansına sahip olduğu görülmüştür. Destek vektör makineleri ile kurulan modelde ise performans artışı gözlemlense bile aşağı örnekleme yaklaşımıyla kurulan modelin daha iyi sonuç verdiği görülmüştür.

Tablo 4.23’te ise Credit Card Fraud veri kümesinde FuzzyCSampling yönteminden sonra yukarı örnekleme yapıldığı zaman oluşacak yeni modellerin performans sonuçları verilmiştir. K-en yakın komşu algoritmasıyla birlikte kurulan modele göre ikinci örnekleme

adımının AUC ve duyarlılık ölçütlerine göre daha iyi sonuç verdiği gözlemlenmiştir. Ancak doğruluk, kesinlik ve F-ölçütü için daha düşük sonuç elde edilmiştir.

Rastgele orman algoritmasıyla oluşturulan modelin performans sonucunun ise k-en yakın komşu algoritmasıyla oluşturulan modelle paralel davrandığı gözlemlenmiştir. Bu modelde de AUC ve duyarlılık ölçütleri için daha iyi sonuçlar elde edildiği halde doğruluk, kesinlik ve F-ölçütü için daha kötü sonuçlar elde edilmiştir. Destek vektör makineleriyle kurulan model de ise model başarımı diğer iki algoritmayla kurulan modellere göre daha düşüktür. Ancak ikinci adımın model performansına etkisi diğer algoritmalarla oluşturulan modellerle paralellik göstermiştir. Tüm modeller göz önüne alındığında FuzzyCSampling yaklaşımından sonra ikinci adım olarak yukarı örnekleme uygulanması, hangi değerlendirme kriterine öncelik verildiğine göre değişmektedir.

Tablo 4.23: Credit Card Fraud veri kümesi FuzzyCSampling+Borderline-SMOTE performansı.

	K-NN	Rastgele Orman	SVM
Doğruluk	90,5	94,9	73,3
AUC	92,0	94,3	75,7
Kesinlik	78,3	90,9	54,2
Duyarlılık	95,7	92,8	82,1
F-ölçütü	86,1	91,8	65,3

Tablo 4.24'te FuzzyCSampling yaklaşımında kullanılan bulanık c-ortalama kümeleme yöntemi, k-ortalamlar kümeleme yöntemi ile değiştirilerek Credit Card Fraud veri kümesi ile oluşturulmuş modellerin performansları gösterilmiştir. K-en yakın komşu algoritmasıyla geliştirilen modelde önerilen yaklaşımda k-ortalamlar kümeleme yönteminin kullanılması, model başarısını dengesiz veri kümesi, SMOTE ve Borderline-SMOTE yöntemleriyle oluşturulan modellerin başarılarından daha iyi sonuç vermiştir. Ancak aşağı örnekleme ve bulanık c-ortalama kümeleme tabanlı FuzzyCSampling yöntemleriyle oluşturulan modellerden daha iyi sonuç vermemiştir.

Rastgele orman algoritması kullanılarak oluşturulan model sonuçları da k-en yakın komşu algoritması kullanılarak oluşturulan modellerin performansları ile paralellik göstermiştir. FuzzyCSampling yönteminin azınlık sınıfının tahminlemesini iyileştirdiği görülmektedir. Destek vektör makineleriyle oluşturulan modelin performansının da dengesiz veri kümesi ve yukarı örnekleme yöntemlerinin kullanılmasıyla oluşturulan modellerden daha iyi

sonuç verdiği gözlemlenmiştir. Ancak aşağı örnekleme ve FuzzyCSampling yöntemleriyle oluşturulan modellerin performanslarını geçememiştir.

Tablo 4.24: Fraud veri kümesi k-ortalamalar kümeleme-FuzzyCSampling performansı.

	K-NN	Rastgele Orman	SVM
Doğruluk	99,0	99,2	97,4
AUC	88,6	90,1	77,3
Kesinlik	87,4	94,2	56,7
Duyarlılık	77,6	80,4	55,9
F-ölçütü	82,2	86,7	56,3

K-ortalamalar kümeleme yaklaşımına dayalı FuzzyCSampling ve Borderline-SMOTE yönteminin birlikte kullanılmasıyla elde edilen yeni veri kümesiyle oluşturulan modellerin performans değerlendirmeleri Tablo 4.25'te verilmiştir.

Tablo 4.25: Fraud veri kümesi k-ortalamalar kümeleme-FuzzyCSampling+Borderline-SMOTE.

	K-NN	Rastgele Orman	SVM
Doğruluk	97,8	99,0	82,5
AUC	91,4	89,0	81,8
Kesinlik	59,0	89,6	12,4
Duyarlılık	84,6	78,3	81,1
F-ölçütü	69,5	83,5	21,6

K-en yakın komşu algoritması ile kurulan modelde veri kümesinin ön işleminde ikinci bir yukarı örnekleme yaklaşımının uygulanmasının model performansını azalttığı gözlemlenmiştir. Rastgele orman ve destek vektör makineleriyle kurulan modellerde de F-ölçütü, kesinlik ve doğruluk ölçütleri için model performanslarının azaldığı gözlemlenmiştir. Ancak destek vektör makineleriyle kurulan modelden ikinci bir aşama olarak yukarı örnekleme kullanılmasıyla azınlık sınıfının tahminlemesini iyileştirdiği gözlemlenmiştir.

Tablo 4.26'da tez kapsamında incelenen ve Pima Indians Diyabet veri kümesiyle model kurulmuş çalışmaların sonuçlarına yer verilmiştir. Tablo 4.27'de isetez kapsamında incelenen ve *Credit Card Fraud* veri kümesiyle model kurulmuş çalışmaların sonuçlarına yer verilmiştir.

Tablo 4.26: Pima veri kümesi ile ilgili çalışmalarla model performansının karşılaştırılması.

Yayın	Sınıflandırma Algoritması	Dengesiz Veri Kümesi Çözümü	Doğruluk	AUC	Kesinlik	Duyarlılık	F-ölçütü
Batista ve diğ. [136]	C4.5	Rastgele Yukarı Örnekleme	Hesaplanmamış	85,32	Hesaplanmamış	Hesaplanmamış	Hesaplanmamış
Batista ve diğ. [136]	C4.5	SMOTE	Hesaplanmamış	85,49	Hesaplanmamış	Hesaplanmamış	Hesaplanmamış
Batista ve diğ. [136]	C4.5	Smote+Tomek	Hesaplanmamış	85,56	Hesaplanmamış	Hesaplanmamış	Hesaplanmamış
Batista ve diğ. [136]	C4.5	Smote+ENN	Hesaplanmamış	83,66	Hesaplanmamış	Hesaplanmamış	Hesaplanmamış
Batista ve diğ. [136]	C4.5	x	Hesaplanmamış	82,33	Hesaplanmamış	Hesaplanmamış	Hesaplanmamış
Batista ve diğ. [136]	C4.5	Rastgele Aşağı Örnekleme	Hesaplanmamış	81,49	Hesaplanmamış	Hesaplanmamış	Hesaplanmamış
Batista ve diğ. [136]	C4.5	CNN	Hesaplanmamış	80,08	Hesaplanmamış	Hesaplanmamış	Hesaplanmamış
Batista ve diğ. [136]	C4.5	CNN+Tomek	Hesaplanmamış	81,71	Hesaplanmamış	Hesaplanmamış	Hesaplanmamış
Batista ve diğ. [136]	C4.5	Tomek	Hesaplanmamış	83,11	Hesaplanmamış	Hesaplanmamış	Hesaplanmamış
Batista ve diğ. [136]	C4.5	OSS	Hesaplanmamış	79,23	Hesaplanmamış	Hesaplanmamış	Hesaplanmamış
Batista ve diğ. [136]	C4.5	NCL	Hesaplanmamış	82,55	Hesaplanmamış	Hesaplanmamış	Hesaplanmamış
Kumari ve Chitra [7]	SVM	x	80,0	Hesaplanmamış	Hesaplanmamış	78,0	Hesaplanmamış
Tao ve diğ. [137]	SVM	x	Hesaplanmamış	80,0	Hesaplanmamış	Hesaplanmamış	57,5
Tao ve diğ. [137]	CSVM	x	Hesaplanmamış	80,8	Hesaplanmamış	Hesaplanmamış	64,7
Tao ve diğ. [137]	HesSVM	x	Hesaplanmamış	80,7	Hesaplanmamış	Hesaplanmamış	64,3
Tao ve diğ. [137]	RUSSVM	Rastgele Aşağı Örnekleme	Hesaplanmamış	81,5	Hesaplanmamış	Hesaplanmamış	65,8
Tao ve diğ. [137]	ROSSVM	Rastgele Yukarı Örnekleme	Hesaplanmamış	65,1	Hesaplanmamış	Hesaplanmamış	80,5
Tao ve diğ. [137]	SMOTESVM	SMOTE	Hesaplanmamış	64,1	Hesaplanmamış	Hesaplanmamış	80,7
Tao ve diğ. [137]	BSMOTESVM	Yukarı Örnekleme	Hesaplanmamış	61,8	Hesaplanmamış	Hesaplanmamış	80,8
Tao ve diğ. [137]	WKSMMOTESVM	Yukarı Örnekleme	Hesaplanmamış	65,0	Hesaplanmamış	Hesaplanmamış	81,5
Tao ve diğ. [137]	AdaSVM	x	Hesaplanmamış	64,4	Hesaplanmamış	Hesaplanmamış	81,0
Tao ve diğ. [137]	OWASVM	x	Hesaplanmamış	60,5	Hesaplanmamış	Hesaplanmamış	80,7
Tao ve diğ. [137]	EFSVM	x	Hesaplanmamış	64,5	Hesaplanmamış	Hesaplanmamış	81,1
Tao ve diğ. [137]	GPFSVM	x	Hesaplanmamış	64,6	Hesaplanmamış	Hesaplanmamış	81,3
Tao ve diğ. [137]	EFLSSVM-CIL	x	Hesaplanmamış	64,7	Hesaplanmamış	Hesaplanmamış	81,1
Tao ve diğ. [137]	EFLSTWSVM-CIL	x	Hesaplanmamış	64,6	Hesaplanmamış	Hesaplanmamış	81,3
Tao ve diğ. [137]	ACFSVM	x	Hesaplanmamış	65,7	Hesaplanmamış	Hesaplanmamış	83,1
Chang ve diğ. [138]	J48	x	74,78	78,55	70,86	88,43	78,68
Chang ve diğ. [138]	Rastgele Orman	x	79,57	86,24	89,40	81,33	85,17
Chang ve diğ. [138]	Naive Bayes	x	78,67	84,63	81,88	86,75	84,24

Tablo 4.27: Fraud veri kümesi ile ilgili çalışmalarla model performansının karşılaştırılması.

Yayın	Sınıflandırma Algoritması	Dengesiz Veri Kümesi Çözümü	Doğruluk	AUC	Kesinlik	Duyarlılık	F-ölçütü
Kumar ve diğ. [139]	Rastgele Orman	x	90,0	Hesaplanmamış	Hesaplanmamış	Hesaplanmamış	Hesaplanmamış
Sailusha ve diğ. [140]	Rastgele Orman	x	Hesaplanmamış	Hesaplanmamış	97,0	89,0	93,0
Mahesh ve diğ. [141]	Rastgele Orman	Aşağı Örnekleme	94,0	Hesaplanmamış	94,0	94,0	94,0
Mahesh ve diğ. [141]	Rastgele Orman	SMOTE	99,0	Hesaplanmamış	91,0	92,0	92,0
Mahesh ve diğ. [141]	Rastgele Orman	SMOTE+TOMEK	99,0	Hesaplanmamış	92,0	94,0	93,0

5. TARTIŞMA VE SONUÇ

Teknolojik gelişmeler ile birlikte üretilen veri miktarı günden güne artmıştır. Firmaların sahip oldukları veriden anlamlı bilgi çıkarma istekleri de artan kaynaklarıyla birlikte artış göstermiştir. Son yıllarda firmalar ellerindeki büyük boyutlu verileri makine öğrenmesi

modellerinin geliştirilmesinde kullanarak rakiplerinin önünde kalmayı hedeflemişlerdir. Ancak bu yarış içerisinde işlenmesi gereken veri miktarının artması makine öğrenmesi algoritmalarının da aşılması gereken zorluklarını ortaya çıkarmıştır. Makine öğrenmesi modellerinin geliştirilmesinde kullanılan kaynak dengesiz bir veri kümesi ise kurululan modelin performansı düşük olmaktadır. Dengesiz veri kümeleri, hedef değişkenleri eşit olmayan dağılımlara sahip olan veri kümeleri olarak ifade edilmiştir. Bu tez çalışması kapsamında öncelikle yayınlanan dengesiz veri kümeleri sorunuyla ilgili literatürün eksiksiz bir değerlendirmesi sunulmuş ve problem çözümünde kullanılacak yeni bir yaklaşım olan FuzzyCSampling önerilmiştir.

Tezin ilk kısmında Scopus veri tabanından dengesiz veri kümeleriyle ilgili çalışmaların verileri toplanarak bir bibliyometrik analiz gerçekleştirilmiştir. Scopus veri tabanına dayalı analiz, dengesiz veri kümeleri probleminin 2000’li yılların başında dikkat çekmeye başladığını göstermiştir. Chawla ve diğerleri [22] AAAI ve ICML konferanslarının ardından bir çalışma yayınlayarak sorunun önemini vurgulamışlardır. He ve Garcia da bu atölyeleri alandaki önemli çalışmalar arasına koymuştur [24]. Sorunla ilgili çalıştayların ardından SMOTE yönteminin yayımlanması, alandaki araştırmaları arttırmıştır.

SMOTE, uzun yıllara dayanan araştırmalara ve çok sayıda çalışmaya rağmen hala en yaygın uygulanan dengesiz veri kümesi problemi çözümüdür. SMOTE, birçok yazar tarafından makalelerinde [20, 29] bahsedilmiştir. SMOTE yaklaşımına olan ilgi birçok varyasyonunun da ortaya çıkmasını sağlamıştır. SMOTE yaklaşımının performansını iyileştirmek için Safe-Level-SMOTE [55], SMOTE-IPF [61], Borderline-SMOTE [44] ve çeşitli varyasyonları üretilmiştir. Anahtar kelime analizi, yukarı örnekleme yaklaşımlarının ve özellikle SMOTE’in yazarların büyük ilgisini çektiğini de desteklemiştir. Dengesiz veri kümeleri sorunu çoğunlukla Amerika Birleşik Devletleri, Çin ve Almanya’dan yazarlar tarafından araştırılmıştır.

Anahtar kelime analizi makine öğrenimi algoritmaları olan rastgele orman ve destek vektör makinelerinin çoğunlukla dengesiz veri kümeleriyle birlikte araştırıldığını göstermiştir. Rastgele orman ve destek vektör makineleri üzerine yapılan araştırmalar, dengesiz veri kümelerinin performansını iyileştirmek için örnekleme tekniklerini birleştirmeye odaklanmışlardır [48, 66]. Dengesiz veri kümeleriyle ilgili en çok atıfta bulunan ilk yazarlar Nitesh V. Chawla ve Francisco Herrera’dır. Chawla [43] özellikle SMOTE

yaklaşımının geliştirilmesine katkıda bulunurken, Herrera'nın incelemeleri [34] büyük ilgi görmüştür.

Derin öğrenmenin kullanım alanları da veri miktarının artmasıyla genişlemiştir. Son yıllarda araştırmacılar, dengesiz veri kümelerinin derin öğrenme ve büyük veri kümeleri üzerindeki etkilerini de çalışmaya başlamıştır. Bibliyometrik analiz sonuçlarına göre SMOTE ve derin öğrenmenin konuları önemini korumuşlardır. Büyük veri araştırmalarındaki dengesiz veri kümeleri, yalnızca iyi bir tahmin elde etmeye değil, aynı zamanda önerilen yaklaşımların hızına da odaklanmıştır.

Sun ve diğerleri [34] alandaki literatürün en kapsamlı incelemesini yayınlamışlardır. Alandaki araştırmaların, bibliyometrik analiz sonucundan da görülen ikili sınıflandırmaya odaklandığını vurgulamışlardır. Çok sınıflı sınıflandırma probleminde dengesiz veri kümeleri üzerine araştırmaların sayısı ikili sınıflandırma ile ilgili araştırmaların sayısına görece daha azdır. Hem Sun ve diğerleri [34] hem de bibliyometrik analiz sonucu, topluluk yaklaşımlarının model performansını iyileştirmesinin ilgi çektiğini göstermiştir.

Dengesiz veri kümelerinin model oluşturma üzerindeki olumsuz etkilerini azaltmak için birçok çözüm önerilmiştir. Bu tez kapsamında bibliyometrik analiz sonrasında bulanık c-ortalama kümeleme ve örnekleme stratejilerine dayanan, dengesiz veri kümelerinden öğrenmeye yönelik FuzzyCSampling adlı yeni bir yöntem önerilmiştir. Önerilen FuzzyCSampling yaklaşımının performansı çeşitli endüstrilerden ve çeşitli veri kümelerinde daha önce önerilen yaklaşımlarla karşılaştırılarak değerlendirilmiştir. Birbirinden farklı büyüklük ve dengesizlik oranlarına sahip veri kümelerini kullanarak oluşturulan modeller, FuzzyCSampling'in model performansını önemli ölçüde iyileştirdiğini göstermiştir.

FuzzyCSampling yaklaşımının performansının değerlendirilmesi için kullanılan Pima Indians diyabet ve *Credit Card Fraud* veri kümeleriyle literatürde yer alan çalışmalar incelenmiştir. Batista ve diğ. [136] farklı veri kümeleri ile dengesiz veri kümesi çözümlerinin performanslarını kıyasladıkları çalışmalarında karar ağaçları algoritması kullanarak modeller oluşturmuşlardır. Pima Indians diyabet veri kümesiyle yaptıkları çalışmada en yüksek AUC değerini rastgele yukarı örnekleme çözümüyle 86,03 olarak bulmuşlardır. Model değerlendirme ölçütü olarak sadece AUC ölçütünü göz önünde bulundurmışlardır. FuzzyCSampling ve k-en yakın komşu algoritması kullanılarak kurulan modelde AUC değeri 87,2 olarak gözlemlenmiştir. Tao ve diğ. [137] önerdikleri algoritma

seviyesinde çözüm olan ACFSVM yaklaşımı ile Pima Indians diyabet veri kümesinde F-ölçütünü 65,7 ve AUC değerini 83,1 olarak hesaplamışlardır. FuzzyCSampling yöntemi ve SVM kullanılmasıyla AUC değeri 25,2 elde edilmiştir. Model performansları arasındaki farkın nedeni veri ön işleme aşamalarındaki farklılık veya kullanılan algorithmadan kaynaklı olabileceği görülmüştür. López ve diğ. [34] yaptıkları çalışmada dengesiz veri kümeleri için önerilen çözümlerin performanslarını karşılaştırmışlardır. Kullandıkları 66 veri kümesi içinde Pima Indians diyabet veri kümesi de yer almaktadır. Ancak raporlamada 66 veri kümesi için AUC değerlerinin ortalaması alındığı için Pima Indians diyabet veri kümesi performansına erişilememektedir. Kumari ve Chitra [7] Pima Indians diyabet veri kümesi ile yaptıkları çalışmada destek vektör makinelerini kullanarak doğruluğu 78 ve duyarlılığı 80 olarak hesaplamışlardır. Veri ön işleme kısmında veri kümesindeki örnek sayısını azaltmışlardır. FuzzyCSampling yöntemi ve destek vektör makineleriyle kurulan modelin doğruluğu 28,7 ve duyarlılığı 8,0 olarak hesaplanmıştır. Destek vektör makineleri için model performansındaki fark, veri ön işleme kısmında da farklılıklar olduğu için önerilen yöntemle ilişkilendirilmemiştir. Mahesh ve diğ. [141] dolandırıcılık işlemlerinin tespit edilebilmesi için makine öğrenmesinin kullanılabileceğini ifade edip açık kaynak olan *Credit Card Fraud* veri kümesi ile model geliştirmişlerdir. En iyi kesinlik (94), duyarlılık (94) ve F-ölçütü (94) değerlerine aşağı örnekleme yöntemi ve rastgele orman algoritmasıyla ulaşmışlardır. FuzzyCSampling yöntemiyle kesinlik (97), duyarlılık (89,2) ve F-ölçütü (93,2) elde edilmiştir. Kumar ve diğ. [139] de *Credit Card Fraud* veri kümesi ve rastgele orman algoritmasını kullanarak dolandırıcılık işlemlerini tespit etmeyi hedefleyen bir model oluşturmuşlardır. Model değerlendirme ölçütü olarak sadece doğruluk kullanılmıştır ve 90 olduğu aktarılmıştır. FuzzyCSampling yöntemiyle doğruluk 96 hesaplanmıştır. Sailusha ve diğ. [140] *Credit Card Fraud* veri kümesi ile hem aşağı örnekleme hem de yukarı örnekleme yöntemlerini birlikte kullanarak k-en yakın komşu algoritmasıyla doğruluğu 97,9, kesinliği 100, duyarlılığı 94 olarak hesaplamışlardır. FuzzyCSampling yöntemiyle geliştirilen modelde doğruluk 94,5, kesinlik 98,3 ve duyarlılık 83,5 olarak hesaplanmıştır. KDD Cup 1999 veri kümesi çok sınıflı sınıflandırma problemi örneği olup hedef sınıflardan iki adet seçilerek ikili sınıflandırma problemi için veri kümesi oluşturulmuştur. Elde edilen veri kümesini karşılaştıracak çalışmaya literatürde rastlanılmamıştır.

Yapılan analizler sonucu seçilen üç veri kümesi ile kurulan modeller için FuzzyCSampling yaklaşımının, k-en yakın komşu algoritması ile kurulan modellerin performansını rastgele

orman algoritması ve destek vektör makineleri ile oluşturulan modellere göre daha fazla geliştirdiği gözlemlenmiştir. Bu çalışma boyunca FuzzyCSampling yönteminin, veri kümesi boyutu veya dengesizlik oranından bağımsız olarak daha iyi performans gösterdiği görülmüştür. FuzzyCSampling performansı ile veri kümesi boyutu arasında bir genelleştirme yapabilmek için çeşitli veri kümeleri üzerinde daha fazla analiz yapılması gerektiği görülmüştür. FuzzyCSampling, dengesizlik oranından bağımsız olarak her zaman az da olsa model performansında bir iyileşme göstermiştir. F-ölçütü ve AUC performansını gözle görülür şekilde arttırıcı etki göstermiştir. Ayrıca FuzzyCSampling yönteminin, F-ölçütü ve AUC ölçütleri için çoğunlukla yukarı örnekleme stratejilerinden daha iyi performans gösterdiği görülmüştür. Bulgular veri kümesine bağlı olarak, önerilen çözümün aşağı örnekleme stratejisinden daha iyi performans verdiğini de göstermiştir.

FuzzyCSampling ayrıca *soft clustering* tekniklerinin bilgi kaybını azaltarak model performansını iyileştirebileceğini de ortaya koymuştur. Sonuçlara bakıldığında dengesiz veri kümeleri için model değerlendirme ölçütlerinden olan doğruluğun modelin yanıltıcı yorumlanmasına neden olabileceği kavramının da desteklendiği görülmüştür. Model kurulumlarında veri kümeleri *hold-out* yöntemi kullanılarak eğitim ve test kümesine ayrılmıştır. Veri kümelerinin yüzde 70'i eğitim için kullanılırken yüzde 30'u test için kullanılmıştır. Gelecek çalışmalarda çapraz geçerleme yöntemi (*cross validation* - CV) kullanılarak da model performans değerlendirmesi gerçekleştirilebilir.

Önerilen FuzzyCSampling yaklaşımı, belirtilen veri kümeleri için model performansını iyileştirmiş olsa da, eksiklikleri gidermek ve model performansını kapsamlı bir şekilde incelemek için daha fazla veri kümesi ile denemesi gerekmektedir. Bu tez kapsamında FuzzyCSampling yaklaşımı sadece yaygın kullanılan üç sınıflandırma algoritması ile test edilmiştir.

FuzzyCSampling yaklaşımındaki kümeleme algoritmaları kullanılırken sezgisel olarak belirlenen bir küme sayısı kullanır. Gelecekteki çalışmalar küme sayılarının otomatik olarak belirlenmesine odaklanacaktır. Önerilen yöntem ikili sınıflandırma problemleri için geliştirilmiş olup devam eden çalışmalarda çok sınıflı sınıflandırma problemlerinin çözümünde de kullanılması hedeflenmektedir.

Bu kapsamda dengesiz veri kümelerine ilişkin gerçekleştirilen bibliyometrik analiz ve yeni önerilen FuzzyCSampling yaklaşımının hem dengesiz veri kümesiyle çalışacak

akademisyenlere hem de sektörde çalışan uygulayıcılara ışık tutmasını umuyoruz.



KAYNAKLAR

- [1]. Ayodele, T. O., 2010, Types of Machine Learning Algorithms, *New Advances in Machine Learning*, 3, Yagang Zhang, Intech, Rijeka, Croatia, 19-48.
- [2]. Mohamed AE., 2017, Comparative study of four supervised machine learning techniques for classification, *International Journal of Applied*, 7(2).
- [3]. Melo-Acosta, G. E. , Duitama-Muñoz, F. ve Arias-Londoño, J. D., 2017, Fraud Detection in Big Data Using Supervised and Semi-Supervised Learning Techniques, *IEEE Colombian Conference on Communications and Computing (COLCOM)*, Cartagena, Colombia, 1-6.
- [4]. Zhang D., Huang H , Q. Chen, ve Jiang, Y. , 2007, A Comparison Study of Credit Scoring Models, *Third International Conference on Natural Computation (ICNC 2007)*, Haikou, China, 1, 15-18.
- [5]. Maraş, A. ve Aydin, S., 2016, Intercorrelation between Singular Spectrum of EEG Sub-Bands and Emotional States, *National Conference on Electrical, Electronics and Biomedical Engineering (ELECO)*, Bursa, Turkey, 486-490.
- [6]. Asri, H., Mousannif, H., Al Moatassime, H. ve Noel, T., 2016, Using Machine Learning Algorithms for Breast Cancer Risk Prediction and Diagnosis, *Procedia Computer Science*, 83, 1064-1069.
- [7]. Kumari, V. A. ve Chitra, R., 2013, Classification of Diabetes Disease Using Support Vector Machine, *International Journal of Engineering Research and Applications*, 3(2), 1797-1801.
- [8]. Burez, J. ve Van den Poel, D., 2009, Handling Class Imbalance in Customer Churn Prediction, *Expert Systems with Applications*, 36(3), 4626-4636.
- [9]. Anjaria, M. ve Guddeti, R. M. R., 2014, A Novel Sentiment Analysis of Social Networks Using Supervised Learning, *Social Network Analysis and Mining*, 4(1), 181.
- [10]. Trifonov, R., Gotseva, D. ve Angelov, V., 2017, Binary classification algorithms, *International Journal of Development Research*, 7(11), 16873-16879.
- [11]. Aly, M., 2005, Survey on multiclass classification methods, *Neural Netw*, 19(1-9), 2.
- [12]. de Carvalho, AC ve Freitas, A.A., 2009, A tutorial on multi-label classification techniques, *Foundations of computational intelligence volume*, 5, 177-195.
- [13]. Caruana, R. ve A. Niculescu-Mizil, 2006, An Empirical Comparison of Supervised Learning Algorithms, *Proceedings of the 23rd International Conference on Machine Learning*, Pittsburgh Pennsylvania, USA, 161-168.

- [14]. García, V., Mollineda, R. A. ve Sánchez, J. S., 2008, A New Performance Evaluation Method for Two-Class Imbalanced Problems, *Joint IAPR International Workshops on Statistical Techniques in Pattern Recognition (SPR) and Structural and Syntactic Pattern Recognition (SSPR)*, Orlando, FL, USA, 917-925.
- [15]. Sohony, I., Pratap, R ve Nambiar U., 2018, Ensemble learning for credit card fraud detection, *In Proceedings of the ACM India Joint International Conference on Data Science and Management of Data*, 289-294.
- [16]. Manek, AS., Samhitha, MR., Shruthy, S., Bhat, VH., Shenoy, PD., Mohan, MC. ve Patnaik LM., 2013, RePID-OK: spam detection using repetitive pre-processing, *In IEEE 2013 International Conference on Cloud & Ubiquitous Computing & Emerging Technologies*, 144-149.
- [17]. Gupta, S. ve Gupta MK., 2022, A comprehensive data-level investigation of cancer diagnosis on imbalanced data, *Computational Intelligence*, 38(1), 156-186.
- [18]. Padurariu, C. ve Breaban, ME., 2019, Dealing with data imbalance in text classification, *Procedia Computer Science*, 159, 736-745.
- [19]. Sun, Y., Wong, A. K. ve Kamel, M. S., 2009, Classification of Imbalanced Data: A Review, *International Journal of Pattern Recognition and Artificial Intelligence*, 23(04), 687-719.
- [20]. Leevy, J. L., Khoshgoftaar, T. M., Bauder, R. A. ve Seliya, N., 2018, A Survey on Addressing High-Class Imbalance in Big Data, *Journal of Big Data*, 5(1), 1-30.
- [21]. Provost, F., 2000, Machine Learning from Imbalanced Data Sets 101, *Proceedings of the AAAI'2000 Workshop on Imbalanced Data Sets*, Austin, Texas, USA, 68, 1-3.
- [22]. Chawla, N. V., Japkowicz, N. ve Koltcz, A., 2003, *Proceedings of the ICML'2003 Workshop on Learning from Imbalanced Data Sets*, Washington, DC, USA.
- [23]. Zięba, M. ve Tomczak, J. M., 2015, Boosted SVM with Active Learning Strategy for Imbalanced Data, *Soft Computing*, 19(12), 3357-3368.
- [24]. He, H. ve Garcia, E. A., 2009, Learning from Imbalanced Data, *IEEE Transactions on knowledge and data engineering*, 21(9), 1263-1284.
- [25]. Wang, S., Li, Z., Chao, W. ve Cao, Q., 2012, Applying Adaptive Over-Sampling Technique based on Data Density and Cost-Sensitive SVM to Imbalanced Learning, *The 2012 International Joint Conference on Neural Networks (IJCNN)*, Brisbane, QLD, Australia, 1-8.
- [26]. Belarouci, S. ve Chikh, M. A., 2017, Medical Imbalanced Data Classification, *Advances in Science, Technology and Engineering Systems Journal*, 2(3), 116-124.
- [27]. Yan, Y., Yang, T., Yang, Y. ve Chen, J., 2017, A Framework of Online Learning with Imbalanced Streaming Data, *Thirty-First AAAI Conference on Artificial Intelligence*, San Francisco, California USA, 2817-2823.

- [28]. Orriols-Puig, A. ve Bernadó-Mansilla, E., 2009, Evolutionary rule-based Systems for Imbalanced Data Sets, *Soft Computing*, 13(3), 213.
- [29]. Weiss, G. M., 2004, Mining with Rarity: a Unifying Framework, *ACM Sigkdd Explorations Newsletter*, 6(1), 7-19.
- [30]. Jo, T. ve Japkowicz, N., 2004, Class imbalances versus small disjuncts, *ACM Sigkdd Explorations Newsletter*, 6(1), 40-49.
- [31]. Prati, R. C., Batista, G. E. ve Monard, M. C., 2004, Class Imbalances versus Class Overlapping: An Analysis of a Learning System Behavior, *Mexican International Conference on Artificial Intelligence*, Mexico City, Mexico, 312-321.
- [32]. Khoshgoftaar, T. M., Van Hulse, J. ve Napolitano, A., 2010, Comparing Boosting and Bagging Techniques with Noisy and Imbalanced Data, *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans*, 41(3), 552-568.
- [33]. More, A. S. ve Rana, D. P., 2017, Review of Random Forest Classification Techniques to Resolve Data Imbalance, *IEEE 1st International Conference on Intelligent Systems and Information Management (ICISIM)*, Aurangabad, India, 72-78.
- [34]. L'ope, V., Fernandez, A., Garcia, S., Palade, V. ve Herrera F., 2013, An insight into classification with imbalanced data: Empirical results and current trends on using data intrinsic characteristics, *Information sciences*, 250, 113-141.
- [35]. Chawla, NV., 2009, Data mining for imbalanced datasets: An overview, *Data mining and knowledge discovery handbook*, 875-886.
- [36]. Salunkhe, UR. ve Mali, SN., 2016, Classifier ensemble design for imbalanced data classification: a hybrid approach, *Procedia Computer Science*, 85, 725-732.
- [37]. Lin, CX. ve Sheng, VS., 2011, Cost-sensitive learning and the class imbalance problem, *Encyclopedia of machine learning*, 231-235.
- [38]. Ng, J. J. ve Chai, K. H., 2015, A Bibliometric Analysis of Project Management Research, *IEEE International Conference on Industrial Engineering and Engineering Management (IEEM)*, Singapore, 976-980.
- [39]. Machado, F. ve Martes, C. D., 2015, Project Management Success: A Bibliometric Analysis, *Revista de Gestão e Projetos-GeP*, 6(1), 28-44.
- [40]. MARAŞ, A. ve EROL, Ç., 2022, Emerging Trends in Classification with Imbalanced Datasets: A Bibliometric Analysis of Progression, *Bilişim Teknolojileri Dergisi*, 15(3), 275-288.
- [41]. Internet: Scopus (2022) , <https://www.scopus.com/>, [Ziyaret Tarihi: 26 Mayıs 2022].
- [42]. Aria, M. ve Cuccurullo, C., 2017, Bibliometrix: An R-tool for Comprehensive Science Mapping Analysis, *Journal of Informetrics*, 11(4), 959-975.
- [43]. Chawla, N. V., Bowyer, K. W., Hall, L. O. ve Kegelmeyer, W. P., 2002, SMOTE: Synthetic Minority Over-Sampling Technique, *Journal of Artificial Intelligence Research*, 16, 321-357.

- [44]. Han, H., Wang, W. Y. ve Mao, B. H., 2005, Borderline-SMOTE: A New Over-Sampling Method in Imbalanced Data Sets Learning, *Advances in Intelligent Computing. ICIC. Lecture Notes in Computer Science*, Hefei, China, 878-887.
- [45]. Chawla, N. V., Lazarevic, A. Hall, L. O. ve Bowyer, K. W., 2003, SMOTEBoost: Improving Prediction of the Minority Class in Boosting, *European Conference on Principles of Data Mining and Knowledge Discovery*, Cavtat-Dubrovnik, Croatia, 107-119.
- [46]. Galar, M. Fernandez, A., Barrenechea, E., Bustince, H. ve Herrera, F., 2011, A Review on Ensembles for the Class Imbalance Problem: Bagging-, Boosting-, and Hybrid-based Approaches, *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 42(4), 463-484.
- [47]. Liu, X. Y., Wu, J. ve Zhou, Z. H., 2008, Exploratory Undersampling for Class Imbalance Learning, *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 39(2), 539-550.
- [48]. López, V., Fernandez, A., Garcia, S., Palade, V. ve Herrera, F., 2013, An Insight into Classification with Imbalanced Data: Empirical Results and Current Trends on Using Data Intrinsic Characteristics, *Information Sciences*, 250, 113-141.
- [49]. Akbani, R., Kwek, S. ve Japkowicz, N., 2004, Applying support vector machines to imbalanced datasets, *European Conference on Machine Learning*, Pisa, Italy, 39-50.
- [50]. Veropoulos, K., Campbell, C. ve Cristianini, N., 1999, Controlling the sensitivity of support vector machines, *Proceedings of the International Joint Conference on AI*, 55-60.
- [51]. Seiffert, C., Khoshgoftaar, T. M., Van Hulse, J. ve Napolitano, A., 2009, RUSBoost: A Hybrid Approach to Alleviating Class Imbalance, *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans*, 40(1), 185-197.
- [52]. Zhou, Z. H. ve Liu, X. Y., 2005, Training Cost Sensitive Neural Networks with Methods Addressing the Class Imbalance Problem, *IEEE Transactions on Knowledge and Data Engineering*, 18(1), 63-77.
- [53]. Estabrooks, A., Jo, T. ve Japkowicz, N., 2004, A Multiple Resampling Method for Learning from Imbalanced Data Sets, *Computational Intelligence*, 20(1), 18-36.
- [54]. Tang, Y., Zhang, Y. Q. N. V. Chawla ve Krasser, S., 2008, SVMs Modeling for Highly Imbalanced Classification, *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 39(1), 281-288.
- [55]. Bunkhumpornpat, C., Sinapiromsaran, K. ve Lursinsap, C., 2009, Safe-Level-Smote: Safe-Level-Synthetic Minority Over-Sampling Technique for Handling the Class Imbalanced Problem, *Pacific-Asia Conference on Knowledge Discovery and Data Mining, Bangkok, Thailand*, 475-482.
- [56]. Wang, S. ve Yao, X., 2009, Diversity Analysis on Imbalanced Data Sets by Using Ensemble Models, *IEEE Symposium on Computational Intelligence and Data Mining*, Nashville, TN, USA 324-331.

- [57]. Mazurowski, M. A., Habas, P. A., Zurada, J. M., Lo, J. Y., Baker, J. A. ve Tourassi, G. D., 2008, Training Neural Network Classifiers for Medical Decision Making: The Effects of Imbalanced Datasets on Classification Performance, *Neural Networks*, 21(2-3), 427-436, 2008.
- [58]. Galar, M, Fernández, A., Barrenechea, E. ve Herrera, F., 2013, EUSBoost: Enhancing Ensembles for Highly Imbalanced Data-Sets by Evolutionary Undersampling, *Pattern Recognition*, 46(12), 3460-3471.
- [59]. Fernández, A., Garcia, S., Herrera, F. ve Chawla, N. V., 2018, SMOTE for Learning from Imbalanced Data: Progress and Challenges, Marking the 15-year Anniversary, *Journal of Artificial Intelligence Research*, 61, 863-905.
- [60]. Triguero, I., Galar, M., Vluymans, Cornelis, S.C., Bustince, Herrera, H.F. ve Saeys, Y., 2015, Evolutionary Undersampling for Imbalanced Big Data Classification, *IEEE Congress on Evolutionary Computation (CEC)*, Sendai, Japan, 715-722.
- [61]. Sáez, J. A., J. Luengo, Stefanowski, J. ve Herrera, F., 2015, SMOTE-IPF: Addressing the Noisy and Borderline Examples Problem in Imbalanced Classification by a Re-sampling Method with Filtering, *Information Sciences*, 291, 184-203.
- [62]. Blagus, R. ve Lusa, L., 2013, Improved Shrunk Centroid Classifiers for High Dimensional Class Imbalanced Data, *BMC Bioinformatics*, 14(1), 1-13.
- [63]. Brown, I. ve Mues, C., 2012, An Experimental Comparison of Classification Algorithms for Imbalanced Credit Scoring Data Sets, *Expert Systems with Applications*, 39(3), 3446-3453.
- [64]. Garfield, E., 2004, Historiographic Mapping of Knowledge Domains Literature, *Journal of Information Science*, 30(2), 119-145.
- [65]. Fernández-Navarro, F., Hervás-Martínez, C., García-Alonso, C. ve Torres-Jiménez, M., 2011, Determination of Relative Agrarian Technical Efficiency by a Dynamic Over-Sampling Procedure Guided by Minimum Sensitivity, *Expert Systems with Applications*, 38(10), 12483-12490, 2011.
- [66]. Kovács, G., 2019, An Empirical Comparison and Evaluation of Minority Oversampling Techniques on a Large Number of Imbalanced Datasets, *Applied Soft Computing*, 83, 105662.
- [67]. Torres-Vásquez, M., Chávez-Bosquez, O., Hernández-Ocaña, B. ve Hernández-Torruco, J., 2020, Classification of guillain-barré syndrome subtypes using sampling techniques with binary approach, *Symmetry*, 12(3), 482.
- [68]. Del Río, S., López, V., Benítez, J. M. ve Herrera, F., 2014, On the Use of Mapreduce for Imbalanced Big Data Using Random Forest, *Information Sciences*, 285, 112-137.
- [69]. Yijing, L., Haixiang, G., Xiao, L., Yanan, L. ve Jinling, L. 2016, Adapted Ensemble Classification Algorithm based on Multiple Classifier System and feature selection for classifying multi-class imbalanced data, *Knowledge-Based Systems*, 94, 88-104.

- [70]. Kang, Q., Chen, X., Li, S. ve Zhou, M., 2016, A Noise-Filtered Under-sampling Scheme for Imbalanced Classification, *IEEE Transactions on Cybernetics*, 47(12), 4263-4274.
- [71]. Hasanin, T. ve Khoshgoftaar, T., 2018, The Effects of Random Undersampling with Simulated Class Imbalance for Big Data, *IEEE International Conference on Information Reuse and Integration (IRI)*, Salt Lake City, UT, USA, 70-79.
- [72]. Hasanin, T., Khoshgoftaar, T., Leevy, J. ve Seliya, N., 2019, Investigating Random Undersampling and Feature Selection on Bioinformatics Big Data, *IEEE Fifth International Conference on Big Data Computing Service and Applications (BigDataService)*, Newark, CA, USA, 346-356.
- [73]. Abdel-Hamid, N. B., ElGhamrawy, S., El Desouky, A. ve Arafat, H., 2018, A Dynamic Spark-Based Classification Framework for Imbalanced Big Data, *Journal of Grid Computing*, 16(4), 607-626.
- [74]. Esfahani, H., Tavasoli, K. ve Jabbarzadeh, A., 2019, Big Data and Social Media: A Scientometrics Analysis, *International Journal of Data and Network Science*, 3(3), 145-164.
- [75]. Öztürk, A., 2009, *Dengesiz veri kümelerinde çok ölçütlü optimizasyon çerçevesinde SVM ile sınıflandırma*, Yüksek Lisans, Fen Bilimleri Enstitüsü.
- [76]. Ayhan, D., 2009, *Mahalanobis Taguchi sistemi ile çoklu sınıflandırma yöntemleri ve dengeli olmayan veri setleri için bir yeniden örnekleme yaklaşımı*, Yüksek Lisans, Fen Bilimleri Enstitüsü.
- [77]. Aydın, Y., 2016, *Dengesiz veri setlerinde trafik işaretlerini tanıma*, Yüksek Lisans, Fen Bilimleri Enstitüsü.
- [78]. Gümüştaş, E., 2019, *Kayıp gözlem içeren dengesiz veri setlerinin topluluk öğrenme algoritmaları ile sınıflandırılması*, Yüksek Lisans, Fen Bilimleri Enstitüsü.
- [79]. Al-Shamaa, Z. Z. R., 2020, *Dengesiz veri kümesinde nadir sınıfın sınıflandırılmasını geliştirmek için alt örnekleme modelleri*, Doktora, Fen Bilimleri Enstitüsü.
- [80]. Sağlam, F., 2020, *Gürültülü gözlemler durumunda dengesiz veride öğrenme için yeni bir yaklaşım*, Yüksek Lisans, Fen Bilimleri Enstitüsü.
- [81]. Dikdere, F., 2021, *Yapay sinir ağı kullanılarak dengesiz veri kümelerinde sınıflandırma başarısının artırılması*, Yüksek Lisans, Fen Bilimleri Enstitüsü.
- [82]. Oğul, H. A., 2021, *Dengesiz veri kümelerinde duygu analizi*, Yüksek Lisans, Lisansüstü Eğitim Enstitüsü.
- [83]. Mulla, G. A. A., 2021, *Yüksek boyutlu dengesiz verilerin sınıflandırılması için smote aşırı örnekleme ile PCA'nın kombinasyonu*, Yüksek Lisans, Fen Bilimleri Enstitüsü.
- [84]. Öztürk, A., 2021, *Ağaç tabanlı sınıflandırma tekniklerinin dengeli olmayan veri setlerinde incelenmesi: Kredi risk örneği*, Yüksek Lisans, Sosyal Bilimler Enstitüsü.

- [85]. Öztürk, H., 2022, *Dengesiz veri setlerinde farklı dengeleme algoritmalarının optimum denge oranlarının sınıflandırma ve regresyon ağaçları yöntemi ile incelenmesi: simülasyon çalışması*, Doktora, Sağlık Bilimleri Enstitüsü.
- [86]. Arık, A. O., 2022, *Saldırı tespit sistemleri için dengesiz veri setlerinde smote ve near miss metotlarına dayalı güçlü gradyan artırma modeli*, Yüksek Lisans, Lisansüstü Eğitim Enstitüsü.
- [87]. Pir, M. Ş., 2022, *Dengesiz veri setlerinde sınıflandırma problemlerinin çözümünde melez yöntem uygulaması*, Yüksek Lisans, Lisansüstü Eğitim Enstitüsü.
- [88]. Jourdan, Z., Rainer, R.K. ve Marshall, 2008, Business intelligence: An analysis of the literature, *Information systems management*, 25(2), 121-131.
- [89]. Kartal, E. ve Balaban, M.E., 2015, *Sınıflandırmaya dayalı makine öğrenmesi teknikleri ve kardiyolojik risk değerlendirmesine ilişkin bir uygulama*, Doktora, Fen Bilimleri Enstitüsü.
- [90]. Mitchell, T. M., 1997, *Machine Learning*, 1st Edition, McGraw-Hill Science/Engineering/Math, ISBN: 0-07-042807-7.
- [91]. Jordan, M.I. ve Mitchell, T.M., 2015, Machine learning: Trends, perspectives, and prospects. *Science*, 349(6245), 255-260.
- [92]. Van Engelen, J.E. ve Hoos, H.H., 2020, A survey on semi-supervised learning, *Machine Learning*, 109(2), 373-440.
- [93]. Ghahramani, Z., 2003, Unsupervised learning, In *Summer school on machine learning*, Springer, Berlin, Heidelberg, 72-112.
- [94]. Sutton, R.S., 1992, Introduction: The challenge of reinforcement learning, In *Reinforcement Learning*, Springer, Boston, MA, 1-3.
- [95]. Zhu, X. and Goldberg, A.B., 2009, Introduction to semi-supervised learning, *Synthesis lectures on artificial intelligence and machine learning*, 3(1), 1-130.
- [96]. Emre, i.e. ve Erol, Ç., 2021, *Psikiyatrik hastalıkların makine öğrenmesi yöntemleri ile ayrıştırılması*, Doktora, Fen Bilimleri Enstitüsü.
- [97]. Dridi, S., 2021, Supervised Learning-A Systematic Literature Review.
- [98]. Khashei, M., Eftekhari, S. ve Parvizian, J., 2012, Diagnosing diabetes type II using a soft intelligent binary classification model, *Review of Bioinformatics and Biometrics*, 1(1), pp.9-23.
- [99]. Hussain, E., Hasan, M., Hassan, S.Z., Azmi, T.H., Rahman, M.A. ve Parvez, M.Z., 2020, Deep learning based binary classification for alzheimer's disease detection using brain mri images, In *2020 15th IEEE Conference on Industrial Electronics and Applications (ICIEA)*, 1115-1120.
- [100]. Gürcan, F., 2018, Multi-class classification of turkish texts with machine learning algorithms, In *2018 2nd International Symposium on Multidisciplinary Studies and Innovative Technologies (ISMSIT)*, 1-5.

- [101]. Rajoub, B., 2020, Supervised and unsupervised learning, *In Biomedical Signal Processing and Artificial Intelligence in Healthcare*, 51-89.
- [102]. Burkardt, J., 2009, K-means clustering, Virginia Tech, Advanced Research Computing, Interdisciplinary Center for Applied Mathematics.
- [103]. Kansal, T., Bahuguna, S., Singh, V. ve Choudhury, T., 2018, Customer segmentation using K-means clustering, *In 2018 international conference on computational techniques, electronics and mechanical systems (CTEMS)*, 135-139.
- [134]. Abdi, H. ve Williams, L.J., 2010, Principal component analysis, *Wiley interdisciplinary reviews: computational statistics*, 2(4), 433-459.
- [105]. Zhu, X.J., 2005, Semi-supervised learning literature survey.
- [106]. Guillaumin, M., Verbeek, J. ve Schmid, C., 2010, Multimodal semi-supervised learning for image classification, *IEEE Computer society conference on computer vision and pattern recognition*, 902-909.
- [107]. Dayan, P. ve Niv, Y., 2008, Reinforcement learning: the good, the bad and the ugly, *Current opinion in neurobiology*, 18(2), 185-196.
- [108]. Holcomb, S.D., Porter, W.K., Ault, S.V., Mao, G. ve Wang, J., 2018, Overview on deepmind and its alphago zero ai, *In Proceedings of the 2018 international conference on big data and education*, 67-71.
- [109]. Cunningham, P. ve Delany, S. J., 2021, K-nearest neighbour classifiers-a tutorial, *ACM Computing Surveys (CSUR)*, 54(6), 1-25.
- [110]. Demirkol, D., 2022, *Diyabetik ayak hastalarının amputasyon riskinin yapay zeka teknikleri ile öngörülmesi*, Doktora, Fen Bilimleri Enstitüsü.
- [111]. Silahtaroglu, G., 2016, *Veri Madenciliği Kavram ve Algoritmaları*, Papatya Bilim Yayıncılık.
- [112]. Pisner, D.A. ve Schnyer, D.M., 2020, *Support vector machine*, Machine learning, 101-121.
- [113]. Cortes, C. ve Vapnik, V., 1995, Support-vector network, *Machine learning*, 20, 1–25.
- [114]. Meyer, D. ve Wien, F.T., 2015, Support vector machines. The Interface to libsvm in package e1071, 28, 20.
- [115]. Ghosh, S., Dasgupta, A. ve Swetapadma, A., 2019, A study on support vector machine based linear and non-linear pattern classification, *In 2019 International Conference on Intelligent Sustainable Systems (ICISS)*, 24-28.
- [116]. Breiman, Leo., 2001, Random forests, *Machine learning*, 45.1, 5-32.
- [117]. Zounemat-Kermani, M., Batelaan, O., Fadaee, M. ve Hinkelmann, R., 2021, Ensemble machine learning paradigms in hydrology: A review, *Journal of Hydrology*, 598, 126266.

- [118]. Wang, C., Deng, C. ve Wang, S., 2020, Imbalance-XGBoost: leveraging weighted and focal losses for binary label-imbalanced classification with XGBoost, *Pattern Recognition Letters*, 136, 190-197.
- [119]. Charbuty, B. ve Abdulazeez, A., 2021, Classification based on decision tree algorithm for machine learning, *Journal of Applied Science and Technology Trends*, 2(01), 20-28.
- [120]. Maji, S. ve Arora, S., 2019, Decision tree algorithms for prediction of heart disease, *In Information and communication technology for competitive strategies*, 447-454.
- [121]. Ghosal, A., Nandy, A., Das, A.K., Goswami, S. ve Panday, M., 2020, A short review on different clustering techniques and their applications, *Emerging technology in modelling and graphics*, 69-83.
- [122]. Yuan, C. ve Yang, H., 2019, Research on K-value selection method of K-means clustering algorithm, 2(2), 226-235.
- [123]. Bezdek, J.C., Ehrlich, R. ve Full, W., 1984, FCM: The fuzzy c-means clustering algorithm, *Computers and geosciences*, 10(2-3), 191-203.
- [124]. Bora, D.J., Gupta, D. ve Kumar, A., 2014, A comparative study between fuzzy clustering algorithm and hard clustering algorithm, arXiv preprint arXiv:1404.6059.
- [125]. Ferri, C., Hern´andez-Orallo, J. ve Modroi, R., 2009, An experimental comparison of performance measures for classification, *Pattern recognition letters*, 30(1), 27-38.
- [126]. Japkowicz, N., 2013, Assessment metrics for imbalanced learning, *Imbalanced learning: Foundations, algorithms, and applications*, 187-206.
- [127]. Japkowicz, N. ve Shah, M., 2011, Evaluating Learning Algorithms: A Classification Perspective, Cambridge University Press
- [128]. Mohammed, R., Rawashdeh, J. ve Abdullah, M., 2020, Machine learning with oversampling and undersampling techniques: overview study and experimental results, *In 2020 11th international conference on information and communication systems (ICICS)*, 243-248
- [129]. Vilorio, A., Lezama, O.B.P. ve Mercado-Caruzo, N., 2020, Unbalanced data processing using oversampling: Machine learning, *Procedia Computer Science*, 175, 108-113.
- [130]. Learning, U. C. I. M., 2016, Pima Indians Diabetes Database. Kaggle, URL <https://www.kaggle.com/uciml/pima-indians-diabetes-database> (Ziyaret Tarihi: 02.04.2020).
- [131]. KDD Cup 1999 Data, 2022, <http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html> (Ziyaret Tarihi: 27.08.2022).
- [132]. ULB, M. L. G., 2020, Credit Card Fraud Detection. Kaggle, URL <https://www.kaggle.com/mlg-ulb/creditcardfraud> (Ziyaret Tarihi: 28.04.2020)

- [133]. Staudemeyer, R.C. ve Omlin, C.W., 2014, Extracting salient features for network intrusion detection using machine learning methods, *South African computer journal*, 52(1), 82-96.
- [134]. Abdi, H. ve Williams, L.J., 2010, Principal component analysis, *Wiley interdisciplinary reviews: computational statistics*, 2(4), 433-459.
- [135]. Raju, V.G., Lakshmi, K.P., Jain, V.M., Kalidindi, A. ve Padma, V., 2020, Study the influence of normalization/transformation process on the accuracy of supervised classification, *In 2020 Third International Conference on Smart Systems and Inventive Technology (ICSSIT)*, 729-735.
- [136]. Batista, G.E., Prati, R.C. Ve Monard, M.C., 2004, A study of the behavior of several methods for balancing machine learning training data, *ACM SIGKDD explorations newsletter*, 6(1), 20-29.
- [137]. Tao, X., Li, Q., Ren, C., Guo, W., He, Q., Liu, R. ve Zou, J., 2020, Affinity and class probability-based fuzzy support vector machine for imbalanced data sets, *Neural Networks*, 122, 289-307.
- [138]. Chang, V., Bailey, J., Xu, Q.A. ve Sun, Z., 2022, Pima Indians diabetes mellitus classification based on machine learning (ML) algorithms, *Neural Computing and Applications*, 1-17.
- [139]. Kumar MS, Soundarya V, Kavitha S, Keerthika ES ve Aswini E 2019, Credit card fraud detection using random forest algorithm, *3rd Int. Conf. Computing and Communications Technologies (IEEE)*, 149-53.
- [140]. Sailusha R, Gnaneswar V, Ramesh R ve Rao GR 2020, Credit card fraud detection using machine learning, *4th Int. Conf. Intelligent Computing and Control Systems (IEEE)*, 1264-70.
- [141]. Mahesh, K.P., Afrouz, S.A. ve Areeckal, A.S., 2022, Detection of fraudulent credit card transactions: A comparative analysis of data sampling and classification techniques, *In Journal of Physics: Conference Series* , 2161, 1, 012072.

ÖZGEÇMİŞ

Kişisel Bilgiler			
Adı Soyadı	Abdullah MARAŞ		
Uyruğu	☒ T.C.	☐	Diğer:

Eğitim Bilgileri	
Lisans	
Üniversite	İstanbul Teknik Üniversitesi
Fakülte	Elektrik-Elektronik Fakültesi
Bölümü	Elektronik ve Haberleşme Mühendisliği
Programı	Elektronik Mühendisliği
Mezuniyet Yılı	2012

Yüksek Lisans	
Üniversite	Bahçeşehir Üniversitesi
Enstitü Adı	Fen Bilimleri
Programı	Bilgi Teknolojileri
Mezuniyet Tarihi	2016

Doktora	
Üniversite	İstanbul Üniversitesi
Enstitü Adı	Fen Bilimleri
Anabilim Dalı	Enformatik Anabilim Dalı
Programı	Enformatik Programı
Mezuniyet Tarihi	2023

Makale ve Bildiriler
Makaleler MARAŞ, A. ve EROL, Çiğdem, 2022, Emerging Trends in Classification with Imbalanced Datasets: A Bibliometric Analysis of Progression, <i>Bilişim Teknolojileri Dergisi</i>, 15(3), 275-288. Bildiriler MARAŞ, A. ve Aydın, S., 2017, Discrimination of psychotic symptoms from controls through data mining methods based on emotional principle components, <i>In CMBEBIH 2017</i>, 26-30, Springer, Singapore.