

**PERFORMANCE INVESTIGATION OF MACHINE
LEARNING METHODS IN 5G AND BEYOND NETWORK
SLICING**

**5G VE ÖTESİ AĞ DİLİMLEMESİNDE MAKİNE
ÖĞRENMESİ YÖNTEMLERİNİN BAŞARIMININ
İNCELENMESİ**

ALPER ENDEŞ

DR. BARIŞ YÜKSEKKAYA

Tez Danışmanı

Lisansüstü Eğitim-Öğretim ve Sınav Yönetmeliğinin Elektrik ve Elektronik Mühendisliği
Anabilim Dalı için Öngördüğü YÜKSEK LİSANS TEZİ olarak hazırlanmıştır.

2022

ABSTRACT

PERFORMANCE INVESTIGATION OF MACHINE LEARNING METHODS IN 5G AND BEYOND NETWORK SLICING

Alper ENDEŞ

Master of Science, Electrical and Electronics Engineering

Supervisor: Dr. Barış Yüksekaya

September 2022, 97 pages

Thanks to the rapid developments in the communication sector in recent years, new applications, and opportunities have emerged. These applications have created the need for low latency, high data rate, high reliability, and security. New proposals had to be given for these needs that the fourth generation (4G) communication systems could not meet, and fifth generation (5G) communication systems emerged. For example, while software defined networking (SDN), and network function virtualization (NFV) technologies are used for programmable structures in 4G systems, it is planned to use network slicing methods in addition to these two technologies in future 5G systems. In this study, 5G Network Slicing, Data Rate Management, and User Handover Mechanisms were created with machine learning models, and additional mechanisms. In order to test these mechanisms, a realistic simulation environment has been developed. 3rd Generation Partnership Project (3GPP) compatible users and base stations are placed in this simulation environment. Five different machine learning models were used in the study, and four different models for network slicing, and a classification mechanism based on the majority of these models' decisions were implemented. In order to test the performance of the created machine learning models in realistic scenarios, user data with margin of error was created. With this user data,

classification has been carried out with data with error margin that can be experienced in real situations. Classifications were performed between the enhanced mobile broadband (eMBB), ultra-reliable low latency communications (URLLC), massive internet of things (MIoT), and vehicle to everything (V2X) network slices that are proposed by 3GPP. The implemented mechanisms were tested in different scenarios, and their performances were compared.

Keywords: 5G, network slicing, classification, machine learning, communication systems, mechanism.

ÖZET

5G VE ÖTESİ AĞ DİLİMLEMESİNDE MAKİNE ÖĞRENMESİ YÖNTEMLERİNİN BAŞARIMININ İNCELENMESİ

Alper ENDEŞ

Yüksek Lisans, Elektrik ve Elektronik Mühendisliği

Danışman: Dr. Barış Yüksekaya

Eylül 2022, 97 sayfa

Son yıllarda haberleşme sektöründeki hızlı gelişmeler sayesinde, yeni uygulamalar ve imkanlar ortaya çıkmıştır. Bu uygulamalar, düşük gecikme, yüksek veri hızı, yüksek güvenilirlik ve güvenlik ihtiyacı doğurmuştur. Dördüncü nesil (fourth generation, 4G) iletişim sistemlerinin karşılayamadığı bu ihtiyaçlar için yeni öneriler verilmesi gerekmiştir ve beşinci nesil (fifth generation, 5G) haberleşme sistemleri ortaya çıkmıştır. Örneğin 4G sistemlerde programlanabilir yapılar için yazılım tanımlı ağ (software defined networking, SDN) ve ağ fonksiyonlarının sanallaştırılması (network function virtualization, NFV) teknolojileri kullanılırken, gelecek 5G sistemlerinde bu iki teknolojinin yanı sıra Ağ dilimleme yöntemlerinin de kullanılması planlanmıştır. Bu çalışmada, makine öğrenmesi ile oluşturulmuş modeller ve ek mekanizmalar ile 5G Ağ Dilimleme, Veri Hızı Yönetim ve Kullanıcı Devir Mekanizmaları oluşturulmuştur. Oluşturulan bu mekanizmaların test edilebilmesi için gerçekçi bir benzetim ortamı geliştirilmiştir. Bu benzetim ortamına 3. Nesil Ortaklık Projesi (3rd Generation Partnership Project, 3GPP) uyumlu kullanıcılar ve baz istasyonları yerleştirilmiştir. Çalışmada beş farklı makine öğrenmesi modeli kullanılmıştır, ağ dilimleme için dört farklı model ve modellerin karar çoğunluğu ile çalışan bir sınıflandırma modeli geliştirilmiştir. Oluşturulan makine öğrenmesi modellerinin

gerçekçi senaryolardaki performansını test edebilmek için, hata payı içeren kullanıcı verisi oluşturulmuştur. Bu kullanıcı verisi ile gerçek durumlarda yaşanabilecek hata paylı veri ile sınıflandırma gerçekleştirilmiştir. Sınıflandırmalar, 3GPP tarafından önerilen gelişmiş mobil şebeke (enhanced mobile broadband, eMBB), çok düşük gecikmeli veri iletişimi (ultra reliable low latency communications, URLLC), büyük nesnelerin interneti (massive internet of things, MIoT) ve araçtan her şeye (vehicle to everything, V2X) ağ dilimleri arasında gerçekleştirilmiştir. Geliştirilen mekanizmalar, farklı senaryolarda test edilip, performansları karşılaştırılmıştır.

Keywords: 5G, ağ dilimleme, sınıflandırma, makine öğrenmesi, haberleşme sistemleri, mekanizma.

TEŐEKKÜR

Tez alıőmam boyunca deęerli katkılarını ve rehberlięini esirgemeyen danıőman hocam Dr. Barıő Yůksekaya'ya teőekkůrlerimi sunarım.

Hayatım boyunca bana baőarma gůcũ ve inancı aőılayan, her an yanımda olan aileme teőekkůr ederim.

Bu alıőma Bilgi Teknolojileri ve İletiőim Kurumu'nun 5G ve Őtesi Yůksek Lisans Destekleme Programı kapsamında Vodafone tarafından desteklenmiőtir. Onların bu desteęi, tez ve dięer akademik alıőmalara yoęunlaőmam konusunda önemli olmuőtur.

İçindekiler

	<u>Sayfa</u>
ABSTRACT	i
ÖZET	iii
TEŞEKKÜR	v
İÇİNDEKİLER	vi
TABLolar	ix
ŞEKİLLER	x
KISALTMALAR	xii
1. GİRİŞ	1
1.1. Tezin Amacı	2
1.2. Tezin Katkıları	2
1.3. Tezin Akışı	3
2. HABERLEŞME SİSTEMLERİNİN GELİŞİMİ	4
2.1. Birinci Nesil (1G)	4
2.2. İkinci Nesil (2G)	4
2.3. Üçüncü Nesil (3G)	4
2.4. Dördüncü Nesil (4G)	5
2.5. Beşinci Nesil (5G)	5
2.6. Literatürde 5G Teknolojisinin Gelişimi	6
3. 5G AĞ DİLİMLEME	8
3.1. Yazılım Tanımlı Ağ	8
3.2. Ağ Fonksiyonlarının Sanallaştırılması	9
3.3. Ağ Dilimleme	9
3.4. Literatürde Ağ Dilimleme Çalışmaları	10
4. MAKİNE ÖĞRENMESİ MODELLERİ	14
4.1. Veri Ön İşleme	15
4.1.1. Normalizasyon (Düzgeleme)	16
4.1.2. Tek-sıcak kodlama	18

4.1.3. Etiket kodlama	19
4.2. K-Ortalama Kümelemesi.....	19
4.3. Yapay Sinir Ağı.....	20
4.3.1. Aktivasyon Fonksiyonları	22
4.3.1.1. Doğrutulmuş Doğrusal Birim (Rectified Linear Unit - ReLU) Fonksiyonu.....	22
4.3.1.2. Sigmoid Fonksiyonu	22
4.3.1.3. Softmax Fonksiyonu	24
4.3.1.4. Tanh (Hiperbolik Tanjant) Fonksiyonu	24
4.3.2. Katmanlar	24
4.3.2.1. Yoğun Katman	25
4.3.2.2. Seyreltme Katmanı	26
4.4. k-En Yakın Komşu	28
4.5. Destek Vektör Makineleri	30
4.6. Rastgele Orman	31
4.6.1. Karar Ağacı	31
4.7. XGBoost - eXtreme Gradient Boosting (Aşırı Gradyan Artırma)	34
5. MAKİNE ÖĞRENMESİ TEKNİKLERİ İLE 5G AĞ DİLİMLEME	36
5.1. Sistem Modeli	39
5.2. Baz İstasyonlarının Benzetim Ortamına Konumlandırılması.....	40
5.3. Kullanıcı ve Baz İstasyonları Arasındaki Kanal Parametresi ve İlişkilendirilmeleri.....	44
5.4. Ağ Dilimleme Mekanizması	47
5.4.1. Yapay Sinir Ağı	48
5.4.2. k-En Yakın Komşu	49
5.4.3. Destek Vektör Makineleri	49
5.4.4. XGBoost.....	50
5.4.5. Ana Ağ Dilimleme Mekanizması	51
5.5. Veri Hızı Yönetim Mekanizması.....	52
5.6. Baz İstasyonları Arasında Kullanıcı Devir Mekanizması.....	54

6. BENZETİM ÇALIŞMALARI VE SONUÇLAR.....	55
6.1. Hata Payı İçermeyen Kullanıcı Verileri.....	59
6.1.1. Yapay Sinir Ağı	59
6.1.2. k-En Yakın Komşu	60
6.1.3. Destek Vektör Makineleri	61
6.1.4. XGBoost - eXtreme Gradient Boosting (Aşırı Gradyan Artırma)	62
6.1.5. Karar Çoğunluğu Mekanizması	63
6.2. Hata Payı İçeren Kullanıcı Verileri	64
6.2.1. Yapay Sinir Ağı	64
6.2.2. k-En Yakın Komşu	65
6.2.3. Destek Vektör Makineleri	66
6.2.4. XGBoost - eXtreme Gradient Boosting (Aşırı Gradyan Artırma)	67
6.2.5. Karar Çoğunluğu Mekanizması	68
6.3. Benzetim Ortamı İstatistikleri.....	70
7. SONUÇ VE TARTIŞMA	72

TABLÖLAR

	<u>Sayfa</u>
Tablo 2.1 Haberleşme sistemleri nesillerinin özellikleri.....	6
Tablo 4.1 Normalizasyon ve tek-sıcak kodlama sonrası veri.	18
Tablo 4.2 Ağ dilimlerine karşılık gelen etiketler.	19
Tablo 4.3 Normalizasyon ve etiket kodlama sonrası veri.....	19
Tablo 5.1 Bazı kullanım tiplerini ve onların gereksinimlerini içeren 3GPP çizelgesi.....	36
Tablo 5.2 5G senaryoları içeren örnek sentetik kullanıcı verileri.	39
Tablo 5.3 Benzetim ortamında kullanılacak baz istasyonu verileri.	44
Tablo 6.1 Hata payı içermeyen veri ile model başarımları.....	64
Tablo 6.2 Hata payı içeren veri ile model başarımları.	69

ŞEKİLLER

	<u>Sayfa</u>
Şekil 3.1	Yazılım Tanımlı Ağ yapısı. 8
Şekil 3.2	Ağ Dilimleme yapısı. 10
Şekil 3.3	5G Ağ Dilimleme için <i>Web of Science</i> tarama sonuçları. 11
Şekil 4.1	Çalışmada kullanılan makine öğrenmesi yöntemleri 15
Şekil 4.2	Normalize edilmemiş veri ile sınıflandırma. 17
Şekil 4.3	Normalize edilmiş veri ile sınıflandırma. 17
Şekil 4.4	K-ortalama kümelemesi örneği. 20
Şekil 4.5	Yapay sinir ağları yapısı..... 21
Şekil 4.6	Doğrutulmuş doğrusal birim fonksiyonu. 23
Şekil 4.7	Sigmoid fonksiyonu. 23
Şekil 4.8	Hiperbolik tanjant fonksiyonu. 25
Şekil 4.9	Seyreltme katmanının öğrenme esnasında çalışması mekanizması. 26
Şekil 4.10	Oluşturulmuş yapay sinir ağı modeli. 28
Şekil 4.11	k-En Yakın Komşu çalışma örneği. 29
Şekil 4.12	Destek vektör makineleri örnek görseli. 30
Şekil 4.13	Karar ağacı örneği. 32
Şekil 4.14	Rastgele orman modeli çalışma mekanizması. 32
Şekil 4.15	Oluşturulan rastgele orman modelinin örnek kısmı..... 33
Şekil 4.16	XGBoost çalışma mekanizması..... 35
Şekil 5.1	Sistem çalışma şeması..... 40
Şekil 5.2	Makro ve Mikro Baz İstasyonları ile oluşturulmuş örnek ortam..... 41
Şekil 5.3	Kümeleme yöntemi çıktısı. 42
Şekil 5.4	Üç boyutlu mesafe tanımı. 45
Şekil 5.5	ANN karar bölgeleri..... 48
Şekil 5.6	k-NN karar bölgeleri. 49
Şekil 5.7	SVM karar bölgeleri..... 50

Şekil 5.8	XGBoost karar bölgeleri.	51
Şekil 5.9	ML modellerinin ortak ağ dilimi karar verme mekanizması	51
Şekil 5.10	Karar çoğunluğu karar bölgeleri.	52
Şekil 5.11	Veri hızını gereksinim seviyesine düşürülme görseli.	54
Şekil 6.1	Benzetim ortamındaki kullanıcılar ve kullanıcıların ağ dilimleri.	55
Şekil 6.2	Hata matrisi.	56
Şekil 6.3	Örnek ROC eğrisi.	58
Şekil 6.4	Hata payı olmayan veri ile yapay sinir ağı ROC eğrisi.	59
Şekil 6.5	Hata payı olmayan veri ile k-En yakın komşu ROC eğrisi.	60
Şekil 6.6	Hata payı olmayan veri ile destek vektör makineleri ROC eğrisi.	61
Şekil 6.7	Hata payı olmayan veri ile XGBoost ROC eğrisi.	62
Şekil 6.8	Hata payı olmayan veri ile karar çoğunluğu ROC eğrisi.	63
Şekil 6.9	Hata paylı veri ile yapay sinir ağı ROC eğrisi.....	65
Şekil 6.10	Hata paylı veri ile k-En yakın komşu ROC eğrisi.....	66
Şekil 6.11	Hata paylı veri ile destek vektör makineleri ROC eğrisi.	67
Şekil 6.12	Hata paylı veri ile XGBoost ROC eğrisi.	68
Şekil 6.13	Hata paylı veri ile karar çoğunluğu ROC eğrisi.....	69
Şekil 6.14	Gerçekleşen ek mekanizmaların istatistik değerleri.	70

KISALTMALAR

3GPP	: 3rd Generation Partnership Project
AR	: Artificial Reality (Yapay Gerçeklik)
CNN	: Convolutional Neural Networks (Evrşimli Sinir Ağları)
DBN	: Deep Belief Networks (Derin İnanç Ağları)
eMBB	: enhanced Mobile Broad Band (Gelişmiş Mobil Şebeke)
G	: Generation (Nesil)
GBR	: Guaranteed Bit Rate (Garantili Bit Hızı)
GS	: Grid Search
k-NN	: k-Nearest Neighbor (k-En Yakın Komşu)
LSTM	: Long Short-Term Memory (Uzun Kısa-Zamanlı Bellek)
MIoT	: Massive Internet of Things (Büyük Nesnelerin İnterneti)
ML	: Machine Learning (Makine Öğrenmesi)
NFV	: Network Function Virtualization (Ağ Fonksiyonlarının Sanallaştırılması)
ANN	: Artificial Neural Network (Yapay Sinir Ağı)
NLOS	: No Line of Sight (Görüş Hattı Olmayan)
URLLC	: Ultra Reliable Low Latency Communications (Çok Düşük Gecikmeli Veri İletişimi)
PL	: Path Loss (Yol Kaybı)
RF	: Random Forest (Rastgele Orman)
SDN	: Software Defined Networking (Yazılım Tanımlı Ağ)
SL	: Shadow Loss (Gölge Kaybı)
SVM	: Support Vector Machines (Destek Vektör Makineleri)
V2X	: Vehicle to Everything (Araçtan Her Şeye)
VR	: Virtual Reality (Yapay Gerçeklik)
QCI	: Quality of Service Class Indicator (Hizmet Kalitesi Sınıf Göstergesi)

1. GİRİŞ

Haberleşme, ilk çağlardan beri insan hayatındaki en önemli olaylardan biri olmuştur. İlk olarak insan vücudu ile haberleşme sağlanıyorken, sonrasında araç ve gereçler kullanılarak haberleşme sağlanmıştır. Bu araç ve gereçler çağlar ilerledikçe ilkel aletlerden hayvanlara, sonrasında yapılan icatlar ile teknolojik aletlere dönüşmüştür. Teknolojik gelişmeler ile, son birkaç on yılda haberleşme sektörü çok ciddi bir gelişim göstermiştir. Çeşitli haberleşme yolları oluşmuş ve insanların kolayca ulaşabileceği imkanlar elde edilmiştir. Son yıllarda hızlı gelişen mobil cihazlar ile haberleşme artık sadece iletişim aracı değil, gelişen uygulamalar ile insan hayatının her kısmında yer alan bir araca dönüşmüştür. Farklı imkanlar ile, mobil cihazların sayısı ve yeni hizmetler ve uygulamaların hayatımıza girme hızı da artmıştır [1].

Artan mobil cihazlar, kullanım şekilleri ve uygulama tiplerinden dolayı beşinci nesil (fifth generation, 5G) haberleşme çok yüksek beklentiler ve olanaklar ile gelmektedir. Farklı uygulama imkanlarından dolayı, kullanıcıların ihtiyacı olan veri hacimlerindeki artış ve farklı teknik gereksinimlere sahip hizmet çeşitleri artış göstermiştir. Bu sebeple, 5G ile mevcut yapıyı geliştirerek, aynı yapı içerisinde farklı hizmetler sunabilen ve kapasitesi daha yüksek olan sistemler oluşturulması düşünülmektedir [2]. 5G ile belirlenen başarımların standartları sayesinde, otonom sistemler, yapay gerçeklik, akıllı fabrikalar gibi çeşitli sektörlerde ilerlemeler gözlenmesi planlanmaktadır.

Son yıllarda yaşanan artışlar ile ihtiyaç duyulan teknik gereksinimlerin artması sebebiyle, haberleşme sağlayıcı şirketler programlanabilir sistem çözümleri geliştirmektedir. Dördüncü nesil (fourth generation, 4G) haberleşme sistemlerinde kullanılan yazılım tanımlı ağ (software defined networking, SDN) ve ağ fonksiyonlarının sanallaştırılması (network function virtualization, NFV) yöntemleri bu amaç için oluşturulmuş ve sistemin daha kontrol edilebilir olmasını sağlamıştır [3]. 5G yapısı oluşturulurken, bu iki sistemden esinlenilmiş ve Ağ Dilimleme yapısı önerilmiştir. Ağ Dilimleme, aynı fiziksel yapı içerisinde, birçok farklı sanal ağ oluşmasını ve bu ağların farklı tipte kullanıcılara hizmet vermesini

sağlayan bir yapı olarak düşünülmüştür. Bu şekilde, sistemlerin daha efektif bir şekilde kullanılması ve yeni hizmetlerin daha kontrol edilebilir ve hızlı bir şekilde devreye sokulması hedeflenmiştir. Bu kontrol edilebilirliğin yeni hizmetlerin pazarlanabilirliğinde önemli rol oynaması planlanmıştır [4].

1.1. Tezin Amacı

Bu tez çalışmasında 5G Ağ Dilimleme için gerçekçi bir benzetim ortamı oluşturulup, çeşitli makine öğrenmesi yöntemleri ile 5G Ağ Dilimi sınıflandırılması yapılması hedeflenmiştir. Bunun yanında, benzetim ortamı içerisinde bulunan kullanıcıların sorunsuz bir hizmet alabilmeleri için farklı sistemler geliştirilmesi hedeflenmiştir. Bu tez çalışması sonlandığında aşağıdaki işlemlerin başarılı bir şekilde gerçekleştirilmesi hedeflenmektedir:

- 3. Nesil Ortaklık Projesi (3rd Generation Partnership Project, 3GPP) standartlarına uygun, farklı kullanım tipleri içeren kullanıcıların oluşturulması.
- Baz istasyonlarının oluşturulup, 3GPP standartlarına göre *Kullanıcı - Baz İstasyonu* ilişkilendirilmesi yapılması.
- İlişkilendirilen kullanıcıların ağ dilimi sınıflandırılmasının yapılması.
- Yoğunluk ile yaşanabilecek baz istasyonu kapasite sorunlarını yönetebilecek bir *Veri Hızı Yönetim Mekanizması* geliştirilmesi.
- Kullanıcı hareketliliğinden oluşabilecek *Kullanıcı - Baz İstasyonu* bağlantılarının değişimi ve baz istasyonları arasında devir işlemlerinin yapılması.

1.2. Tezin Katkıları

Bu tez çalışmasının ana katkıları aşağıdaki gibi özetlenebilmektedir:

- Oluşturulan ağ dilimleme makine öğrenmesi modellerinin gerçek zamanlı bir benzetim ortamında test edilmesi.

- Çeşitli makine öğrenmesi modelleri oluşturulup, modellerin başarılarının karşılaştırılması.
- Benzetim ortamı içerisine baz istasyonu faktörü ekleyip, gerçekçi bir benzetim çalışması olmasının sağlanması.
- Genel sistemin başarılı bir şekilde çalışması için, Veri Hızı Yönetim Mekanizması ve Kullanıcı Devir Mekanizması oluşturulması.

1.3. Tezin Akışı

Bu tez çalışmasının devamının akışı şu şekildedir:

- **Bölüm 2**'de haberleşme sistemlerinin geçmişten günümüze gelişimini ve teknolojinin 5G yapısı ile ağ dilimlemeye nasıl ilerlediği anlatılmaktadır.
- **Bölüm 3**'de 5G ağ dilimlemeye yol açan SDN ve NFV teknolojilerinden ve bu teknolojilerin 5G ağ dilimleme içerisindeki rollerinden bahsedilmektedir.
- **Bölüm 4** bu çalışma içerisinde kullanılan makine öğrenmesi modellerinin çalışma prensiplerini, yapılarını ve avantajlarını içermektedir.
- **Bölüm 5** bu tez kapsamında oluşturulan 5G Ağ Dilimleme sistem modelini, sistem içerisinde kullanılan yöntemleri ve sistem içerisinde bulunan mekanizmaları içermektedir.
- **Bölüm 6**'de oluşturulan sistem modelinin test edildiği benzetim ortamından ve makine öğrenmesi modelleri ile alınan sonuçlardan bahsedilmektedir.
- **Bölüm 7**'de sonuçlar ve tartışma yer almaktadır.

2. HABERLEŐME SİSTEMLERİNİN GELİŐİMİ

Geçtiđimiz birkaç on yıl boyunca, 1G'den 5G'ye nesiller arasında muazzam bir deđişim yaşanmıştır. G, farklı teknikler ve teknolojiler içeren haberleşme standartlarını ifade etmektedir. Haberleşme nesillerinin en başından itibaren sistemlerin en uygun ve en etkili şekilde kullanılması en önemli hedeflerden biri olmuştur. Bu sebeple nesillerin kullandığı teknolojiler, düzenli olarak gelişmiş ve ilerlemiştir [5].

2.1. Birinci Nesil (1G)

1G, kablosuz haberleşmeyi ilk olarak mümkün kılan nesil olmuştur. Analog olarak haberleşme teknolojisini benimsemiştir. Yalnızca sesli olarak iletişim sağlamaktadır. 1G'de, Frekans Bölmeli Çoklu Erişim (frequency division multiple access, FDMA) [6] teknolojisi kullanılmıştır.

2.2. İkinci Nesil (2G)

2G ile analog teknoloji yerine dijital modülasyon kullanılmaya başlanmıştır. 2G ile, sesli iletişimin yanı sıra kısa mesaj ve multimedya mesaj servisleri ile mesaj iletişimi de sağlanmıştır. 2G'de, Zaman Bölmeli Çoklu Erişim (time division multiple access, TDMA) [7] ve Kod Bölmeli Çoklu Erişim (code division multiple access, CDMA) [8] teknolojileri kullanılmıştır.

2.3. Üçüncü Nesil (3G)

3G sistemi, 3GPP organizasyonu tarafından belirlenen standartlar ile oluşturulmuştur. 2G'de kullanılan dijital modülasyon servislerinin daha da geliştiđi bir nesildir. 2G'de sunulan ses ve mesaj iletişimlerinin yanı sıra televizyon, video, internette gezinme gibi imkanlar da sunmuştur. 3G'de, 1G ve 2G'de kullanılan (FDMA, TDMA, CDMA) teknolojilerin tamamı

kullanılmıştır. Bu teknolojilerin birlikte kullanılması 4G’de gelecek yeni teknolojilerin önünü açmıştır.

2.4. Dördüncü Nesil (4G)

4G’ye gelene kadar, çeşitli iletişim imkanları sunulmuştur. Yüksek seviyede gelişim gösteren, iletişim kalitesi, kapasite ve güvenlik gibi özellikler sayesinde, 3G’de sunulan imkanların yüksek kalite ile sunulması sağlanmıştır.

4G sistemlerinde kullanılan Dikey Frekans Bölmeli Division Multiple Access (orthogonal frequency division multiple access, OFDMA) [9], Çoklu Giriş Çoklu Çıkış (multiple-input multiple-output, MIMO) [10] teknolojilerinin yanı sıra, 5G’ye gelirken Ağ Dilimleme teknolojisine imkan sağlayacak olan SDN ve NFV teknolojileri de kullanılmıştır.

2.5. Beşinci Nesil (5G)

4G ile sunulan haberleşme hizmetlerinin kalite artışı gerçekleşmiştir. Bu artış sebebiyle, 4G’den sonra haberleşme cihazları ve uygulamalar hızlıca gelişme göstermeye başlamıştır. Yükselen kalite standartları ve uygulamalar ile birlikte 5G’ye olan beklenti oldukça yüksek olmuştur [11].

5G teknolojisi, 4G’de kullanılmış olan teknolojilerden yararlanıp, onları geliştirmiştir. Bu teknolojiler OFDMA, Büyük MIMO (massive multiple-input multiple-output, m-MIMO) [12] ve Ağ Dilimleme [13] olmuştur. SDN ve NFV teknolojileri, 5G Ağ Dilimleme teknolojisinin oluşturulması konusunda büyük önem göstermiştir.

Haberleşme sistemleri, 1G’den 5G’ye düzenli olarak yükselen bir gelişim göstermiştir. Nesillerin teknik özelliklerini ve kullandığı teknolojileri Tablo 2.1’de görülmektedir.

Tablo 2.1 Haberleşme sistemleri nesillerinin özellikleri.

	1G	2G	3G	4G	5G
Kullanım Zamanı	1980'ler	1990'lar	2000'ler	2010'lar	2020'ler
Teknoloji(ler)	FDMA	CDMA TDMA	CDMA FDMA TDMA	OFDMA MIMO SDN NFV	OFDMA m-MIMO Ağ Dilimleme
İndirme hızı	2.4 kbits/s	64 kbits/s	2 Mbits/s	1 Gbits/s	10 Gbits/s
Gecikme	-	~500 ms	~100 ms	~50 ms	<1 ms

2.6. Literatürde 5G Teknolojisinin Gelişimi

Yazarlar, [14] çalışmasında, önceki nesillerde kullanılmaya başlanan SDN ve NFV teknolojilerinden ve bu teknolojiler kullanılarak geliştirilen 5G Ağ Dilimleme konseptinden bahsetmiştir. Çalışma içerisinde ağ dilimleme konseptleri, mimarileri ve zorluklarına değinilmiştir. Uluslararası standartlar dikkate alınarak değinilen başarımlar, yönetim ve güvenlik konuları ağ dilimlenimin ne şekilde olacağına dair fikir vermektedir.

Yazarlar, [15] çalışmasında 5G teknolojisinin özellikle mobil iletişim alanındaki etkisinden bahsedilmiştir. 1G'den itibaren 5G'ye gelene kadar yaşanan gelişmelerden ve 5G ile birlikte teknolojinin geldiği noktada düşük masraflı ve yüksek kaliteli hizmet verme imkanına değinilmiştir. 5G ile değişecek mobil iletişim sektörünün bilgisayar sektörlerine etki edeceğinden de bahsedilmiştir.

Farklı bir konu ele alınarak, [16] çalışmasında 5G alanında yapılan araştırmalar ve gelişmelere değinilmiştir. 5G yapısının esnekliği, kapasitesi ve bulut teknoloji servis imkanlarına değinilmiştir. Bu imkanların geleceğe ait haberleşme imkanları sunacağı öne sürülmüştür. Çalışmanın genelinde temiz, esnek ve çoğunlukla kullanılacak olan 5G haberleşme standartlarına değinilmiştir.

Bir diđer alıřmada [17] ise 5G ađ yapısının teknik detaylarından bahsedilmiřtir. Ađ yapısının esnekliđine ve gcne, ayrıca bu zelliklere sahip olmasını sađlayan ađ dilimi gibi teknolojilere deđinilmiřtir. Ađ yapısının programlanabilirliđinden ve yapı ierisinde ayrı řekilde kontrol edilebilir zelliklerinden bahsedilmiřtir. Ayrıca, kresel standartlar dikkate alınarak 5G teknolojisinin geliřtirilmesi sırasında uyulabilecek bir yol haritası nerilmiřtir.

Son olarak, [18] alıřmasında 5G teknolojisinin hayata geirilme ařamalarında dikkate alınabilecek zelliklere ve standartlara deđinilmiřtir. Ayrıca, hayata geirilme ařamasında gerekebilecek en iyileme senaryoları ve hayata geirildikten sonra ađ yapısının durumu ile ilgili bilgiler verilmiřtir.

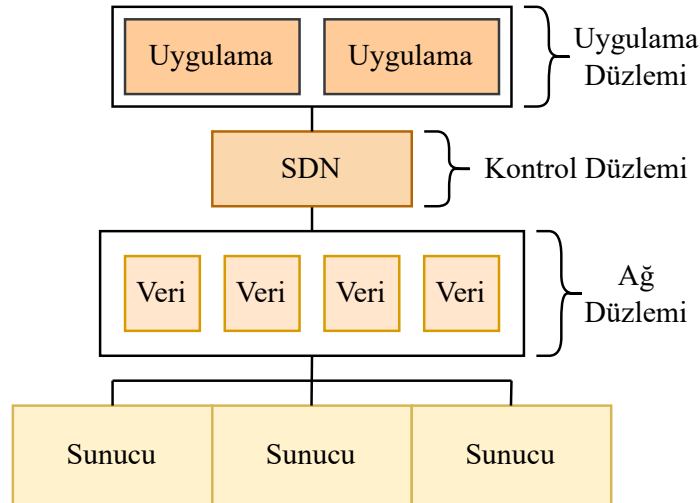
3. 5G AĞ DİLİMLEME

Bu kısımda, 5G Ađ Dilimleme teknolojisini mümkün kılan SDN ve NFV teknolojilerinden ve bu teknolojilerin Ađ Dilimleme içerisinde nasıl kullanıldığına değinilecektir. SDN, NFV ve Ađ Dilimleme aralarındaki benzerlikler ve kullanım avantajlarına değinilecektir.

3.1. Yazılım Tanımlı Ađ

SDN, donanımsal altyapı ile köprü oluşturmak için oluşturulmuş bir teknolojidir. SDN sayesinde, altyapı ile iletişim kurmak, altyapıyı yönlendirmek, altyapıyı programlanabilir bir hale getirmek hedeflenmiştir. SDN ile altyapı yapılandırmasının daha verimli olması ve geleneksel sistem yerine bulut sistemine yaklaşılması sağlanmıştır [19].

SDN ile altyapının programlanabilir olması sayesinde, yüksek seviyede kontrol edilebilir ve hızlı bir yapı sağlanmıştır. Ayrıca kaynakların çeşitli kullanılıp, özelleştirilmiş ađ yapıları oluşturulmasını mümkün kılmıştır. Ađ yapısının tamamının kontrol edilebilir olması sebebiyle bütünsel bir görüntü sağlayıp, daha güvenli bir yapı sunmaktadır [20]. SDN teknolojisinin yapısı Şekil 3.1'de görülmektedir.



Şekil 3.1 Yazılım Tanımlı Ađ yapısı.

3.2. Ağ Fonksiyonlarının Sanallaştırılması

NFV, altyapıda kullanılan fiziksel donanımlar yerine sanal makineler kullanılması teknolojisidir. NFV sayesinde, fiziksel ağ hizmetleri sistemleri, sanallaştırılmış servislere dönüştürülmüştür [21]. Bu değişim ile yeni servisler çok daha dinamik şekilde, fiziksel bir değişim yapmadan uygulanabilmiştir.

NFV ile altyapının fiziksellikten sanallığa dönüştürülmesi sağlanmıştır. Bu sayede, sadece gerekli seviyede sanal sistemlerin oluşturulmasıyla maliyet azaltılmıştır. Bu durumda, gerektiğinde sistemin genişletilmesi ya da daraltılması daha kolay olmuştur. Ayrıca, aynı operasyonlar için daha az cihaz kullanılması sağlanmıştır.

SDN ve NFV teknolojilerinin birbirlerine bağımlılıkları bulunmamaktadır. Ancak, birlikte kullanıldığında yüksek avantajlar getirmektedirler. NFV ile oluşturulan yapıların, SDN ile yürütülmesi birçok açıdan avantajlı bir yapı oluşturmaktadır [22].

3.3. Ağ Dilimleme

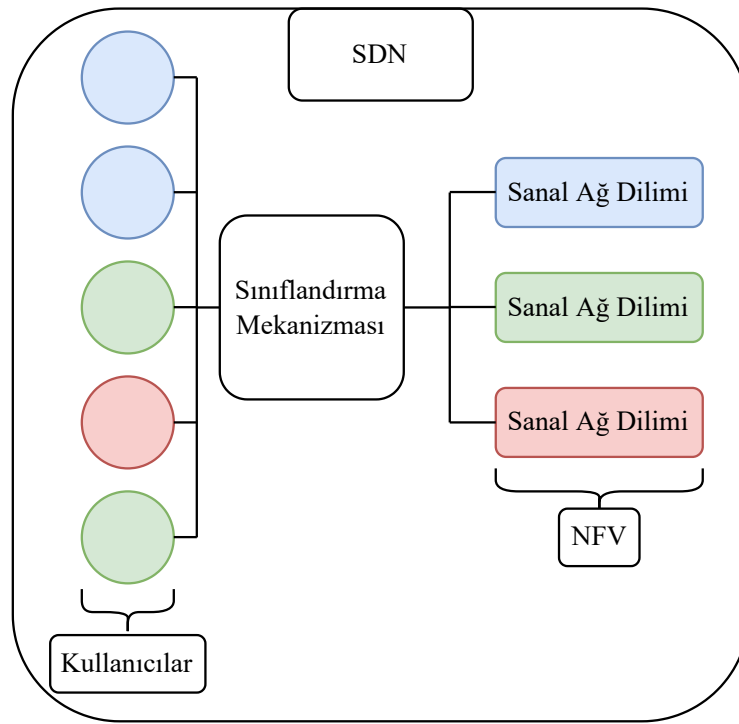
Ağ dilimleme, aynı fiziksel yapı içerisinde farklı amaçlar için kullanılabilen yapıların oluşturulmasını sağlayan bir teknolojidir. 5G ile değişecek olan haberleşme servislerinin sağlanabilmesi konusunda önemli bir rol üstlenmektedir.

SDN teknolojisi, ağ dilimleme için temel unsurlardan biridir. SDN ile, kullanıcıların ihtiyaçlarına uygun uygulamaların ve gereksinimleri bulunduran sistemin kontrolü sağlanmaktadır. NFV ise başka bir temel unsurdur. NFV sayesinde, ağ dilimleri içerisinde oluşan kullanıcı odaklı yapıların oluşturulması, yönetiminin sağlanması ve sanal yapıların hayat döngüsünün kontrol edilmesi sağlanmaktadır. Farklı ağ dilimli yapısından dolayı, aynı fiziksel yapı içerisinde farklı gereksinim ihtiyacı olan kullanıcıların ihtiyacının karşılanabilmesinin yanı sıra, dilimli yapı ile daha güvenli bir ağ yapısı sağlanmaktadır [23].

3GPP dokümanlarında, oluşturulacak Ağ Dilimleme yapısında, dört farklı ağ dilimi yer alması sunulmuştur. Bunlar *gelişmiş mobil şebeke (enhanced mobile broadband, eMBB)*,

çok düşük gecikmeli veri iletişimi (*ultra reliable low latency communications, URLLC*), büyük nesnelerin interneti (*massive internet of things, MIoT*) ve araçtan her şeye (*vehicle to everything, V2X*) olarak belirlenmiştir. Sunulan dilimlerin her biri, isimlerinde de belirtildiği gibi farklı amaçlara hizmet etmektedir ve Ağ Dilimleme sayesinde aynı fiziksel yapı içerisinde bulunabilmektedir.

Ağ Dilimleme teknolojisinin yapısı ve içerisinde bulundurduğu teknolojiler Şekil 3.2’de görülmektedir.

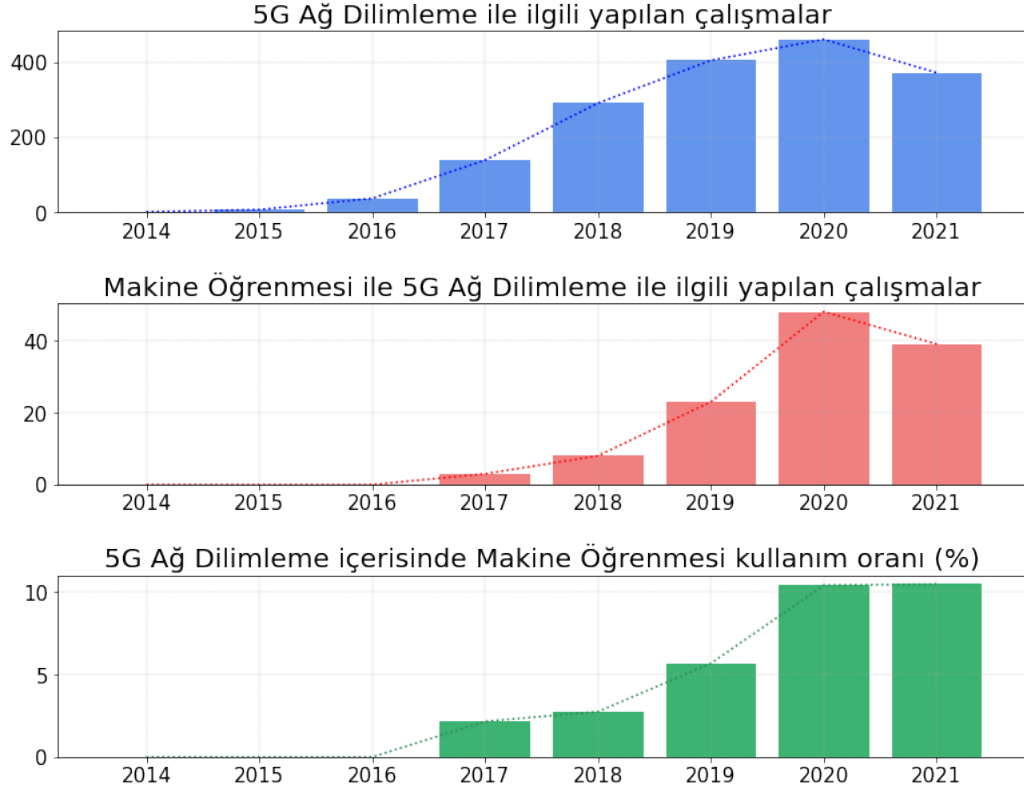


Şekil 3.2 Ağ Dilimleme yapısı.

3.4. Literatürde Ağ Dilimleme Çalışmaları

5G Ağ Dilimleme ile ilk olarak 2014 yılında *Web of Science* üzerinden taranabilen bir çalışma yapılmıştır. Yapılan ilk çalışmalarda makine öğrenmesi yöntemleri bulunmaz iken, sonrasında yıllar ilerledikçe 5G Ağ Dilimleme içerisinde kullanılan makine öğrenmesi yöntemleri artış göstermiştir. 2021 yılı dahil olmak üzere, 5G Ağ Dilimleme ile ilgili 1574 çalışma bulunurken, bu çalışmaların 118 tanesinde makine öğrenmesi kullanılmıştır. 118

makine öğrenmesi kullanılan çalışmanın 110 tanesi son üç yıl içerisinde yer almaktadır. Bahsedilen bilgilerin görselleştirildiği grafik Şekil 3.3'te görülmektedir.



Şekil 3.3 5G Ağ Dilimleme için *Web of Science* tarama sonuçları.

Çalışmalarda makine öğrenmesi farklı amaçlar için kullanılmıştır. Kullanıcı ağ dilimi sınıflandırması en yoğun kullanım sebebi olmaktadır. Ağ dilimi sınıflandırmasının yanı sıra, ağ dilimi rezervasyonları, ağ dilimi kaynak durumu yönetimlerine de çalışmalarda yer verilmiştir. Bu çalışmalar içerisinde, yapılan tez çalışmasına etkisi olanlar da bulunmuştur.

Yazarlar, [24] çalışmasında makine öğrenmesi yöntemiyle ağ dilimleme gerçekleştirilmiştir. Makine öğrenmesi algoritmalarından, rastgele orman ve derin sinir ağı modelleri geliştirilmiştir. Çalışmada, 65 bin çeşitli kullanıcı kullanılmıştır, bu kullanıcılar 3GPP kurallarına göre oluşturulmuştur. Ayrıca ağ arızası durumunda kullanılmak üzere, ana ağ dilimlerinin yanı sıra ek bir *yönetici* dilimi kullanıp hizmete devam edilmesini sağlamışlardır. Çalışma sonucunda, arıza durumunda sistemin başarılı çalıştığı görülmüştür. Ayrıca, kullanıcıların ağ dilimleri %95 oranında başarılı bir şekilde gerçekleştirilmiştir. Yazarlar,

gelecek çalışmalarında, oluşturulan modelin genel bir 5G ekosisteminde test edilmesini ve sistemin daha başarılı çalışmasını sağlayacak ek mekanizmaların geliştirilmesini düşünmüşlerdir.

Farklı bir yaklaşım olarak, [25] çalışmasında yazarlar derin pekiştirmeli öğrenme ile düzenli aralıklar ile kullanılmayan ağ kısımlarını minimum gereksinimleri karşılayabilecek halde rezerve etmişlerdir. Çalışmalarında derin pekiştirmeli öğrenme modeli kullanıp, bu model ile rezerve edilen kısımları daha iyi yönetmeyi hedeflemişlerdir. Geliştirilen model %90 oranında başarı elde etmiştir. Pekiştirmeli öğrenme bir makine öğrenme tipi olup, elde edilmek istenen sonuca yönelik ödül-ceza mantığı ile çalışan bir öğrenme tipidir. Modelin amacı ödülleri en yüksek seviyeye çıkartmaktır ve bunun sayesinde hedeflenen noktaya ulaşmış olmaktadır.

Bir diğer çalışmada ise [26] yazarlar derin öğrenme modeli oluşturarak, kullanıcı gereksinimleri ve ağ durumunu göz önünde bulundurarak ağ dilimleme gerçekleştirilmiştir. Kullanıcı verisi, zamana bağımlı ve zamandan bağımsız değişkenler olarak ikiye ayrılmıştır. Değişkenler toplam iletilen ve alınan veri boyutları, toplam meşgul ve boş zaman gibi özelliklerden oluşmaktadır. Çalışmalarında uzun kısa-süreli bellek modeli kullanıp, modelin yapısını ve öğrenme sürecindeki yaklaşımlarını detaylı bir şekilde açıklamışlardır. Geliştirilen model ağ dilimleme işleminde %75 oranında başarı elde etmiştir. Uzun kısa-süreli bellek, derin öğrenme modellerinden biri olmaktadır. İçerisinde geri bildirim sistemi bulunmaktadır ve bağlantılı durumlarda kullanılmaktadırlar.

Yazarlar, [27] çalışmasında sinir ağı ve derin inanç ağı modelleri oluşturulup, toplu karar modeli kullanılarak ağ dilimi gerçekleştirilmiştir. İki model aynı karara ulaşmadığında modellerin ağırlıklarını değiştirip tekrar karar vermeye yönelten bir mekanizma oluşturulmuştur. Ayrıca kullanıcılar kullanıcı cihaz tipi, kullanım süresi, teknik parametreler gibi değişkenler ile 3GPP standartlarına göre oluşturulmuştur. Çalışmada kullanılan değişkenlerin sayısal aralıkları çalışmada detaylı şekilde belirtilmiştir. Geliştirilen model, farklı senaryolarda %93 oranında başarı sağlamıştır. Derin inanç ağı, derin sinir ağı

yapısına benzeyen, içerisinde gizli katmanlar bulunduran ve katmanlar ve düğümler arasında kendine özgü bağlantılar içeren bir yapıdır.

Yazarlar, [28] çalışmasında, otonom araçlara yoğunlaşmış ve onlar için ağ dilimleme gerçekleştirmiştir. Araçlar arası iletişim, paket gecikmesi, paket kaybı ve veri hızı gibi değişkenler kullanılmıştır. Araçların uzaysal bilgileri görüntü işleme yapılarak araç tipinin belirlenmesi sağlanmıştır, bunun için görüntü işleme derin öğrenme modeli geliştirilmiştir. Sonrasında, araçların teknik gereksinimlerinin tahmin edilmesi için pekiştirmeli uzun kısa-sürelili hafıza modeli oluşturulmuştur. Yazarlar, sonuçları görseller ile göstermiştir. Geliştirilen modelin, gerçeğe yakın sonuçlar elde ettiği görülmüştür.

Son olarak, [29] çalışmasında, yeni teknolojik gelişmeler ile farklı ihtiyaçlara yönelik hizmetleri sağlayabilmek için haberleşme ağlarının yazılım tanımlı olması ihtiyacına değinilmiştir. Bu sebeple, kullanıcıların kullandığı uygulama tipini tanımlayan bir derin öğrenme modeli geliştirilmiştir. Çalışma içerisinde farklı uygulama tiplerinin veri hacimleri, kullanım süreleri ve veri hızlarına odaklanılmıştır. Derin sinir ağı modelinin grafik kartı kullanarak eğitilmesi sonucunda, 39 gün süren testler sonucunda uygulama tanımlama konusunda %93 oranında başarı elde etmişlerdir.

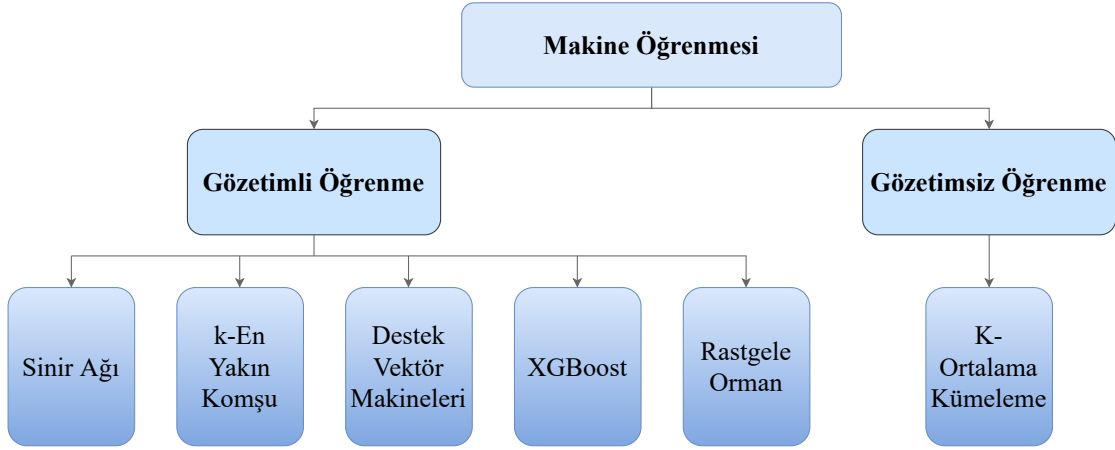
4. MAKİNE ÖĞRENMESİ MODELLERİ

Bu bölümde, sistem içerisinde kullanılmış olan makine öğrenmesi (machine learning, ML) metotları, metotların temel kavramları, çalışma prensipleri ve sınıflandırma için kullanılacak verinin bu metotlar için hazırlanması ifade edilecektir.

Literatürde ML yöntemleri, kullanım amaçlarına uygun değişkenler ile oluşturulup bir sınıflandırma ya da tahmin yapmak için kullanılmaktadırlar. Bu yöntemlerin seçimi ve kullanımı elde bulunan veri durumuna ve hedeflenen çıktıya göre değişiklik göstermektedir. ML gözetimli öğrenme, gözetimsiz öğrenme ve pekiştirmeli öğrenme olarak kendi içerisinde üç kategoriye bölünmektedir. Bu çalışmada gözetimli ve gözetimsiz öğrenme modellerine başvurulmuştur.

Gözetimli öğrenme, öğrenme modellerinde kullanılan verinin etiketli olduğu öğrenme tekniğidir. Gözetimli öğrenme tekniğinde, modeller çok değişkenli eğitim verisini öğrenirken, hangi sınıfa bağlı olduğunu bilerek öğrenme yapmaktadırlar. Gözetimsiz öğrenme ise öğrenme modellerinin etiketsiz veriler kullanarak öğrenme yaptığı tekniktir. Bu teknikte modeller, verileri analiz ederek içerisindeki gizli yapıları matematiksel bağlamlara oturtarak bulmaya çalışırlar. Literatürde veri içerisindeki objelerin hangi sınıfa bağlı olduğunu belirten belirteç *etiket* olarak adlandırılmaktadır.

Bu tez çalışmasında; gözetimli öğrenme tekniklerinden yapay sinir ağı (artificial neural network, ANN), k-en yakın komşu (k-nearest neighbor, k-NN), destek vektör makineleri (support vector machines, SVM), xgboost (XGB) ve rastgele orman (random forest, RF) kullanılmıştır. Ayrıca gözetimsiz öğrenme tekniği olarak da k-ortalama kümelemesi yöntemine başvurulmuştur. Çalışmada kullanılan modellerden k-NN, SVM, RF ve k-ortalama kümelemesi teknikleri için açık kaynak ML kütüphanesi olan Scikit-learn'e [30], ANN modelinin oluşturulması için başka bir açık kaynak kütüphane olan Keras'a [31] başvurulmuştur. Şekil 4.1 üzerinde çalışmada kullanılan ML yöntemlerinin kategorize edilmiş hali görülmektedir.



Şekil 4.1 Çalışmada kullanılan makine öğrenmesi yöntemleri

Ağ dilimi ve kullanım tipi sınıflandırmaları için kullanılacak ML modelleri, oluşturulan sentetik veri ile eğitilmektedir. Modellerin başarıları ve verimliliklerinde kullanılmakta olan verinin değişkenleri önem göstermektedir [32]. Bu sebeple ML modelleri oluşturulmadan önce veri ön işleme gerçekleştirilmiştir.

Veri modeller için hazır hale geldikten sonra ML modelleri oluşturulmuştur. Modeller oluşturulurken Scikit-learn *Grid Search (GS)* fonksiyonu, *çapraz doğrulama* kullanılmıştır. GS, ML modellerinin yapısını belirleyen özellikler seçilirken kullanılan bir fonksiyondur. Özellikleri belirlemek için, modelin sahip olabileceği potansiyel özelliklerin kombinasyonlarını veri üzerinde deneyip en başarılı kombinasyonları çıktılmaktadır [33]. Çapraz doğrulama, eğitim verisinin belirli sayıda alt kümelere bölünüp, her bir kümenin sırayla onaylama verisi olarak kullanılması işlemidir. Bu sayede, verinin genelindeki farklı örnekler ile model başarımının değerlendirilmesi sağlanmaktadır [34]. Bu çalışmada GS ve çapraz doğrulama birlikte kullanılıp, modeller için en doğru özelliklerin belirlenmesi sağlanmıştır.

4.1. Veri Ön İşleme

Veri ön işleme, ham verileri modeller için uygun, kullanışlı ve anlaşılabilir hale dönüştürmek için kullanılan bir tekniktir. Veri ön işleme üç ana bölüme ayrılmaktadır. Bu kategoriler veri

temizleme, veri dönüştürme ve veri azaltma işlemleridir [35].

Veri temizleme, genellikle gerçek veriler ile çalışılırken başvurulan bir tekniktir. Bunun sebebi gerçek veriler içerisinde eksiklikler ya da bozukluklar bulunmasıdır [36]. Veri temizleme işlemi ile eksik ve bozuk verilerin kurtulması ya da yerlerine yenilerinin konulması amaçlanmaktadır. Veri dönüştürme, verinin değişkenlerinin modellere hazırlanması sürecidir. Bu sürecin içerisinde veriyi normalize etmek ve veri değişkenleri seçimi yapılması gibi adımlar yer almaktadır [37]. Veri azaltma işlemi ise çok büyük hacimli veriler ile çalışılırken kullanılmaktadır. Yüksek hacimli verilerin, veri analizi ve veri üzerinde yapılacak işlemlerin verimini düşürmesinden dolayı bu yola başvurulmaktadır [38].

Bu çalışmada kullanılmış olan veri sentetik olduğundan dolayı veri temizleme ve veri azaltma işlemlerine gerek duyulmamaktadır. Çalışma içerisinde farklı veri dönüştürme teknikleri kullanılmıştır. Kullanılan teknikler veri normalizasyonu ve farklı kategorize etme metotları olmaktadır.

4.1.1. Normalizasyon (Düzgeleme)

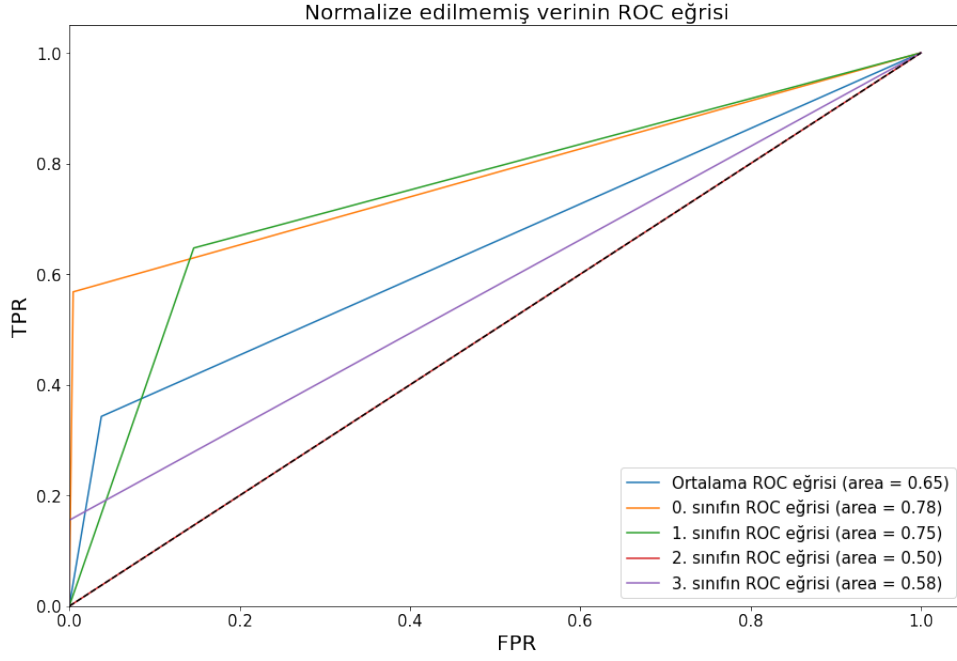
Normalizasyon işlemi verinin değişkenleri birbirlerinden matematiksel olarak farklı aralıklara sahip olduğunda, farklı ölçeklendirildiğinde ve çok farklı olduğu zaman uygulanmaktadır [39].

Bu çalışmada kullanılan *paket kaybı oranı*, *veri hızı* gibi değişkenlerinin birbirlerinden farklı matematiksel aralıklarda yer alması, veri için normalizasyon işlemini gerekli kılmaktadır. Veri üzerinde normalizasyon uygulanırken kullanılan formül Denklem 1'de görülmektedir.

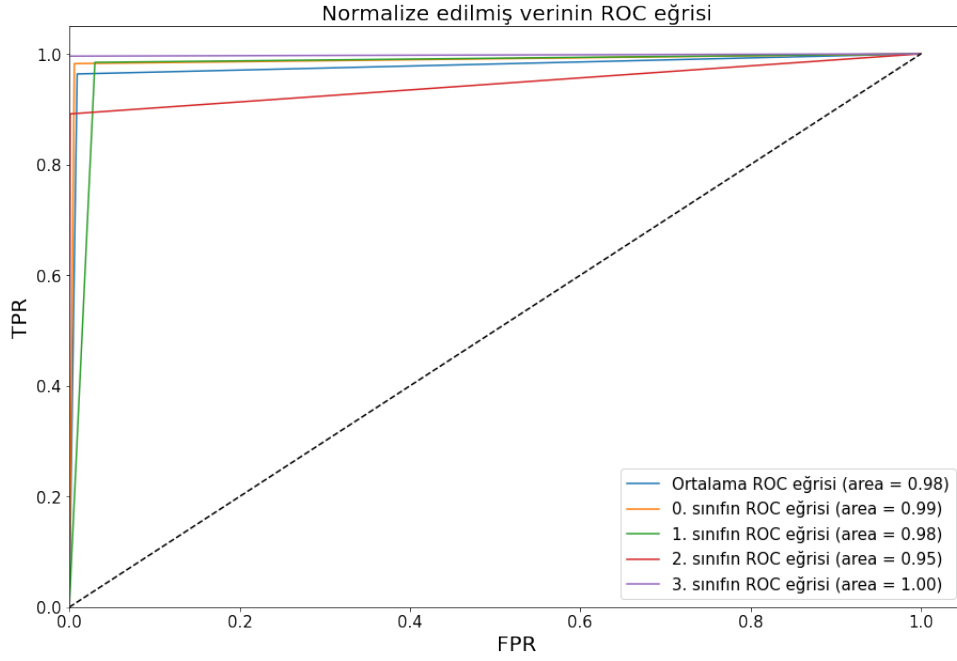
$$x_{\text{norm}} = \frac{x - x_{\text{min}}}{x_{\text{max}} - x_{\text{min}}}, \quad (1)$$

Denklem 1'de x kullanılacak ham veriyi, x_{norm} normalize edilmiş veriyi ve x_{min} ve x_{max} ise ham veri kümesindeki sırasıyla en küçük ve en büyük değerleri göstermektedir.

Normalizasyonun sınıflandırma başarımı üzerindeki etkisi Şekil 4.2 ve Şekil 4.3 incelendiğinde görülmektedir.



Şekil 4.2 Normalize edilmemiş veri ile sınıflandırma.



Şekil 4.3 Normalize edilmiş veri ile sınıflandırma.

Şekil 4.2 ve Şekil 4.3 üzerinde görülen grafiklerin değerleri modelin her bir sınıf için başarısını belirlemektedir.

ML modellerinin veri üzerinde çalışabilmesi için modele verilecek verilerin sayısal hale getirilmesi gerekmektedir. Veri içerisinde hem girdi hem de çıktı verileri arasında sayısal olmayan değişkenler bulunmaktadır. Literatüre bakıldığında değişken içerisindeki sınıf sayısı çok fazla olmayan senaryolarda tek-sıcak kodlama kullanıldığı görülmektedir [40]. Ancak bu çalışmada kullanılan farklı ML modellerinin farklı teknikleri ve özellikleri olduğundan dolayı sadece tek-sıcak kodlama yeterli olmamaktadır. Çok girişli ve çok çıkışlı modellerde tek-sıcak kodlama (4.1.2. Kısım) kullanılırken, çok girişli tek çıkışlı modeller için etiket kodlama (4.1.3. Kısım) kullanılmaktadır.

4.1.2. Tek-sıcak kodlama

Tek-sıcak kodlama, verinin değişkenlerinin ikili sistem ile temsil edilmesi anlamına gelmektedir. Verinin değişkenlerinin içerisindeki her bir sınıf için bir değişken atanıp, o değişkenin ikili sistem üzerinde temsil edilmesi işlemidir. Tek-sıcak kodlama ismini, değişkenleri etiketlerken kullandığı teknikten almaktadır. Bu teknik ile değişkenin temsil ettiği sınıf 1 değerini alırken diğer tüm değişkenler 0 değerini aldığından tek-sıcak kodlama denmektedir. Normalizasyon sonrasında tek-sıcak kodlama ile veri ön işleme bitmiş veriler Tablo 4.1'de görülmektedir.

Tablo 4.1 Normalizasyon ve tek-sıcak kodlama sonrası veri.

Hareketlilik	...	Paket Kaybı Oranı	Veri Hızı	MIoT	URLLC	V2X	eMBBB
0.021795	...	0.006078	0.000062	1	0	0	0
0.807625	...	0.000075	0.058610	0	0	1	0
0.004817	...	0.000084	0.000135	0	1	0	0

4.1.3. Etiket kodlama

Etiket kodlama da tek-sıcak kodlama gibi deęişkenleri sayısallaştırma işlemidir. Sayısallaştırma işlemi belirlenen etiketlere göre yapılmaktadır. Verideki deęişken sayısının yüksek olduęu senaryolar ve tek çıkışlı ML modelleri kullanımında kullanılmaktadır. Etiket kodlamada, veri deęişkenleri yerine belirlenen sayısal etiketler atanmakta ve sonrasında modellerin eğitiminde kullanılmaktadır. Bu çalışmada etiket kodlama uygulanan aę dilimleri ve etiketleri Tablo 4.2’de görölmektedir. Normalizasyon sonrasında etiket kodlama ile veri ön işleme bitmiş veriler Tablo 4.3’te görölmektedir.

Tablo 4.2 Aę dilimlerine karşılık gelen etiketler.

Aę dilimi	Etiket numarası
MIoT	0
URLLC	1
V2X	2
eMBB	3

Tablo 4.3 Normalizasyon ve etiket kodlama sonrası veri.

Hareketlilik	...	Paket Kaybı Oranı	Veri Hızı	Aę Dilimi
0.021795	...	0.006078	0.000062	0
0.807625	...	0.000075	0.058610	2
0.004817	...	0.000084	0.000135	1

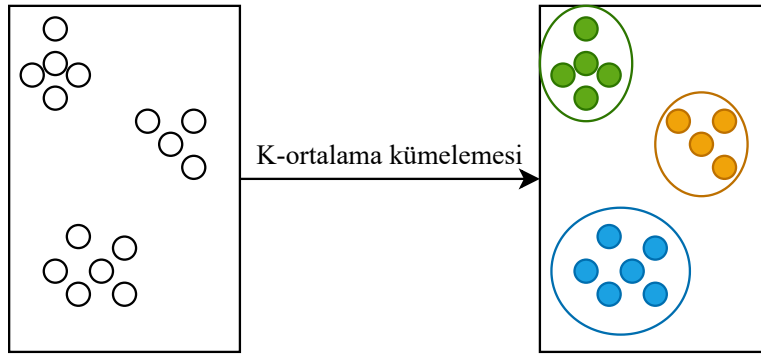
4.2. K-Ortalama Kümelemesi

K-ortalama kümelemesi en temel gözetimsiz öğrenme algoritmalarından biridir ve kümeleme problemlerindeki başarısı ile bilinmektedir [41]. Algoritmanın ana amacı, tüm verileri belirtilen K adet kümeye bölmektir. Sonuç olarak K adet küme merkezi ve merkezlere uzaklıklarına göre kümelendirilmiş veriler elde edilmektedir. Bu çalışmada kullanılmış olan Scikit-learn k-ortalama kümelemesi formülü Denklem 2’de görölmektedir.

$$J = \sum_{i=1}^k \sum_{j=1}^n \|\mathbf{x}(i) - \mathbf{v}(j)\|^2 \quad (2)$$

Denklem 2’de görülen k ve n sırasıyla kümelerin merkez sayısını ve veri noktası sayısını, $\mathbf{x}(i)$ ve $\mathbf{v}(j)$ sırasıyla veri noktasını ve küme merkezini belirtmektedirler. Çıktı J ise küçültülmek istenen kümeleme işlevi olmaktadır.

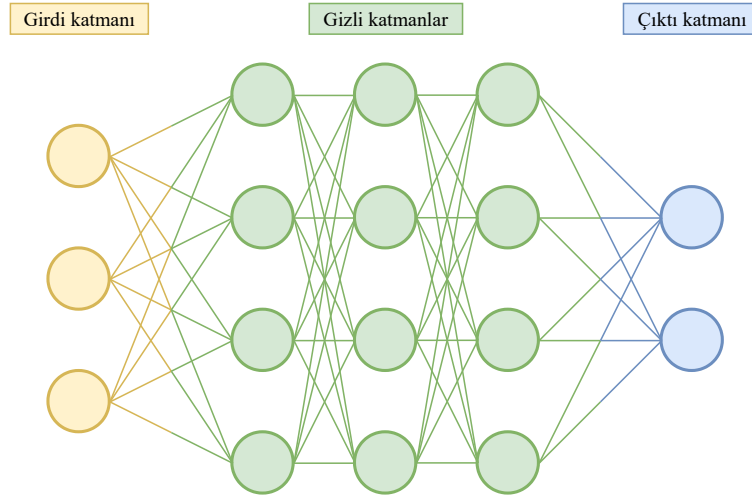
K-ortalama kümeleme modelinin çalışma mekanizması sonrasında oluşan çıktı örneği Şekil 4.4’te görülmektedir.



Şekil 4.4 K-ortalama kümeleme örneği.

4.3. Yapay Sinir Ağı

ANN gözetimli bir öğrenme algoritmasıdır. Makine öğrenmesinin içerisinde ve derin öğrenme algoritmalarının kalbinde yer almaktadır. Yapay sinir ağlarının çalışma prensipleri ve katmanlı yapılarından dolayı ismi insan beyninden esinlenmiştir. ANN, yapısı bir girdi katmanı, bir ya da daha fazla gizli ara katmanları ve bir çıktı katmanı içeren düğüm katmanlarından oluşmaktadır. Her bir katman birbiriyle bağlantılı olduğundan ve katmanların çalışma prensipleri belirlenebildiğinden dolayı ANN modelleri çok amaçlı ve esnek olarak kullanılabilirler [42]. Bahsedilen katmanlı ve bağlantılı yapı Şekil 4.5’te görülmektedir.



Şekil 4.5 Yapay sinir ağı yapısı.

ANN yapısı oluşturulurken, girdi katmanındaki düğüm sayılarını kullanılan verinin değişken sayısı belirlemektedir. Aynı şekilde çıktı katmanındaki düğüm sayısını verinin kaç sınıflı olduğu belirlemektedir. Örneğin Şekil 4.5'te görülen ANN modeli üç değişkenli girdi, iki sınıflı çıktı verisi için oluşturulmuş bir model olmaktadır. ANN, esnek ve kontrol edilebilir yapısı sebebiyle farklı veri tiplerinde kullanılabilir. Bu sebeple sınıflandırma görevlerinde tercih edilmektedir [43].

ANN modellerinin eğitimi esnasında iki farklı hareket gerçekleşmektedir. Bunlar ileri ve geri yayılım hareketleridir. İleri hareket, girdi katmanından başlayarak düğümler ile ilerleyerek çıktı katmanına kadar olan harekettir. Düğümlerin aktivasyon fonksiyonları ile değişen değer çıktı katmanında hesaplanıp hata payı belirlenmektedir. Sonrasında geri yayılım hareketi ile düğümlerdeki değerler güncellenip bu hata payı azaltılmaya çalışılmaktadır. Geri yayılım, hata payını azaltırken gradyan iniş tekniğini kullanmaktadır [44].

Bu çalışmada ANN modeli oluşturulurken Keras kullanılmıştır. ANN modeli içerisinde bulunan aktivasyon fonksiyonları ve katmanlar, Keras kütüphanesinden faydalanılarak seçilmiş ve uygulanmıştır.

4.3.1. Aktivasyon Fonksiyonları

Aktivasyon fonksiyonu ANN içerisindeki düğümlerde yer almaktadır ve öğrenme safhasında iken, düğümlerin aktif olup olmayacağını belirlemektedir. Aktivasyon fonksiyonu ANN modellerinin doğrusal olmayan problemleri çözebilmesini de sağlamaktadır. Geri yayılım hareketi yapılırken kullanılan gradyan iniş tekniği kullanıldığından, aktivasyon fonksiyonunun türevlenebilir olması gerekmektedir. Her fonksiyonun özelliği kendine has olduğundan, aktivasyon fonksiyonu seçimi ve kullanımı ANN modellerinin başarısında ve etkinliğinde önemli rol oynamaktadır [45]. Bu kısımda Keras kütüphanesindeki temel aktivasyon fonksiyonu çeşitleri incelenecektir.

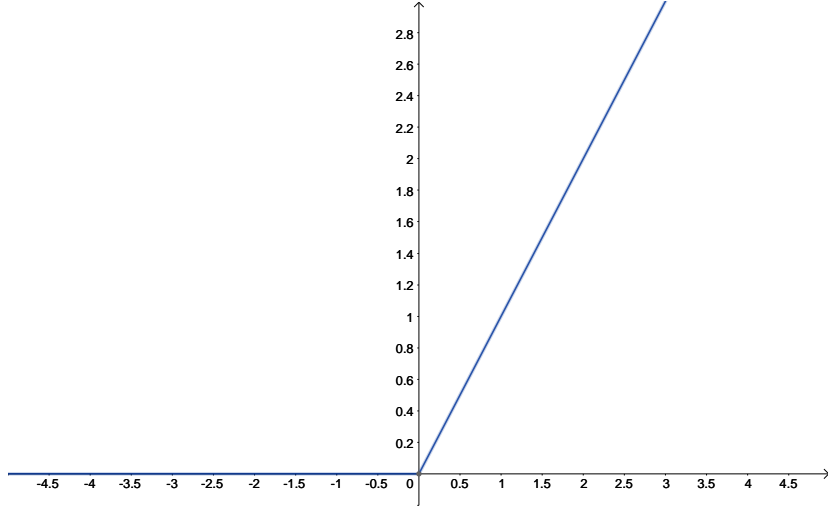
4.3.1.1. Doğrutulmuş Doğrusal Birim (Rectified Linear Unit - ReLU) Fonksiyonu

ReLU doğrusal olmayan ve türevlenebilir bir fonksiyondur. Yapısı gereği tüm düğümleri aynı anda aktive etmemektedir. Denklem 3'te ve Şekil 4.6'da görüldüğü gibi, düğüme gelen değer sıfırın altında ise o düğüm aktive olmamaktadır. Tüm düğümler yerine belli düğümleri aktifleştirdiğinden ve basit bir matematiksel denkleme sahip olduğundan dolayı, ReLU hesaplama ve kapasite açısından verimli bir aktivasyon fonksiyonudur [46]. ReLU satürasyona uğramadığından dolayı gradyan iniş hesaplamaları hızlı yapılmaktadır. Ancak, geri yayılma hareketi esnasında aktifleşmemiş düğümlerin değerleri güncellenememektedir.

$$f_{\text{ReLU}}(x) = \max(0, x) \quad (3)$$

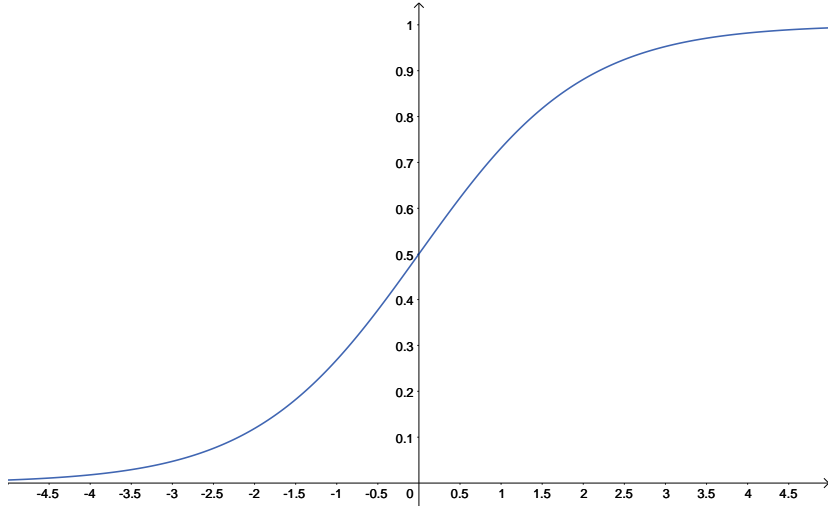
4.3.1.2. Sigmoid Fonksiyonu

Sigmoid doğrusal olmayan ve türevlenebilir bir fonksiyondur. Fonksiyon girdi olarak aldığı değeri 0 ile 1 arasında bir değere yerleştirir. Değer büyüdükçe 1'e yaklaşırken, küçüldükçe 0'a yaklaşmaktadır. Sigmoid fonksiyonunun değerleri yoğun olarak $(-3, 3)$ değerleri arasında dağılım göstermektedir, bu değerlerin dışında türev oldukça küçüktür. Sigmoid fonksiyon formülü ve yapısı Denklem 4'te ve Şekil 4.7'de görülmektedir.



Şekil 4.6 Doğrutulmuş doğrusal birim fonksiyonu.

$$f_{\text{Sigmoid}}(x) = \frac{1}{1 + \exp(-x)} \quad (4)$$



Şekil 4.7 Sigmoid fonksiyonu.

Sigmoid fonksiyonu 0 ile 1 arasında değer aldığından dolayı ikili sınıflandırma durumlarında başarılı çalışmaktadır. 0'a yakın değerlerde bir sınıf, 1'e yakın değerlerde diğer sınıf şeklinde belirtilerek sınıflandırma yapılabilmektedir.

4.3.1.3. Softmax Fonksiyonu

Softmax doğrusal olmayan, sigmoid fonksiyonu ile benzerlikler taşıyan ve türevlenebilen bir fonksiyondur. Çoklu sınıflandırma durumlarında tercih edilmekte ve başarılı olmaktadır [47]. Softmax fonksiyonu girdi olarak aldığı k adet gerçek değerler barındıran vektörü, toplamları 1 olan k adet sayıya dönüştürmektedir. Böylece vektör değerleri olasılık olarak yorumlanabilmektedir. Küçük olan girdi değerleri fonksiyon tarafından düşük olasılıklara, yüksek olan girdi değerleri ise yüksek olasılıklara dönüşmektedir. Softmax fonksiyonu için çoklu sigmoid benzetmesi yapılabilmektedir. Bunun sebebi, vektör içerisindeki her bir değer için sigmoid fonksiyonunun çıkarılmasıdır. Fonksiyonun formülü Denklem 5'te görülmektedir. Denkleminde yer alan \mathbf{z} girdi vektörünü, z_i ve z_j girdi vektörünün değerlerini ve k ise sınıf sayısını belirtmektedir.

$$f_{\text{Softmax}}(\mathbf{z})_i = \frac{\exp(z_i)}{\sum_{j=1}^k \exp(z_j)} \quad (5)$$

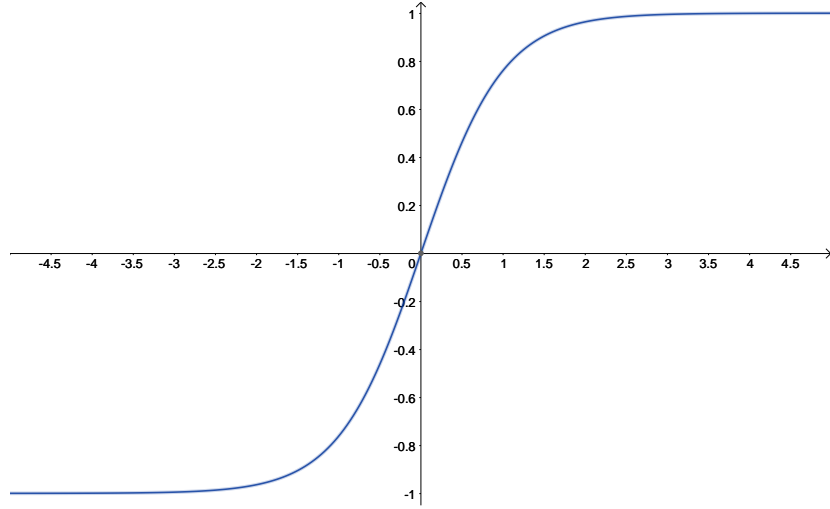
4.3.1.4. Tanh (Hiperbolik Tanjant) Fonksiyonu

Hiperbolik tanjant fonksiyonu doğrusal olmayan ve türevlenebilen bir fonksiyondur. Sigmoid ile çok büyük benzerlikler göstermektedir. Fonksiyon girdi olarak aldığı değeri -1 ile 1 arasında bir değere yerleştirir. Bu sebeple sigmoid fonksiyonundan farklı olarak simetriktir. Şekil 4.8'de görüldüğü yapılarından dolayı, verileri merkezi 0 noktası çevresinde dağıtabilmektedir.

$$f_{\text{tanh}}(x) = \frac{2}{1 + \exp(-2x)} - 1 \quad (6)$$

4.3.2. Katmanlar

ANN yapısında katman, belli bir derinlikte bulunan ve birlikte çalışan düğümler topluluğuna denmektedir. Dikey olarak toplanmış düğümler topluluğu bir katman olmaktadır.



Şekil 4.8 Hiperbolik tanjant fonksiyonu.

Katmanlar genel anlamda girdi katmanı, gizli katman(lar) ve çıktı katmanı olarak üç tipe ayrılmaktadır. Girdi katmanı başlangıç katmanı olmaktadır ve burada bulunan veriler ham haldedir. Ara katman(lar) gerçek operasyonların gerçekleştiği, farklı özelliklere sahip katmanların bulunabildiği katmanlardır. Çıktı katmanı ise basit operasyonların gerçekleştiği son katmandır. ANN modellerinin katman derinliği hesaplanırken girdi katmanı hesaba katılmamaktadır, ara katman(lar) ve çıktı katmanı derinliği belirlemektedir.

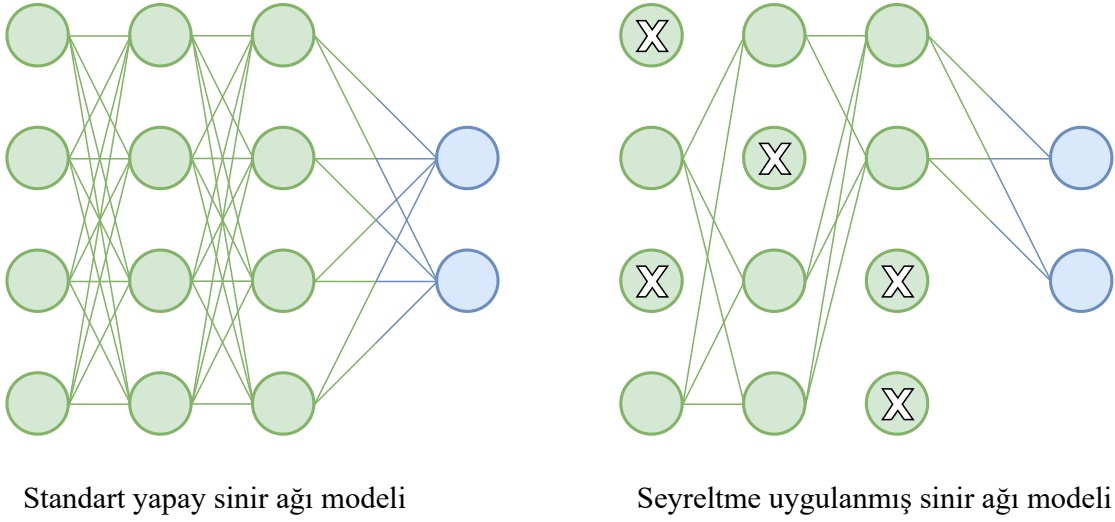
Bu çalışmada yoğun katman ve seyrelme katmanı olmak üzere iki farklı çeşit katman kullanılmıştır. Tekrarlayan katman, evrimsel katman ve havuz katmanları gibi katman çeşitleri görüntü işleme, metin sınıflandırma gibi konularda [48, 49] kullanıldıklarının dolayı bu çalışmada bu katmanlar incelenmemektedir.

4.3.2.1. Yoğun Katman

Yoğun katmanlar, önceki katmanlar ile derin bağlantılı olan katmanlardır. Yoğun katmanların düğümleri önceki katmanın her düğümüyle bağlantı içerisindedir. Bu sebeple ANN modellerinde yoğunlukla tercih edilmektedir. Her düğüm katmanı belirlenen aktivasyon fonksiyonu ile çalışmaktadır. Yoğun katman düğümleri, önceki katmandan elde ettiği çıktıları girdi olarak alarak aktivasyon fonksiyonuna göre hesaplamalar yapıp sonraki katmana iletmektedir.

4.3.2.2. Seyreltme Katmanı

Seyreltme katmanı, ANN modellerinin öğrenme becerisini arttırmayı hedeflemektedir. Model öğrenme safhasındayken, önceden belirlenen bir oranda rastgele olarak düğümleri devre dışı bırakarak her bir öğrenme turunda farklı düğümlerin aktif olmasını sağlamaktadır. Bu sayede modelin veriyi ezberlemesi değil, öğrenmesi hedeflenmektedir [50]. Bu sebeple seyreltme katmanı sadece öğrenme safhasında rol oynamaktadır, tahmin ya da sınıflandırma safhasında iken seyreltme katmanı devreden çıkmaktadır. Seyreltme katmanının öğrenme aşamasındaki etkisi Şekil 4.9'da görülmektedir.



Şekil 4.9 Seyreltme katmanının öğrenme esnasında çalışması mekanizması.

Bu çalışmada ANN modeli oluştururken katman sayısı, düğüm sayısı, aktivasyon fonksiyonu ve katman çeşitleri gibi özellikler üzerinde durulmuştur. Modelin veriye aşırı uyum gösterip ezbere düşmemesi ve aynı zamanda yetersiz kalmaması için teorik topolojik formüller kullanılmıştır [51]. Sonuç olarak, modelde kullanılacak toplam düğüm sayısı, ilk katmanda bulunacak düğüm sayısı ve ikinci katmanda kullanılacak düğüm sayısı Denklem 7 ile hesaplanmıştır.

$$D_{toplam} = 2\sqrt{(m+2)N}$$

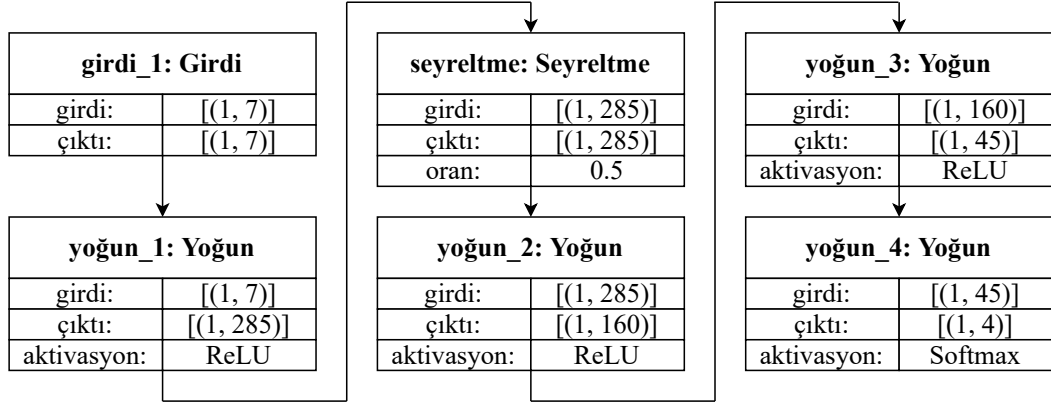
$$D_{birinci} = \sqrt{(m+2)N} + 2\sqrt{N/(m+2)} \quad (7)$$

$$D_{ikinci} = 2\sqrt{N/(m+2)}$$

Denklem 7’de yer alan N toplam veri örneği sayısını, D düğümü ve m sınıflandırmadaki sınıf sayısını belirtmektedir.

Çalışmada 10000 örnek veri ve 4 ağ dilimi sınıflı eğitim verisi kullanılmıştır. Belirtilen denklemler sonucunda toplam düğüm sayısı 490, birinci katmandaki düğüm sayısı 285, ikinci katmandaki düğüm sayısı 160 olarak hesaplanmaktadır. Modelin başarısı için kalan düğüm sayısını tamamlama amacıyla 45 düğümlü bir üçüncü bir katman oluşturulmuştur. Ayrıca, aşırı uyum göstermenin önüne geçmek adına 0.5 oranlı bir seyreltme katmanı eklenmiştir. Bu oranın 0.5 seçilmesinin amacı, modelin düğümlerinin yarısının her eğitim devrinde devre dışı kalmasını sağlayıp eğitim kalitesini yükseltmek olmuştur [52].

Çoklu sınıflı bir sınıflandırma problemi ile çalışıldığından dolayı, çıktı katmanındaki aktivasyon fonksiyonu çoklu sınıflı seçimlerde uyumlu çalışan *softmax* fonksiyonu tercih edilmiştir. Çıktı katmanından önceki yoğun katmanlarda, hızlı hesaplama ve etkinliği sebebiyle *ReLU* fonksiyonu tercih edilmiştir. Sonuç olarak, Şekil 4.10’da görülmekte olan 4 yoğun ve 1 seyreltme olmak üzere 5 katmanlı bir ANN modeli oluşturulmuştur. Aşırı uyum göstermesinin önüne geçmek amacıyla eklenen seyreltme katmanının yanı sıra modelin yapısının da oldukça az katmanlı olması hedeflenmiştir. Düğüm sayısı hesaplamaları kullanılarak hesaplanan birinci ve ikinci katmanlara ek olarak, toplam düğüm sayısına ulaşabilmek için üçüncü bir yoğun katman eklenmiştir. Bu sebeple toplam 5 katmanlı bir yapı elde edilmiştir.



Şekil 4.10 Oluşturulmuş yapay sinir ağı modeli.

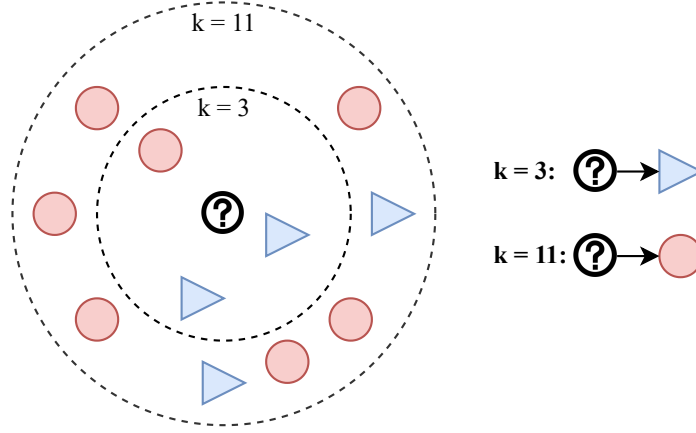
Model yapısı oluşturulduktan sonra eğitim aşamasında uygulanacak, modelin kendisinin başarı ölçülerini belirleyeceği, en iyileyici, kayıp ve doğruluk fonksiyonlarının ayarlanması gerekmektedir. En iyileyici fonksiyonlar model içerisindeki ağırlıkları ve öğrenme oranını belirleyerek kaybı düşürmeye çalışan fonksiyonlardır. Kayıp ve doğruluk fonksiyonları, model eğitim esnasında öğrenme yaparken, modelin kendi başarısını değerlendirmesini sağlayan fonksiyonlardır. Belirlenen ölçülere göre model nasıl bir başarı ile öğrenim yaptığını hesaplar.

En iyileyici fonksiyonlar arasında *Adam* tercih edilmiştir. Bu en iyileyici fonksiyon, etkili çalışma prensipleri ve uyarlamalı yapısından dolayı çoklu sınıflı sınıflandırma problemlerinde tercih edilmektedir [53]. Keras, tek-sıcak kodlama ile etiketlenmiş çoklu sınıflı sınıflandırma problemleri için özel kayıp ve doğruluk fonksiyonları geliştirmiştir. Bunlar kayıp fonksiyonu için *CategoricalCrossentropy*, doğruluk fonksiyonu için *CategoricalAccuracy* olarak belirtilmiştir. Bu sebeple kayıp ve doğruluk fonksiyonları için bu iki fonksiyon seçilmiştir.

4.4. k-En Yakın Komşu

k-NN gözetimli bir öğrenme algoritmasıdır. k-NN modellerinde, veriler kendilerine yakın olan verilere göre sınıflandırılmaktadırlar. Modelin çalışma mekanizması, bir örneğin sınıfı çevresindeki en yakın verilerin çoğunluğuna göre karar verilir şeklinde özetlenebilmektedir.

k-NN modelleri veriye özel olup, eğitildikleri veri sınıfına göre şekil almaktadırlar [54]. Modelin ismindeki k , modelin sınıflandırma esnasında dikkate aldığı en yakın komşu sayısıdır. k-NN modelinin çalışma mekanizması ve farklı k değerleri ile nasıl çalıştığı Şekil 4.11’de görülmektedir.



Şekil 4.11 k-En Yakın Komşu çalışma örneği.

Şekil 4.11’de görüldüğü üzere, $k = 3$ durumunda en yakın 3 komşu veriye bakılıp çoğunluğa göre üçgen sınıflandırılması yapılırken, $k = 11$ durumunda en yakın 11 komşuya bakıldığında daire sınıflandırması yapılmıştır. Bu sebeple k-NN modelleri oluşturulurken seçilecek k değeri ve mesafe hesaplama fonksiyonu önemli olmaktadır.

Bu çalışmada oluşturulan k-NN modeli için Scikit-learn kullanılmıştır. k-NN modeli için k değeri ve mesafe fonksiyonlarının belirlenmesi için GS kullanılmıştır. Scikit-learn içerisinde iki farklı mesafe fonksiyonu bulunmaktadır: bunlar Öklid mesafesi ve Manhattan mesafedir. İki mesafenin de formülleri sırasıyla, Denklem 8’de ve Denklem 9’de görülmektedir.

$$d(x, y) = \sqrt{\sum_{i=1}^n (x_i - y_i)^2} \quad (8)$$

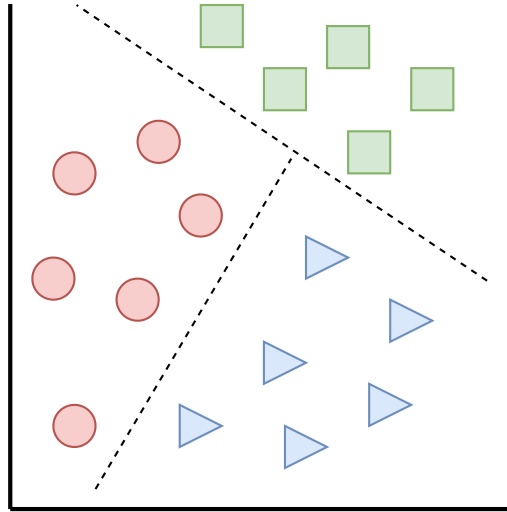
$$d(x, y) = \sum_{i=1}^n |x_i - y_i| \quad (9)$$

Denklemlerde yer alan x_i ve y_i veri noktalarını göstermektedirler. Ayrıca n değeri, veri içerisindeki boyutu belirtmektedir. Noktaların n kadar değerlerinin arasındaki mesafe hesaplanmaktadır.

GS ile birlikte çapraz doğrulama uygulandıktan model için en başarılı değerler bulunmuştur. Bunun sonunda k-NN modeli $k = 5$ ve Manhattan mesafesi kullanılarak oluşturulmuştur.

4.5. Destek Vektör Makineleri

SVM gözetimli bir öğrenme algoritmasıdır. Çoklu sınıflandırma durumlarında kullanışlı olmaktadır. SVM modellerinde, veriler fonksiyon doğruları ile ayrıştırılıp sınıflandırılmaktadırlar. Sınıflandırmanın yapılmasını sağlayan bu doğruya *hiper düzlem*, hiper düzleme yakın olan ve hiper düzlemin pozisyonunu ve özelliklerini etkileyen veri noktalarına da *destek vektörler* denmektedir [55]. Oluşan sınıflandırıcı hiper düzlemler Şekil 4.12'de görülmektedir.



Şekil 4.12 Destek vektör makineleri örnek görseli.

Şekilde görülen görseldeki daire, üçgen ve kare farklı sınıfları temsil ederken, kesikli çizgiler sınıfları ayıran hiper düzlemleri temsil etmektedirler.

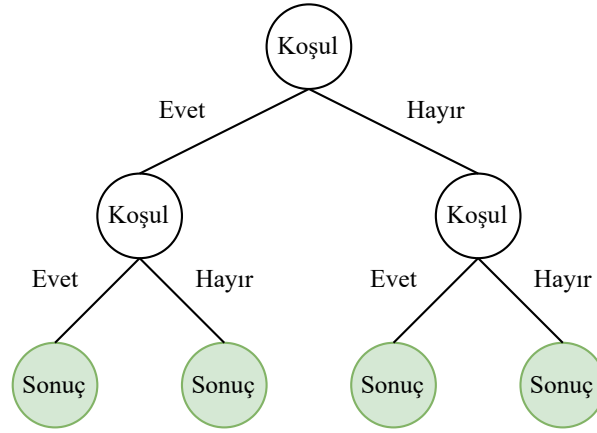
Scikit-learn SVM modelleri, çoklu sınıflı sınıflandırma problemleri için iki farklı hiper düzlem oluşturma yöntemine sahiptir. Bunlar *ovr* ve *ovo* olarak isimlendirilmiştir. *OVR* (*one-vs-rest*) yönteminde hiper düzlem oluşturulurken, seçilen sınıf ayrılarak diğer tüm sınıflar aynı sınıfa ait kabul edilmektedir. *OVO* (*one-vs-one*) yönteminde hiper düzlem oluşturulurken, seçilen sınıf her bir diğer sınıf ile tek tek değerlendirilmektedir. Ayrıca, hiper düzlem oluşturulurken farklı dereceden düzlemler oluşturulabilmektedir. GS ile birlikte çapraz doğrulama uygulandıktan sonra en uygun değerler bulunmuştur. Bu özellikler *OVO* yöntemi ile polinom fonksiyonudur.

4.6. Rastgele Orman

Rastgele orman gözetimli bir topluluk öğrenme algoritmasıdır. Topluluk modeli birden çok tekli modelin birleşip daha güçlü modeli oluşturmasına denmektedir [56]. Rastgele ormanı topluluk modeli yapan etken, içerisinde kullanılan çoklu karar ağaçlarıdır. Rastgele orman içerisindeki karar ağaçları birbirlerinden bağımsız olarak oluşturulmaktadır. Modelin çıktısı, içerisindeki karar ağaçlarının çıktılarına göre belirlenmektedir [57].

4.6.1. Karar Ağacı

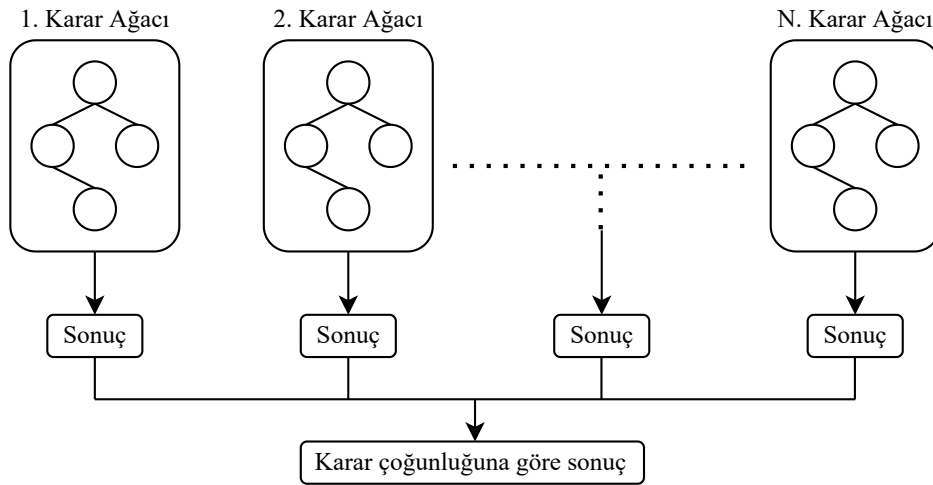
Karar ağacı gözetimli bir öğrenme algoritmasıdır. Sınıflandırma problemlerinde sıkça kullanılmaktadır. Sınıflandırma modeli, veri içerisindeki koşullar ve sonuçlara göre oluşturulmaktadır. Model yapısı kök, yaprak düğümleri ve dallardan oluşmaktadır. Koşul senaryoları düğümlerde yer alırken, koşullar ile ulaşılan sonuçlara da yaprak düğümleri denmektedir [58]. Model, yapısal benzerliğinden dolayı karar ağacı olarak isimlendirilmiştir. Örnek bir karar ağacı modeli Şekil 4.13'te görülmektedir.



Şekil 4.13 Karar ağacı örneği.

Şekil 4.13'te görülen yeşil renkli düğüm noktaları yaprak düğümleri göstermektedir.

Karar ağaçlarından oluşan, ortak karar mekanizması ile çalışan rastgele orman algoritmasının çalışma mekanizması Şekil 4.14'te görülmektedir.



Şekil 4.14 Rastgele orman modeli çalışma mekanizması.

Bu çalışma için oluşturulan veri ile eğitilmiş rastgele orman modelinin bir kısmı Şekil 4.15'te görülmektedir. Şekilde görülen düğüm, çalışmada kullanılan veri ile ilgili koşullar görülmektedir.

```

paket gecikmesi <= 0.19
entropy = 0.15
örnekler = 2002
değer =
[2002, 000]
[2002, 000]
[ . ]
[ . ]
[ . ]
[755, 1247]
[1798, 204]

```

Şekil 4.15 Oluşturulan rastgele orman modelinin örnek kısmı.

Şekil 4.15'te görülen *değer*, her bir sınıf için yer alan dizi ile, belirtilen koşullarda oluşan pozitif ya da negatif durumlarını göstermektedir.

Scikit-learn rastgele orman modelleri oluşturulurken model içerisindeki karar ağacı sayısı ve karar çoğunluğunun hesaplanacağı kriterler belirleyici olmaktadır. Bu kriterler *entropy* ve *gini* olarak adlandırılmışlardır. Entropy katsayısı belirsizliği gösteren, Gini katsayısı ise rastgele seçildiğinde belirli bir değişkenin yanlış sınıflandırılma derecesini veya olasılığını gösteren ölçülerdir. Bu ölçülerin formülleri Denklem 10'da ve Denklem 11'de görülmektedir.

$$Entropy = \sum_{i=1}^L -f(i) \log f(i) \quad (10)$$

Denklem 10'da yer alan L verilerdeki etiket sayısını, $f(i)$ bir düğümdeki etiket i sayısını belirtmektedir.

$$Gini = 1 - \sum_{i=1} f(i)^2 \quad (11)$$

Denklem 11'de yer alan $f(i)$, i sınıfının bulunma olasılığı anlamına gelmektedir.

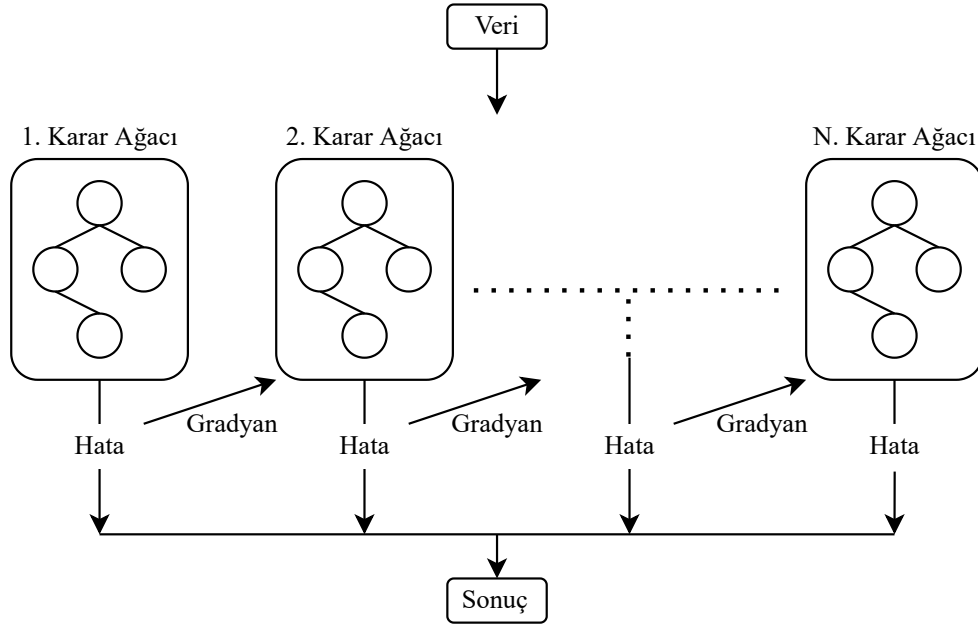
GS ile birlikte çapraz doğrulama uygulandıktan sonra rastgele orman içerisindeki karar ağacı sayısı ve karar kriteri için en uygun ölçüler bulunmuştur. Bu ölçüler 50 karar ağacı ve *entropy* kriteri olmuştur.

4.7. XGBoost - eXtreme Gradient Boosting (Aşırı Gradyan Artırma)

XGBoost gözetimli bir topluluk öğrenme algoritmasıdır. XGBoost algoritmasında yer alan *makine öğrenimini artırma* tekniği de topluluk modellerinde kullanılan tekniklerden biridir. Topluluk modeli içerisindeki ilk modelin oluşturulup, ilk modelin kötü başarımlar gösterdiği kısımlara yoğunlaşarak sıradaki modelin oluşturulduğu ve bu şekilde ilerletildiği bir tekniktir [59]. Böylece oluşturulan her yeni model ile ana topluluk modelinin başarımlarını yükseltilmiş olmaktadır. XGBoost makine öğrenimi artırma tiplerinden *gradyan artırma* yöntemini, karar ağacı algoritması ile birlikte kullanmaktadır.

Gradyan artırma, makine öğrenimi artırma modelleri ile aynı prensiplerde çalışmaktadır. Başarımlar ölçümü yaparken önceki modelin başarımlarını hata payının gradyanına dayalı olarak ayarlamaktadır [60]. İsmi de buradan almaktadır. Oluşturulan her yeni model, hata payını en aza indiren yönde ilerlemektedir. Karar ağacının çalışma şekli de Kısım 4.6.1.'de yer almaktadır.

XGBoost modelleri bu iki yöntemi birlikte kullanmaktadır. Oluşturulan karar ağacı modellerinin çıktılarını, gradyan artırma yöntemiyle değerlendirilip sonraki karar ağaçlarının oluşturulmasını sağlamaktadır [61]. Algoritmanın çalışma şekli Şekil 4.16'da görülmektedir.



Şekil 4.16 XGBoost çalışma mekanizması.

XGBoost modelleri tahmin ve sınıflandırma durumlarında kullanılabilirler. Bu sebeple farklı gradyan artırıcı yöntemleri vardır. Bu çalışmada sınıflandırma amacıyla kullanıldığından, gradyan artırıcı olarak *gbtree* kullanılmıştır.

5. MAKİNE ÖĞRENMESİ TEKNİKLERİ İLE 5G AĞ DİLİMLEME

Önerilen çalışmada, farklı amaçlar ile kullanılma hedefleriyle birçok ML modeli geliştirilmiştir. Geliştirilen ML modellerinin test edilebilmesi için gerçekçi benzetim ortamları geliştirilmesi gerekmektedir. Bu amaçla, 1 km × 1 km bir kentsel arazi içerisine 5 adet baz istasyonu yerleştirilmiş ve çeşitli senaryolar içeren farklı sayıda kullanıcılar ile benzetim ortamları oluşturulmuştur.

Ortamdaki kullanıcıların, 3GPP dokümanları baz alınarak, 5G haberleşme senaryolarına uygun özelliklere ve gereksinimlere sahip olacak şekilde benzetime dahil edilmesi planlanmıştır. Bu dokümanlardan; kullanıcı tipleri ve kullanıcıların paket gecikmesi, paket kaybı, veri miktarı, kaynak tipi gibi değişkenler [62], V2X kullanım senaryolarına ve AR, VR senaryolarına özel gereksinimler [63] ve kentsel, kapalı alan senaryolarındaki gereksinimler [64] bilgileri elde edilmiştir. Veri oluşturma aşamasında dikkate alınacak olan bu dokümanlarda yer alan kullanım tiplerini ve gereksinimlerini Tablo 5.1’de görülebilmektedir.

Tablo 5.1 Bazı kullanım tiplerini ve onların gereksinimlerini içeren 3GPP çizelgesi.

QCI	Kaynak Tipi	Öncelik Seviyesi	Paket Gecikmesi (ms)	Paket Kaybı Oranı	Kullanım Tipi
1	DC-GBR	3	50 ms	10e-3	V2X mesajları; Elektrik dağıtımı; Süreç otomasyonu
2	DC-GBR	1.5	100 ms	10e-3	Görevle İlgili Kritik Video kullanıcı düzlemi
3	GBR	2	100 ms	10e-2	Diyaloğlu iletişim
4	GBR	5	300 ms	10e-6	Diyalog Dışı Video (Arabelleğe Alınmış Akış)
65	GBR	0.7	75 ms	10e-2	Görevle İlgili Kritik kullanıcı düzlemi Bas Konuş sesi (ör. MCPTT)
66	GBR	2	100 ms	10e-2	Görev Açısından Kritik Olmayan kullanıcı düzlemi Bas Konuş sesi
67	GBR	4	150 ms	10e-3	Diyaloğlu Video (Canlı Akış)
75	GBR	2.5	50 ms	10e-2	V2X mesajları
71	GBR	5.6	150 ms	10e-6	"Canlı" Çıkış Bağlantısı Akışı
72	GBR	5.6	300 ms	10e-4	"Canlı" Çıkış Bağlantısı Akışı
73	GBR	5.6	300 ms	10e-8	"Canlı" Çıkış Bağlantısı Akışı
74	GBR	5.6	500 ms	10e-8	"Canlı" Çıkış Bağlantısı Akışı
76	GBR	5.6	500 ms	10e-4	"Canlı" Çıkış Bağlantısı Akışı
5	Non-GBR	1	100 ms	10e-6	IMS Sinyalizasyonu
6	Non-GBR	6	300 ms	10e-6	Video (Tamponlu Akış); TCP tabanlı (ör. p2p dosya paylaşımı vb.)
7	Non-GBR	7	100 ms	10e-3	Ses, Video (Canlı Akış), Etkileşimli Oyun
8	Non-GBR	8	300 ms	10e-6	Video (Tamponlu Akış); TCP tabanlı (ör. www, e-posta, sohbet, ftp)
10	Non-GBR	9	1100 ms	10e-6	Video (Tamponlu Akış); TCP tabanlı (örneğin, aşamalı video vb.)
69	Non-GBR	0.5	60 ms	10e-6	İşle İlgili Kritik Görev gecikmesine duyarlı sinyalizasyon (ör. MC Video sinyali)
70	Non-GBR	5.5	200 ms	10e-6	Görevle İlgili Kritik Veriler
79	Non-GBR	6.5	50 ms	10e-2	V2X mesajları
80	Non-GBR	6.8	10 ms	10e-6	Düşük gecikmeli eMBB uygulamaları (TCP/UDP tabanlı); Arttırılmış gerçeklik

Dünya genelinde 5G haberleşme sisteminin yaygınlaşması için 2025 yılı öngörülmektedir [65]. Bu sebeple literatürdeki çalışmalar sentetik veriler üzerinden ilerlemektedir. Bu çalışmada da 3GPP dokümanlarından faydalanılarak ağ dilimleme ve kullanım tipi sınıflandırmaları için kullanılacak olan sentetik bir veri seti oluşturulmuştur. Benzetim çalışmalarında ve sınıflandırmada kullanılacak bu değişkenler kullanıcı konumu, hareketi, kaynak tipi, paket gecikmesi, paket kaybı ve veri hızı olarak belirlenmiştir.

Bu çalışmada ağ dilimleri eMBB, URLLC, MIoT ve V2X olarak belirlenmiştir. eMBB var olan mobil servislerin uzantısı olarak görülmektedir. eMBB yüksek veri hızlarıyla geniş alanda hizmet veren, URLLC paket gecikmesi ve paket kaybı değişkenlerinde yüksek standartlara sahip olan, MIoT yüksek sayıda cihazlar arasındaki iletişimi sağlayan ve V2X araçlardan diğer cihazlara yaşanan iletişimin gerçekleştiği dilimler olarak planlanmaktadır.

Tablo 5.1’de çeşitli 3GPP dokümanlarından faydalanarak oluşturulan bir takım kullanım tipleri görülmektedir. Bu kullanım tipleri; *dosya paylaşımı, kamuya açık kablosuz ağ, telefon ile konuşmak, duraksız yayın ile iletişim, AR/VR/Oyun, arabelleğe alınmış video, sağlık hizmetleri, kesintili ya da kesintisiz otomasyon, elektrik dağıtımı, akıllı sistemler, kamu güvenliği, akıllı ev/araba, akıllı nakliye, sensör bildirim, otomatik sürüş, video paylaşımı* olarak belirlenmişlerdir.

Sentetik veri oluşturulurken Python [66] kullanılmıştır. Bunun için 3GPP dokümanlarında yer alan kullanım tiplerinin özelliklerinin alt ve üst sınırları baz alınmıştır. Eğitim verisi oluşturulurken tüm sınıflar için eşit sayıda örnek oluşturulmuştur. Kullanılmış sözde kod Algoritma 1’de görülmektedir.

Algorithm 1 Sentetik veri oluřturma Python kodu

Input: Kullanıcı sayısı (N)

Output: Sentetik kullanıcı verisi

```
for  $i = 0$  to  $N$  do
  if Ağ Dilimi =  $X$  then
    kullanım tipi = rastgele( $X$  ağ diliminin kullanım tipleri)
    if kullanım tipi =  $Y$  then
      Kaynak Tipi =  $Y$  kullanım tipine özel kaynak tipi
      Paket Gecikmesi = rastgele( $A, B$ )
      Paket Kaybı Oranı = rastgele( $C, D$ )
      Veri Hızı = rastgele( $E, F$ )
      Hareketlilik = rastgele( $G, H$ )
    end if
  end if
end for
return Sentetik kullanıcı verisi
```

Algoritma 1’de görülen N oluřturulmak istenen kullanıcı sayısını belirtmektedir. Algoritma içerisinde, ağ dilimlerine ve kullanım tiplerine dikkat edilerek, 3GPP dokümanlarında belirtilen özelliklere göre kullanıcılar oluřturulmuřtur. Algoritmada yer alan A, B, C, D, E, F, G, H deęerleri sembolik olarak yer almaktadır. Her biri, farklı kullanım tiplerindeki alt ve üst seviyeleri temsil etmektedir. Ayrıca *rastgele* ise belirtilen deęerler arasından rastgele bir deęer seęilmesini saęlayan iřlevi temsil etmektedir.

3GPP dokümanları baz alınarak ve Algoritma 1’da görülen kod kullanılarak, benzetim çalışmalarında kullanılmak üzere sentetik olarak üretilen kullanıcı verileri Tablo 5.2’de görölmektedir.

Tablo 5.2’de yer alan *hareketlilik* kullanıcının hareket hızını, *kaynak tipi* kullanıcının anlaşmasına göre hizmetinde bir kayıp yaşamaması (guarenteed bitrate, GBR), gecikme ve kayıp yaşamaması (delay-critical guarenteed bitrate, DC-GBR) ya da bu konuda bir

Tablo 5.2 5G senaryoları içeren örnek sentetik kullanıcı verileri.

Hareketlilik (m/s)	Kaynak Tipi	Paket Gecikmesi (ms)	Paket Kaybı Oranı	Veri Hızı (byte)	Kullanım Tipi	Ağ Dilimi
0.47	GBR	49.26	5e-06	555	Elektrik Dağıtım	URLLC
1.85	NON-GBR	240.42	3e-02	160534	Dosya Paylaşımı	eMBB
0.74	GBR	25.43	8.6e-05	95334	Sensör Bildirimi	V2X
1.13	NON-GBR	11.73	2.3e-04	2834542	AR/VR/Oyun	eMBB
66.79	GBR	64.61	8e-06	200664	Otomatik Sürüş	V2X
1.50	DC-GBR	1.24	5.6e-07	397	Sağlık Hizmetleri	URLLC
1.60	GBR	186.30	3.5e-04	265	Akıllı Ev/Araba	MIoT

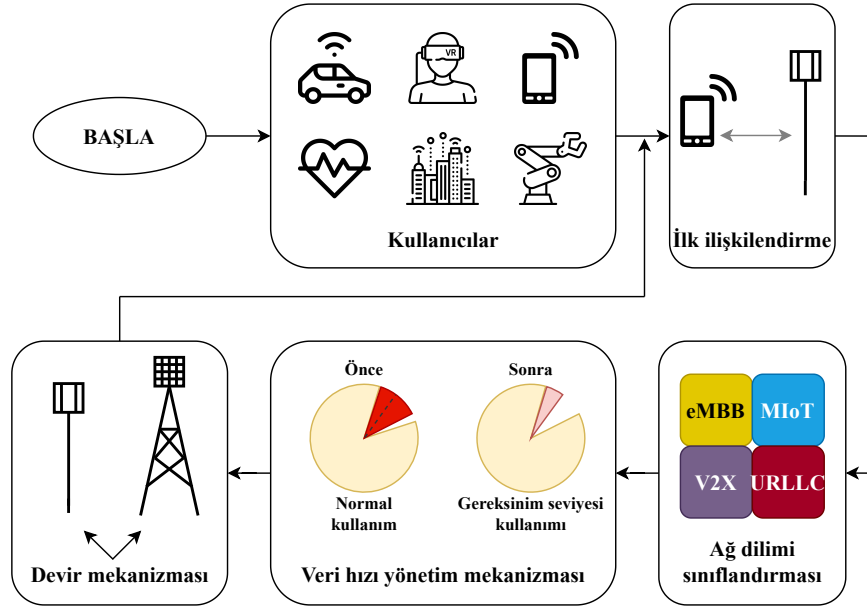
garantisinin olmamasını (non guaranteed bitrate, NON-GBR), *paket gecikmesi* iletilen veri paketlerinin vericiden alıcıya ne kadar gecikme ile ulaştığını, *paket kaybı oranı* veri paketi iletilirken veri içerisindeki kayıp oranını ve *veri hızı* ise veri paketlerinin iletilme hızını belirtmektedir.

5.1. Sistem Modeli

Çeşitli senaryolar içeren yüksek sayıdaki kullanıcıların, benzetim ortamında sorunsuz şekilde hizmet alması sağlanması hedeflenmektedir. Kullanıcıların sadece doğru ağ dilimine yerleştirilmesi değil, hareket halindeyken kanal parametresi değişimlerinden dolayı hizmet kesintisi yaşamaması, baz istasyonu yoğunluklarından dolayı hizmet kapasitesi yetmemesinden kaynaklı hizmet alamaması gibi olumsuzluklardan da etkilenmemesi amaçlanmaktadır.

Sistemin benzetim çalışması süresince sorunsuz çalışması ve yüksek başarımlı göstermesi çeşitli ML modelleri ve ek mekanizmalar kullanımıyla hedeflenmektedir. Benzetim çalışması içerisinde bir kullanıcının karşılaşılabileceği durumlar düşünülüp, o durumlara çözüm bulunması amaçlanmıştır. Bu doğrultuda ilk önce kullanıcı kanal parametreleri baz alınarak bir kullanıcı-baz istasyonu ilişkilendirilmesi yapılmıştır, sonrasında baz istasyonu belirlenen kullanıcının ML modelleri yardımıyla hangi ağ dilimine ait olduğu belirlenmiştir. Ağ dilimi sınıflandırmasının ardından, baz istasyonunda yaşanabilecek olası bir kapasite azalması probleminde RF modeli içeren veri hızı yönetim mekanizması devreye girmektedir. Son olarak kullanıcının hareketliliği senaryolarında olası bir kanal parametresi sıralaması

değişimi durumunda da devir mekanizması ile kullanıcı devri gerçekleştirilmektedir. Şekil 5.1’de bahsedilen sistem modelinin çalışma sırası görülmektedir.



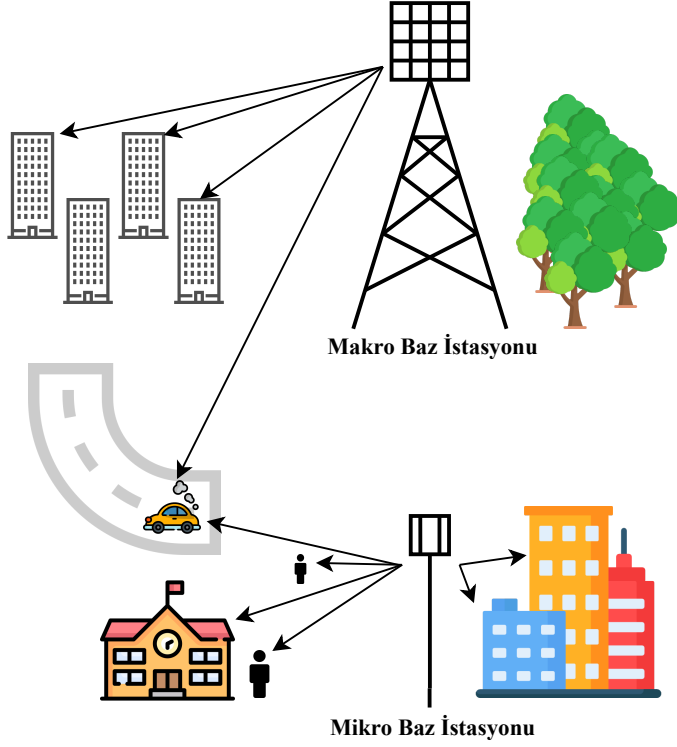
Şekil 5.1 Sistem çalışma şeması.

Bu kısmın devamı, Şekil 5.1’de görülen sistem çalışma mekanizması sırasıyla anlatılacaktır.

5.2. Baz İstasyonlarının Benzetim Ortamına Konumlandırılması

Kullanıcı verisinin oluşturulması sonrasında; baz istasyonlarının oluşturulması, benzetim ortamında konumlandırılması ve baz istasyonunun tipinin belirlenmesi gerekmektedir. Gerçekçi bir benzetim ortamı oluşturulurken farklı tip, makro ve mikro, baz istasyonları konumlandırılması planlanmıştır.

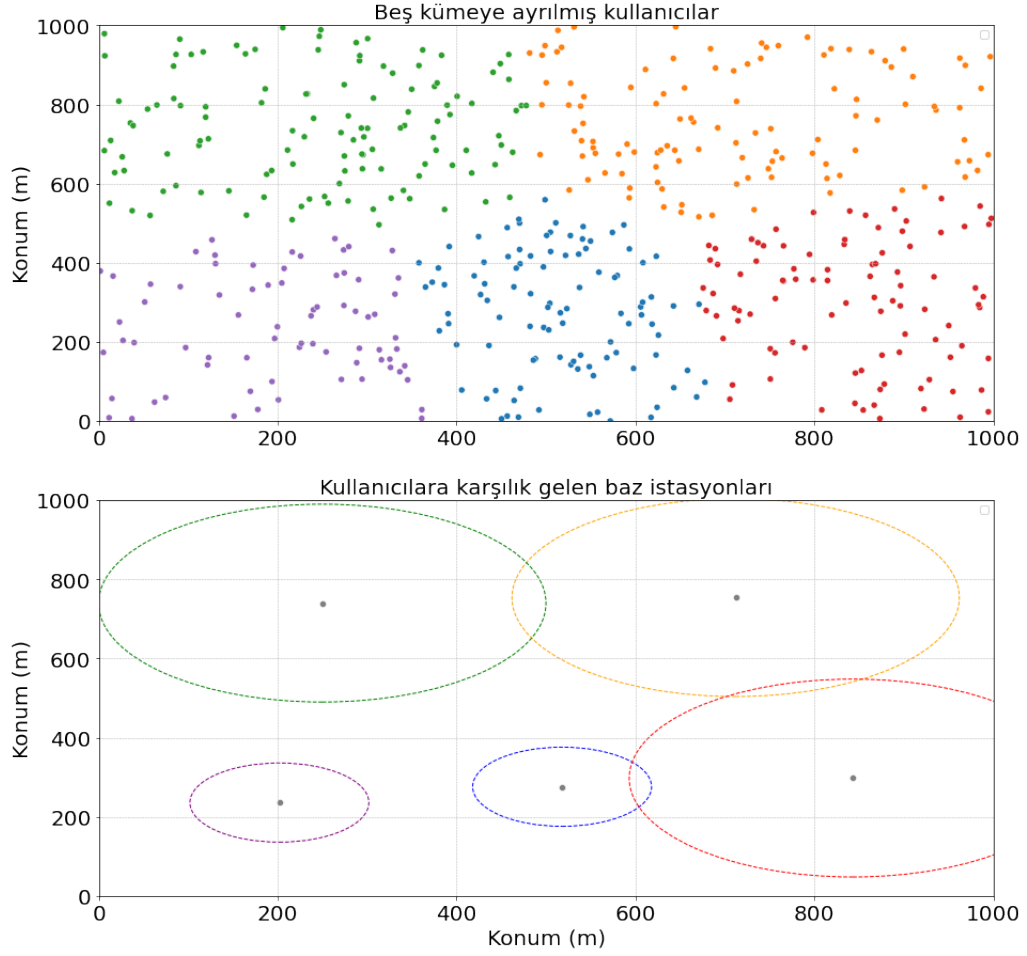
Makro baz istasyonları geniş kapsama alanına sahip, yüksek sayıda kullanıcıya hizmet verebilen, ancak yapısından dolayı sıkışık alanlarda kullanılmaya elverişli olmayan, mikro baz istasyonları ise dar kapsama alanına sahip, düşük sayıda kullanıcıya hizmet verebilen ancak sıkışık alanlar için elverişli ve efektif çalışan baz istasyonlarıdır [67], [68]. Bahsedilen baz istasyonu tipleri Şekil 5.2’de görülmektedir.



Şekil 5.2 Makro ve Mikro Baz İstasyonları ile oluşturulmuş örnek ortam

Baz istasyonlarının konum bilgisine karar verilme aşamasında benzetim alanındaki geçmiş kullanıcı verileri kullanılmıştır. Kullanıcı konum verilerinden faydalanılarak ML yöntemlerinden k-ortalama kümelemesi (4.2. Kısım) kullanılarak baz istasyonlarının merkez konumları belirlenmiştir.

Baz istasyonlarının merkez konum bilgileri elde edildikten sonra, merkezlere yerleştirilecek olan baz istasyonu tipine karar verilmiştir. Bu karar verilirken geçmiş kullanıcı veri ihtiyaçları göz edilmiştir. En yoğun bölgelere makro baz istasyonu konumlandırılırken, diğer bölgelere mikro baz istasyonları konumlandırılmıştır. Bu mekanizmanın sözde kodu Algoritma 2’de, geçmiş kullanıcıların kümelmesi ve kümelenen kullanıcılara göre baz istasyonu tipi belirlenmesi Şekil 5.3’te görülmektedirler.



Şekil 5.3 Kümeleme yöntemi çıktısı.

Algoritma 2’de kullanılan; N baz istasyonu sayısını, x ve y lokasyon koordinatlarını, v geçmiş kullanıcı veri bilgilerini, B çıktı olarak elde edilen baz istasyonu bilgilerini, $k_ortalama_kümelemesi$ kullanıcıları konumlarına göre gruplarına ayırıp kullanıcıları bu gruplara göre etiketleyen ve ayrılmış grupların merkezlerini belirten işlevi, $birleştir$ farklı verileri birleştirmek için kullanılan işlevi, $indeks$ bir liste içerisinde belirtilen noktanın nerede olduğunu belirtmektedirler.

Geçmiş kullanıcı konumları kullanarak merkez noktalar belirlenmesinin ardından, geçmiş kullanıcı veri gereksinimlerinden yararlanılarak baz istasyonu tipi belirlenmiştir. Baz istasyonu tipi belirlendikten sonra, bu tiplere uygun özellikler ile baz istasyonları

Algorithm 2 Baz istasyonlarının konumlarının ve tiplerinin belirlenmesi.

Input: Baz istasyonu sayısı (N), Geçmiş kullanıcıların konum bilgileri (x, y), Geçmiş kullanıcıların veri hızı gereksinim bilgileri (v)

Output: Baz istasyonlarının özellikleri (B)

merkez, etiket = $k_{\text{ortalama kümelemesi}}(x, y)$

Etiketlenmiş kullanıcı listesi = $\text{birleştire}(v, \text{etiket})$

birinci = ikinci = üçüncü = $-\infty$

for $i = 0$ to N **do**

if (Etiketlenmiş kullanıcı listesi)[i] > birinci **then**

 üçüncü = ikinci

 ikinci = birinci

 birinci = (Etiketlenmiş kullanıcı listesi)[i]

else if (Etiketlenmiş kullanıcı listesi)[i] > ikinci **then**

 üçüncü = ikinci

 ikinci = (Etiketlenmiş kullanıcı listesi)[i]

else if (Etiketlenmiş kullanıcı listesi)[i] > üçüncü **then**

 üçüncü = (Etiketlenmiş kullanıcı listesi)[i]

end if

end for

for $J = 0$ to N **do**

if J in (Makro baz istasyonları) **then**

 Ekle: Tip(1)

else

 Ekle: Tip(0)

end if

end for

if Tip = 1 **then**

 uzunluk = 25

 kapasite = $5e7$

 yarıçap = 250

else

 uzunluk = 10

 kapasite = $1e7$

 yarıçap = 100

end if

$B = (\text{kapasite}, \text{merkez}, \text{yarıçap}, \text{uzunluk})$

return B

oluşturulmuştur. Benzetim ortamında kullanılmak üzere oluşturulan baz istasyonları özellikleri Tablo 5.3'te görülmektedir.

Tablo 5.3 Benzetim ortamında kullanılacak baz istasyonu verileri.

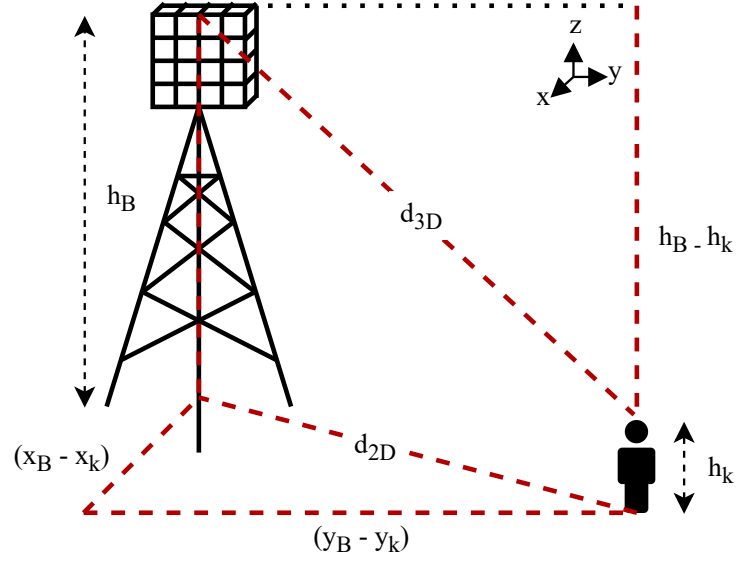
Baz istasyonu tipi	Kapasite (byte)	Merkez	Yarıçap (m)	Uzunluk (m)
Makro	5e7	(712, 754)	250	25
Mikro	1e7	(202, 236)	100	10
Mikro	1e7	(518, 276)	100	10
Makro	5e7	(843, 298)	250	25
Makro	5e7	(250, 740)	250	25

5.3. Kullanıcı ve Baz İstasyonları Arasındaki Kanal Parametresi ve İlişkilendirilmeleri

Kullanıcılar ve baz istasyonları oluşturulması sonrasında bu ögelerin birbiri ile ilişkilendirilmesi gerekmektedir. Bu ilişkilendirme için bir kanal parametresi oluşturulmuştur. Kanal parametresinin oluşturulması aşamasında yol kaybı (path loss, PL) ve gölgeleme kaybı (shadow loss, SL) parametreleri baz alınmıştır [69].

PL sinyal baz istasyonundan kullanıcıya iletilirken yolda yaşanan kayba, SL ise alıcı ve verici arasındaki yayılma yolunu tıkayan nesnelere nedeniyle alınan sinyal gücünün dalgalanma etkisine denmektedir.

Benzetim çalışmaları kentsel arazi içerisinde gerçekleştiğinden dolayı, görüş hattı olmayan (non line of sight, NLOS) ve üç boyutlu mesafe kullanılmıştır. Üç boyutlu mesafe hesaplama görseli ve denklemi sırasıyla, Şekil 5.4'te ve Denklem 12'de görülmektedir.



Şekil 5.4 Üç boyutlu mesafe tanımı.

$$d_{3D} = \sqrt{(h_k - h_B)^2 + (x_k - x_B)^2 + (y_k - y_B)^2} \quad (12)$$

Denklem 12’de yer alan x_k, x_B, y_k, y_B kullanıcı ve hücre konumlarını belirtirken, h_B, h_k ise yüksekliklerini belirtmektedir.

Hesaplanan üç boyutlu mesafe kullanılarak makro ve mikro baz istasyonları için PL hesaplanması Denklem 13’te görülmektedir.

$$\begin{aligned} PL_{UMa} &= 10^{(32.4+20 \log_{10} f_c+30 \log_{10}(d_{3D}))/10} \\ PL_{UMi} &= 10^{(32.4+20 \log_{10} f_c+31.9 \log_{10}(d_{3D}))/10} \end{aligned} \quad (13)$$

Denklem 13’te görülen PL_{UMa} makro baz istasyonlarının, PL_{UMi} ise mikro baz istasyonlarının PL değerini ifade etmektedir. Denklemde yer alan f_c $3.5GHz$ olarak seçilmiştir. Sonrasında makro ve mikro baz istasyonlarına özel gölge solması dikkate alınarak SL hesaplanmıştır.

$$\begin{aligned}
SL_{UMa} &= (10^{\sigma_{UMa}/10})rastgele(0, 1) \\
SL_{UMi} &= 10^{\sigma_{UMi}/10}rastgele(0, 1)
\end{aligned}
\tag{14}$$

Denklem 14'te görülen σ gölge solmasını, SL_{UMa} makro baz istasyonlarının, SL_{UMi} ise mikro baz istasyonlarının SL değerini belirtmektedir.

Daha sonra Gauss dağılımı kullanılarak oluşturulan kanalın Rayleigh dağılımı ve sonrasında daha önce hesaplanan PL ve SL kullanılarak bir kanal parametresi oluşturulmuştur.

$$\begin{aligned}
G(x) &= \frac{1}{\sqrt{2\pi\sigma^2}} \exp \frac{\pi^2}{2\sigma^2} \\
R(x) &= \|G(x)G(x)\| \\
P(x) &= \frac{R(x)}{PLSL}
\end{aligned}
\tag{15}$$

Denklem 15'te yer alan $G(x)$ Gaussian dağılımını, $R(x)$ Rayleigh dağılımını ve $P(x)$ de oluşturulan kanal parametremizi temsil etmektedir.

Kullanıcı kanal parametreleri hesaplandıktan sonra baz istasyonlarına olan atamalar bu parametre baz alınarak yapılmaktadır. Baz istasyonları arasında, kullanıcı ile en güçlü kanal parametresine sahip olan kullanıcı ile eşleştirilmektedir. Kanal parametresi belirlenen eşğin altında olan kullanıcılar hizmet almamaktadırlar. Bahsedilen mekanizmanın sözde kodu Algoritma 3'te görülmektedir.

Algoritma 3'te kullanılan; N baz istasyonu sayısını, \mathbf{B} baz istasyonu özelliklerini, (x, y) kullanıcı konum bilgisini, b algoritma sonucunda kullanıcı ile eşleştirilen baz istasyonunu belirtmektedir.

Algorithm 3 Kullanıcı-Baz istasyonu ilişkisi kurulması.

Input: Baz istasyonu sayısı (N), Baz istasyonlarının özellikleri (\mathbf{B}), Kullanıcının konum bilgisi ((x, y)), Eşik (T)

Output: Kullanıcının atandığı baz istasyonu (b)

for $i = 0$ to N **do**

$$d_{3D} = \sqrt{(1.5 - \mathbf{B}_{i,h})^2 + (x - \mathbf{B}_{i,merkez(x)})^2 + (y - \mathbf{B}_{i,merkez(x)})^2}$$

if $\mathbf{B}_i = \text{Makro}$ **then**

$$\sigma = 7.8$$

$$PLUMa = 10^{(32.4 + 20 \log_{10} f_c + 30 \log_{10}(d_{3D}))/10}$$

$$SLUMa = (10^{\sigma_{UMa}/10}) \text{rastgele}(0, 1)$$

$$G_i = \frac{1}{\sqrt{2\pi\sigma^2}} \exp \frac{\pi^2}{2\sigma^2}$$

$$R_i = \|G_i G_i\|$$

$$P_i = \frac{R_i}{PLSL}$$

else if $\mathbf{B}_i = \text{Mikro}$ **then**

$$\sigma = 8.2$$

$$PLUMi = 10^{(32.4 + 20 \log_{10} f_c + 31.9 \log_{10}(d_{3D}))/10}$$

$$SLUMi = 10^{\sigma_{UMi}/10} \text{rastgele}(0, 1)$$

$$G_i = \frac{1}{\sqrt{2\pi\sigma^2}} \exp \frac{\pi^2}{2\sigma^2}$$

$$R_i = \|G_i G_i\|$$

$$P_i = \frac{R_i}{PLSL}$$

end if

end for

for $i = 0$ to N **do**

if $P_i = \arg \max(P)$ AND $\arg \max(P) \geq T$ **then**

$$b = i + 1$$

else if $\arg \max(P) \leq T$ **then**

Kullanıcı ile baz istasyonu ilişkilendirilemedi: $b = 0$

end if

end for

return b

5.4. Ağ Dilimleme Mekanizması

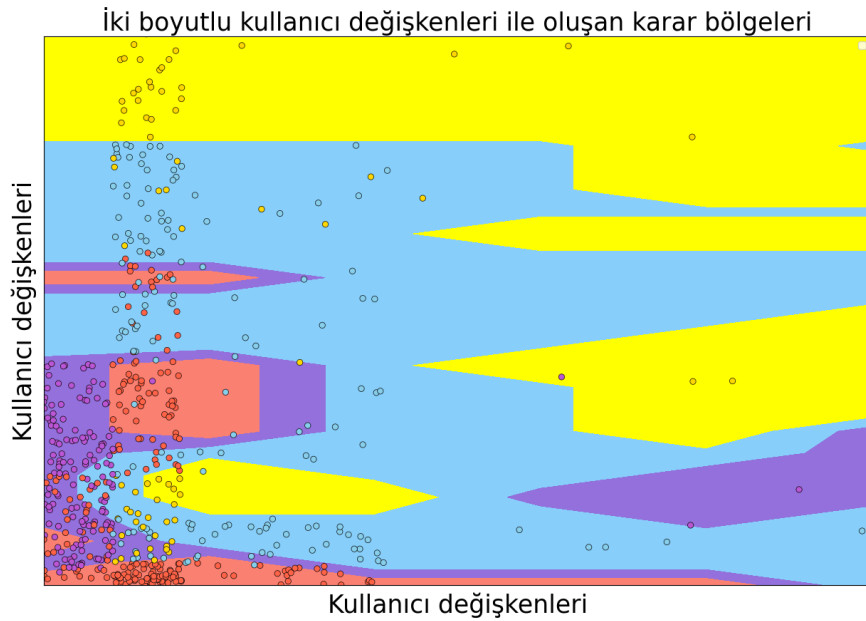
Kullanıcı ile baz istasyonu eşleştirilmesi gerçekleştirildikten sonra farklı ML modelleri kullanılarak kullanıcıların gereksinimlerine en uygun ağ dilimine yerleştirilmesi işlemi gerçekleştirilmektedir. Ağ dilimi sınıflandırması için ANN, k-NN, SVM ve XGBoost olmak üzere dört farklı ML modeli kullanılmaktadır. Her bir ML modeli kendi benzetim ortamlarında test edilmektedir. Bu kısmın devamında, kullanılan makine öğrenmesi modellerinin seçilme sebeplerinden bahsedilmektedir. Birbirlerinden farklı özelliklere sahip,

ancak her biri kullanılan veriye uygun olan bu modellerin başarımları karşılaştırıldıktan sonra modellerin kararından yola çıkılarak, ortak karar içeren bir ana ağ dilimleme mekanizmasına değinilecektir.

5.4.1. Yapay Sinir Ağı

Çalışmada kullanılan veriler sentetik olarak oluşturulduğundan dolayı yüksek sayıda veri örneği ile öğrenme gerçekleştiren ANN modeli ile ilgili bir zorluk ile karşılaşılmamıştır. ANN modeli eldeki veriyi ezberlemek yerine öğrenmeye odaklandığından dolayı, oluşabilecek hata payı durumlarına çok yüksek seviyede adaptasyon gösterip başarılı bir sınıflandırma yapmaya devam edebilmektedir [70].

Kısım 4.3.'te bahsedildiği gibi ANN modeli oluşturulurken toplam ve birinci ve ikinci katmanlardaki düğüm sayılarına teorik yaklaşımlar ile karar verilmiştir. Bunun sonucunda oluşturulan model aşırı uyum gösterip ezberlemeye düşmeden veriyi öğrenmeyi başarmıştır. Çalışmada ağ dilimi sınıflandırmasını gerçekleştirecek olan ANN modelinin karar bölgeleri Şekil 5.5'te görülmektedir.

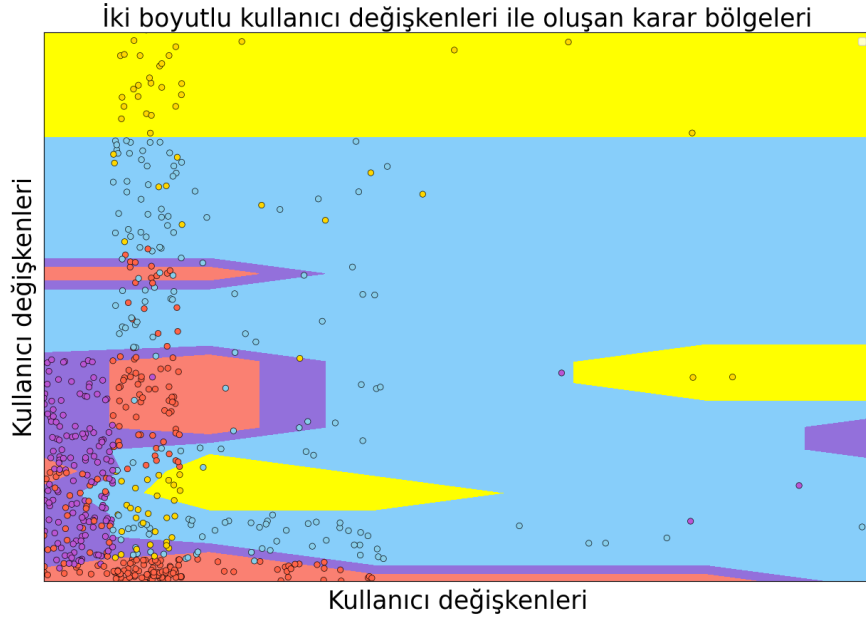


Şekil 5.5 ANN karar bölgeleri.

5.4.2. k-En Yakın Komşu

k-NN modelleri sınıflandırma problemlerinde yoğunlukla kullanılan modellerdir. Birçok algoritma ikili sınıflandırmadan çoklu sınıflandırmaya geçerken sorun yaşarken, k-NN modelleri için bu kolay olmaktadır [71]. k-NN modelleri veriyi öğrenmek yerine ezberlemeye çalışır, bu sebeple esneklik payı düşüktür. Ancak, ezberleme yaklaşımı ve düşük parametreliliğinden dolayı k-NN modelleri çok hızlı çalışmaktadır.

Kullanılan sentetik veri ile eğitildikten sonra oluşan karar bölgeleri Şekil 5.6'da görülmektedir. Görülen her renkli bölge ağ dilimi bölgelerini, noktalar kullanıcıları, noktaların rengi ise kullanıcıların modelin kararına göre buldukları ağ dilimini belirtmektedir.



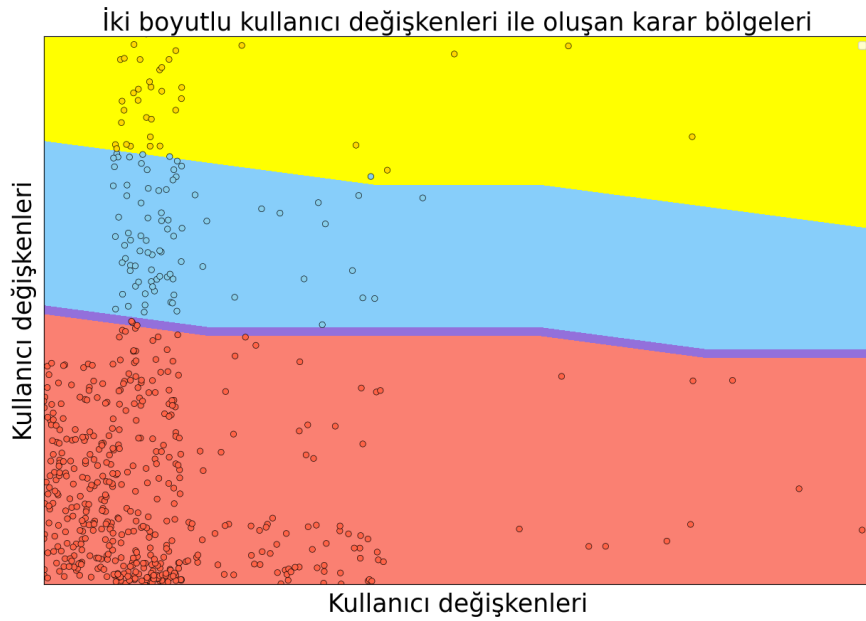
Şekil 5.6 k-NN karar bölgeleri.

5.4.3. Destek Vektör Makineleri

SVM modelleri yüksek değişkenli verilerle birlikte başarılı çalışabilmektedir. Sınıflar arası mesafe oluşturulup sınıflandırma gerçekleştirdiğinden dolayı düşük hata paylarına iyi tepki gösterebilmektedir [72]. Veri içerisinde yoğun üst üste binmeler yaşandığı durumlarda

başarımı düşmektedir. Ancak, çalışmada kullanılan veri içerisinde kullanım tipine göre farklılıklar söz konusu olduğundan bu çalışmada böyle bir sorun söz konusu olmamaktadır.

Kullanılan sentetik veri ile eğitildikten sonra oluşan karar bölgeleri Şekil 5.7’de görülmektedir. Görülen her renkli bölge ağ dilimi bölgelerini, noktalar kullanıcıları, noktaların rengi ise kullanıcıların modelin kararına göre buldukları ağ dilimini belirtmektedir.



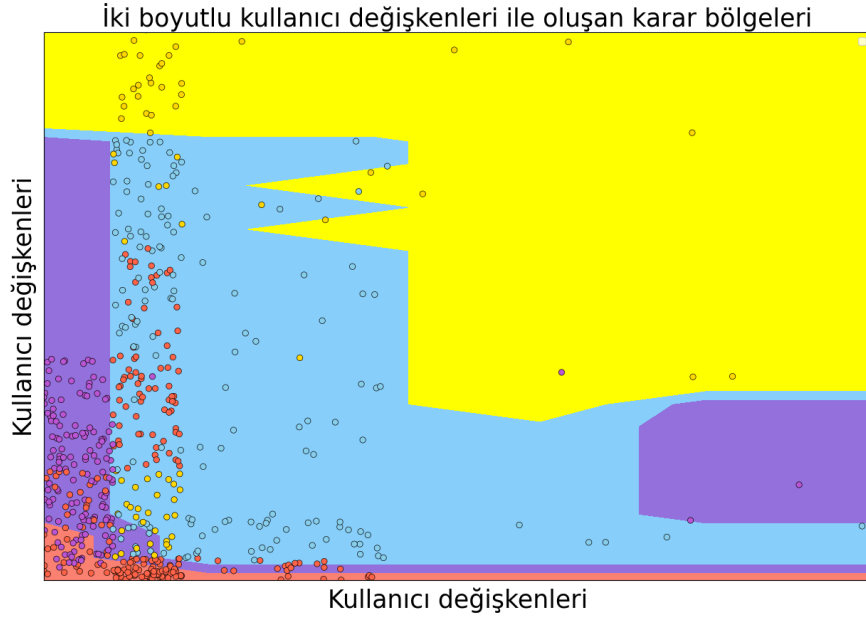
Şekil 5.7 SVM karar bölgeleri.

5.4.4. XGBoost

XGBoost modelleri sınıflandırma ve yakınsama çalışmalarında yoğun olarak kullanılmaktadır [73]. Yapısal veriler ile yüksek başarımlar vermektedir. Bu sebeple oluşturulan sentetik veri XGBoost modeline uygun olmaktadır. Yapısı gereği, öğrenirken sürekli hata düzelttiğinden uç değerlerin sınıflandırmasında çok hassas davranmaktadır.

Kullanılan sentetik veri ile eğitildikten sonra oluşan karar bölgeleri Şekil 5.8’de görülmektedir. Görülen her renkli bölge ağ dilimi bölgelerini, noktalar kullanıcıları,

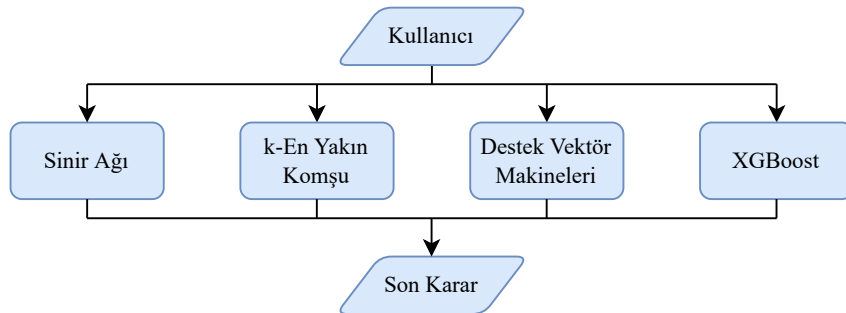
noktaların rengi ise kullanıcıların modelin kararına göre buldukları ağ dilimini belirtmektedir.



Şekil 5.8 XGBoost karar bölgeleri.

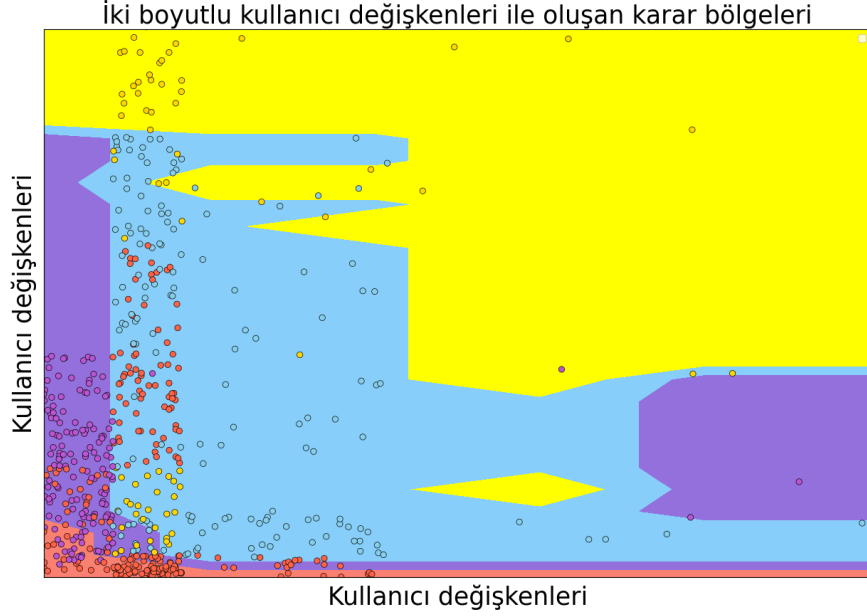
5.4.5. Ana Ağ Dilimleme Mekanizması

Her bir modelin kendine özgü avantaj ve dezavantajları bulunmaktadır. Farklı senaryolarda birbirlerine üstün gelen başarımlar göstermektedirler. Bu sebeple, ana karar mekanizması için dört modelin de gücüne başvurulması düşünülmüştür. Bahsedilen ana ağ dilimleme mekanizması Şekil 5.9’da görüldüğü gibi dört ML modelinin ortak çoğunluk kararına göre son kararı belirleyip sonuca ulaşmaktadır.



Şekil 5.9 ML modellerinin ortak ağ dilimi karar verme mekanizması

Karar çoğunluğu mekanizması ile elde edilen karar bölgeleri Şekil 5.10'da görülmektedir.



Şekil 5.10 Karar çoğunluğu karar bölgeleri.

Modellerin karar bölgeleri görsellerinde yer alan noktalar kullanıcıları, bölgeler ise ağ diliminin kapsadığı alanı belirtmektedir. *Sarı* ögeler eMBB dilimini, *mavi* ögeler MlOT dilimini, *mor* ögeler V2X dilimini, *kırmızı* ögeler URLLC dilimini temsil etmektedir.

5.5. Veri Hızı Yönetim Mekanizması

Benzetim ortamı içerisinde zorlu durumların yönetilebilmesi için yan mekanizmalar üretilmesi düşünülmüştür. Bu mekanizmalardan biri veri hızı yönetme mekanizmasıdır. Veri hızı yönetme mekanizması, yeni bir kullanıcı ile baz istasyonuna ilişkilendirmesi yapılırken, baz istasyonunun kapasitesi belli bir eşiğin altında olduğu zaman devreye girmektedir. 3GPP standartlarında çalışmada kullanılan her bir kullanım tipinin 5G gereksinimleri belirtilmektedir.

Mekanizmanın içerisinde bir RF modeli bulunmaktadır. RF modeli sayesinde, belirtilen gerekli durumlar oluştuğu takdirde, yeni ilişkilendirilen kullanıcının veri havuzu içerisindeki, *dosya paylaşımı*, *kamuya açık kablosuz ağ*, *telefon ile konuşmak*, *duraksız yayın ile iletişim*,

arttırılmış gerçeklik (*augmented reality, AR*)/yapay gerçeklik (*virtual reality, VR*)/Oyun, arabelleğe alınmış video, sağlık hizmetleri, kesintili ya da kesintisiz otomasyon, elektrik dağıtımı, akıllı sistemler, kamu güvenliği, akıllı ev/araba, akıllı nakliye, sensör bildirim, otomatik sürüş, video paylaşımı, kullanım tipi sınıflandırması yapılmaktadır. Kullanıcının kullanım tipi belirlendikten sonra, o kullanım tipi için belirtilen 5G standartlarında veri hızı hizmeti verilmektedir. Veri hızı yönetim mekanizmasının çalışma prensibi Algoritma 4'te görülmektedir.

Algorithm 4 Veri hızı yönetim mekanizması.

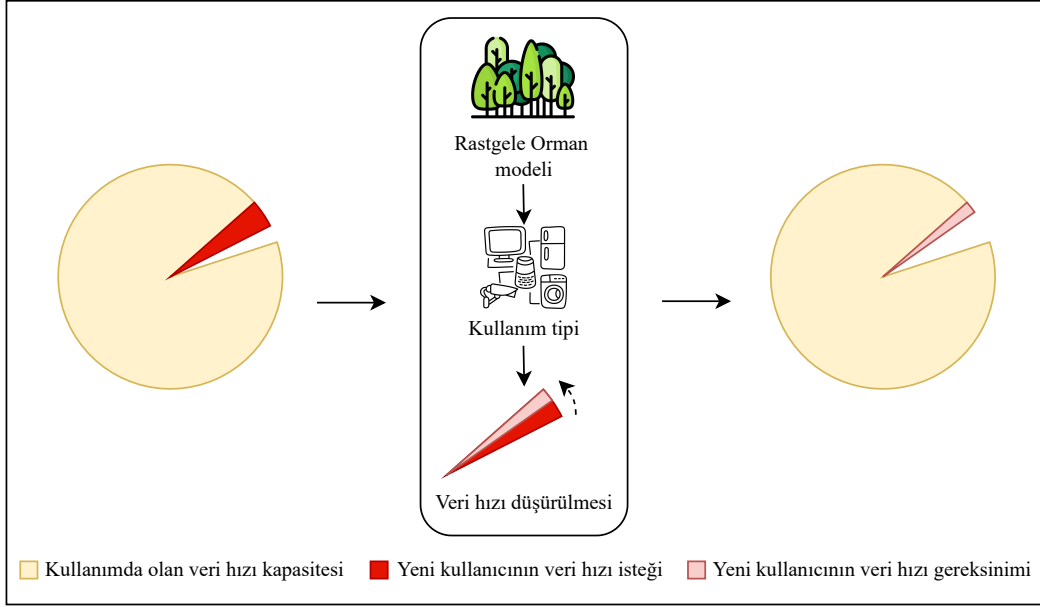
Input: Baz istasyonunun kalan kapasitesi (K), baz istasyonunun toplam kapasitesi (K_{Toplam}), veri hızı düşürülmesi için eşik oranı (T), kullanıcı değişkenleri

Output: Kullanıcı veri hızı (R)

```
if  $K/K_{\text{Toplam}} \leq T$  then
    kullanım tipi =  $RF\_modeli$ (kullanıcı değişkenleri)
     $R = \min$ (kullanım tipi)
else
     $R =$  kullanıcı değişkenleri[veri hızı]
end if
return  $R$ 
```

Algoritma 4'te kullanılan; K baz istasyonunun kalan kapasitesini, K_{Toplam} baz istasyonunun toplam kapasitesini, T mekanizmanın devreye gireceği eşik oranını, RF_modeli kullanım tipini sınıflandırma için kullanılan ML modelini, *kullanıcı değişkenleri* ise sınıflandırmada kullanılan *hareketlilik, kaynak tipi, paket gecikmesi, paket kaybı oranı ve veri hızı* değişkenlerini belirtmektedir.

Veri hızı yönetim mekanizması ile benzetim ortamındaki baz istasyonların başarılı bir şekilde hizmet verme süresinin uzatılması planlanmaktadır. Mekanizmanın devreye girdiği durumdaki etkisi Şekil 5.11'de görülmektedir.



Şekil 5.11 Veri hızını gereksinim seviyesine düşürülme görseli.

5.6. Baz İstasyonları Arasında Kullanıcı Devir Mekanizması

Gerçekçi senaryolarda gerçekleşebilecek zorlu durumlara başa çıkabilmek adına oluşturulmuş bir diğer mekanizma ise kullanıcı devri mekanizması olmaktadır. Kullanıcıların düzenli hareketliliği ve baz istasyonları arasındaki kanal parametresi değişiklikleri düşünülerek bir devir mekanizması oluşturulmuştur. Devir mekanizmasının sözde kodu Algoritma 5’te görülmektedir.

Algorithm 5 Kullanıcı devir mekanizması algoritması

Input: Kullanıcı sayısı (N), baz istasyonu sayısı (M), i . kullanıcı ile j . baz istasyonu arasındaki kanal parametresi ($h_{i,j}$), kullanıcı-baz istasyonu ilişkilendirme vektörü (\mathbf{a})

Output: Yeni ilişkilendirme vektörü (\mathbf{a})

```

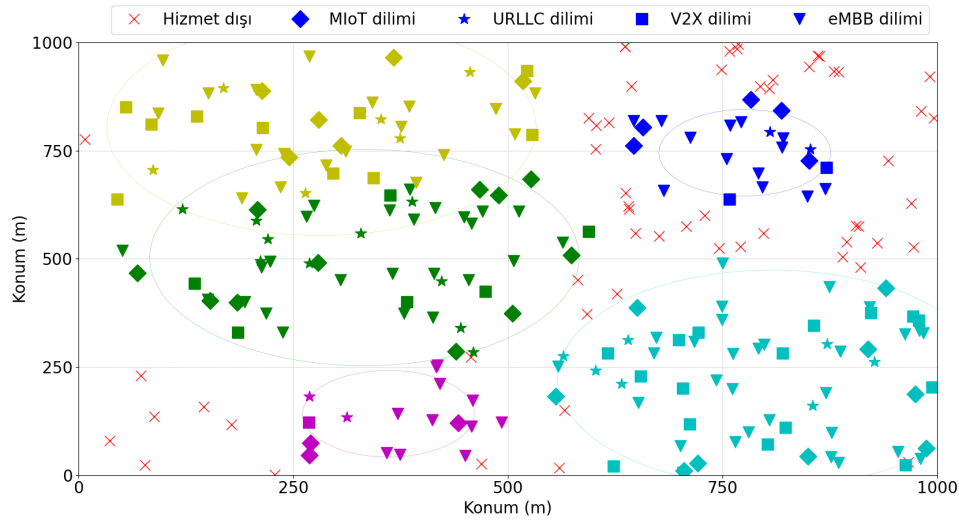
for  $i = 0$  to  $N$  do
  if  $\max(\mathbf{h}_i) > h_{i,\mathbf{a}[i]}$  then
     $\mathbf{a}[i] = \arg \max(\mathbf{h}_i)$ 
  else
    İlişkiyi değiştirme
  end if
end for
return  $\mathbf{a}$ 

```

6. BENZETİM ÇALIŞMALARINI VE SONUÇLAR

Tez kapsamında oluşturulan farklı ML modellerinin ve sistem mekanizmalarının test edilebilmesi için bir benzetim ortamı oluşturulması amaçlanmıştır. Bu ortam 1 km × 1 km kentsel bir arazi olarak varsayılmıştır. Ortamda 250 kullanıcı ve 5 baz istasyonu yer alması öngörülmüştür. Çalışmanın amacı ve zorlu kısmı ağ dilimleme olduğu için, ağ dilimleme için dört farklı ML modeli oluşturulmuştur. Ayrıca, ağ dilimleme başarısını artırmak amacıyla dört modelin ortalama kararı ve karar çoğunluğuna göre çalışan ağ dilimi sınıflandırma mekanizmaları da geliştirilmiştir.

Oluşturulan ML modellerinin ve ek mekanizmaların test edildiği, 250 kullanıcı ve 5 baz istasyonundan oluşan benzetim ortamı Şekil 6.1’de görülmektedir.



Şekil 6.1 Benzetim ortamındaki kullanıcılar ve kullanıcıların ağ dilimleri.

Şekil 6.1’de görülen benzetim ortamındaki kullanıcıların renkleri bağlı oldukları baz istasyonunu belirtirken, kullanıcı sembolleri atandıkları ağ dilimini belirtmektedir. ◆, ★, ■ ve ▼ sembolleri sırasıyla MIoT dilimini, URLLC dilimini, V2X dilimini ve eMBB dilimini belirtmektedirler.

Tez çalışması sırasında Vodafone Telekomünikasyon A.Ş. ile yapılan görüşmeler sonucunda kullanıcı verilerinde hata payları oluşabileceğinden bahsedilmiştir. Kullanıcıların kaynak tiplerine göre bu hata payının değişebileceği ancak hassas durumlarda bile bir hata payının olabileceği belirtilmiştir. Bu sebeple, benzetim çalışmaları hata payı olmayan ve hata payı olan veriler şeklinde iki kolda yürütülüp, her iki senaryoda da ML modellerinin başarıları değerlendirilmiştir.

Modellerin başarıları değerlendirilirken çeşitli ölçüler kullanılmıştır. Bunlar kesinlik, duyarlılık ve F1-Skoru olarak belirlenmiştir. Bu ölçüler sınıflandırma modellerinin çeşitli bakış açıları ile başarılarının gözlemlenmesini sağlamaktadır. Ölçüler hesaplanırken kullanılan hata matrisi Şekil 6.2’de görülmektedir.

		Ön Görülen Durum	
		Pozitif	Negatif
Gerçek Durum	Toplam Durumlar		
	Pozitif	TP	FN
	Negatif	FP	TN

Şekil 6.2 Hata matrisi.

Şekil 6.2’de görülen doğru pozitif (TP) gerçek durumda pozitif iken model tarafından da pozitif sınıflandırılan, doğru negatif (TN) gerçek durumda negatif iken model tarafından da negatif sınıflandırılan, yanlış pozitif (FP) gerçek durumda negatif iken model tarafından pozitif sınıflandırılan, yanlış negatif (FN) ise gerçek durumda pozitif iken model tarafından negatif olarak sınıflandırılan verilerdir. Modellerin başarılarını ölçen kesinlik, duyarlılık ve F1-skoru ölçüleri hata matrisindeki bu değerlerden hesaplanmaktadır [74].

Kesinlik, model tarafından pozitif olarak sınıflandırılan verilerin gerçekten kaç tanesinin pozitif olduğunu belirten ölçüdür. Pozitif ön görülenlerin doğruluk oranıdır. Formülü Denklem 16’da görülmektedir.

$$Kesinlik = \frac{TP}{TP + FP} \quad (16)$$

Duyarlılık, gerçekten pozitif olan verilerin kaç tanesinin pozitif olarak sınıflandırıldığını belirten ölçüdür. Ön görülen pozitiflerin gerçek pozitiflere oranıdır. Formülü Denklem 17’de görülmektedir.

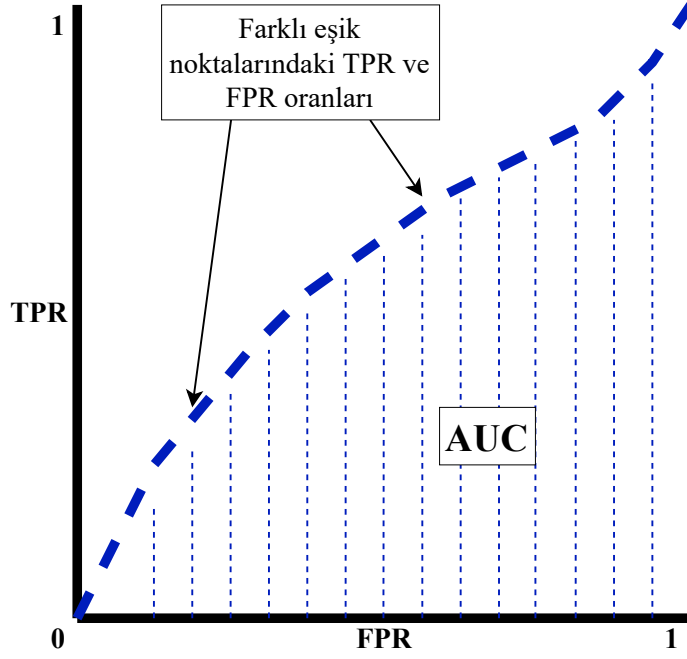
$$Duyarlılık = \frac{TP}{TP + FN} \quad (17)$$

F1-Skoru, duyarlılık ve kesinlik değerlerinden faydalanılarak üretilen bir ölçüdür. Bu iki değerın harmonik ortalaması F1-Skoru belirtmektedir. F1-Skoru, bir modelin genel doğruluğunu belirlemek için kullanılan en genel kapsamlı ölçüdür [75]. Formülü Denklem 18’de görülmektedir.

$$F1 = \frac{2 \times Duyarlilik \times Kesinlik}{Duyarlilik + Kesinlik} \quad (18)$$

Başarı ölçüleri hesaplanırken ağırlıklı ortalama ve makro ortalama baz alınmıştır. Ağırlıklı ortalama sınıfların örnek sayısına göre ağırlıklandırılıp ona göre ortalama alınması işlemine denmektedir [76]. Benzetim çalışmasında kullanılan kullanıcı verisinin sınıflarının dağılımı dengeli olmayacağından ağırlıklı ortalama kullanılmıştır. Makro ortalama ise örnek sayısına bakılmaksızın, sınıfların başarılarının ortalaması alınmaktadır.

Model başarımlı sonuçları belirtilen ölçü değerleri tablosu ve işlem karakteristik eğrisi (Receiver Operating Characteristic curve, ROC curve) kullanılarak görselleştirilmiştir. ROC eğrisi modellerin farklı sınıflandırma eşiklerinde nasıl başarımlı verdiğini gösteren bir grafikdir. Şekil 6.3’te örnek bir ROC eğrisi görülmektedir.



Şekil 6.3 Örnek ROC eğrisi.

Şekil 6.3'te yer alan grafikteki TPR doğru pozitiflerin tüm pozitif örnek sayısına oranını belirtirken, FPR yanlış pozitiflerin tüm negatiflere olan oranını belirtmektedir. Bu değerlerin formül ile gösterimi Denklem 19'da görülmektedir. AUC ise eğri altında kalan alanı (Area Under Curve, AUC) ifade etmektedir. Bir modelin ROC eğrisi altında kalan alanın büyüklüğü, modelin başarısını belirtmektedir. AUC ne kadar büyük ise, model o kadar başarılı çalışıyor demektir [77].

$$\begin{aligned}
 TPR &= \frac{TP}{TP + FN} \\
 FPR &= \frac{FP}{FP + TN}
 \end{aligned}
 \tag{19}$$

Benzetim çalışmalarında hata payı içermeyen ve hata payı içeren kullanıcı verileri senaryoları yer almaktadır. Her bir senaryo içerisinde çalışmada kullanılan ML modellerinin ve tüm ML modellerinin karar çoğunluğuna göre davranan bir ana ağ dilimi karar mekanizmasının başarıları incelenmiştir. İnceleme işlemi her iki senaryoda da yer almıştır.

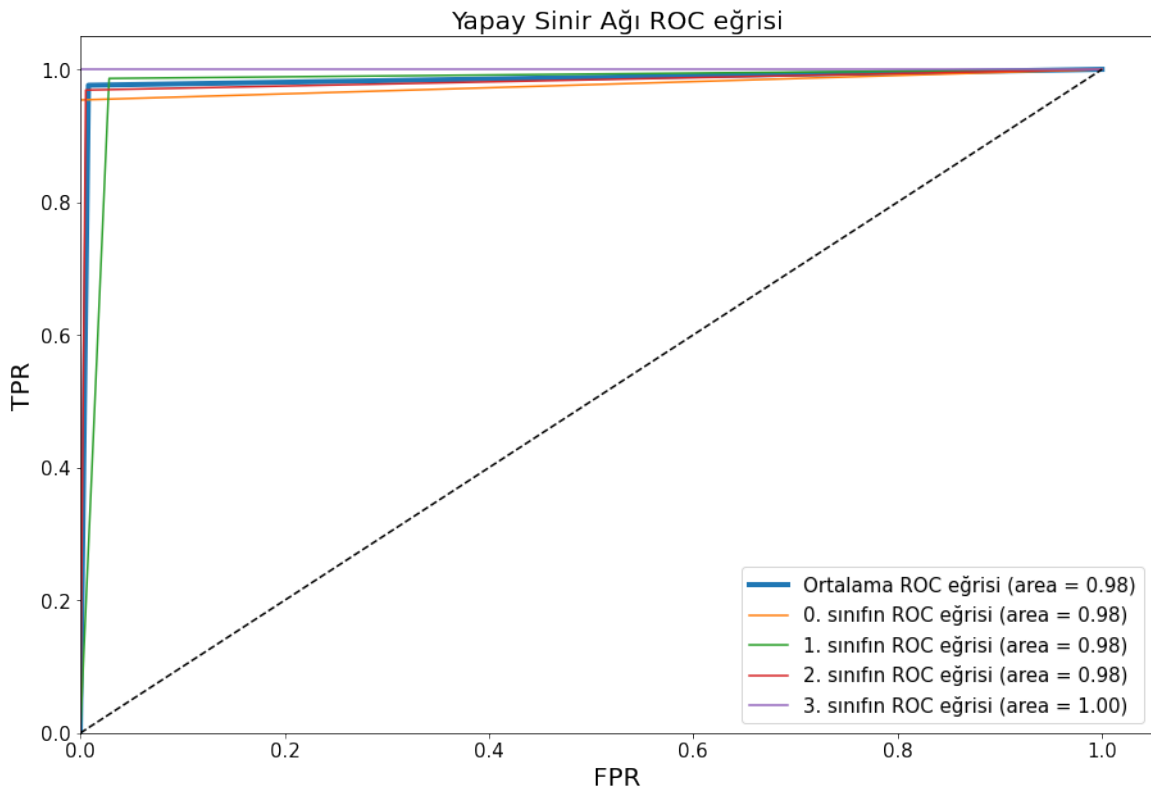
6.1. Hata Payı İçermeyen Kullanıcı Verileri

Bu kısımda hata payı eklenmeden oluşturulmuş kullanıcı verileri ile model başarımları değerlendirilmiştir. Hata payı eklenmemiş kullanıcı verisi, eğitim veri setindeki özelliklerin aynısını taşıırken, benzer sayısal sınırlara uyularak oluşturulmuştur.

İdeal koşullar altında kullanıcı verisi sınıflandırılmak üzere elde edilirken, veride bir hata payı olması beklenmemektedir. Bu sebeple çalışma içerisine hata payı içermeyen kullanıcı verisi ideal senaryo olarak eklenmiştir.

6.1.1. Yapay Sinir Ağı

Oluşturulan ANN modeli ile hata payı içermeyen veri ile elde edilen başarı sonuçları Şekil 6.4'te ve Tablo 6.1'de görülmektedir.



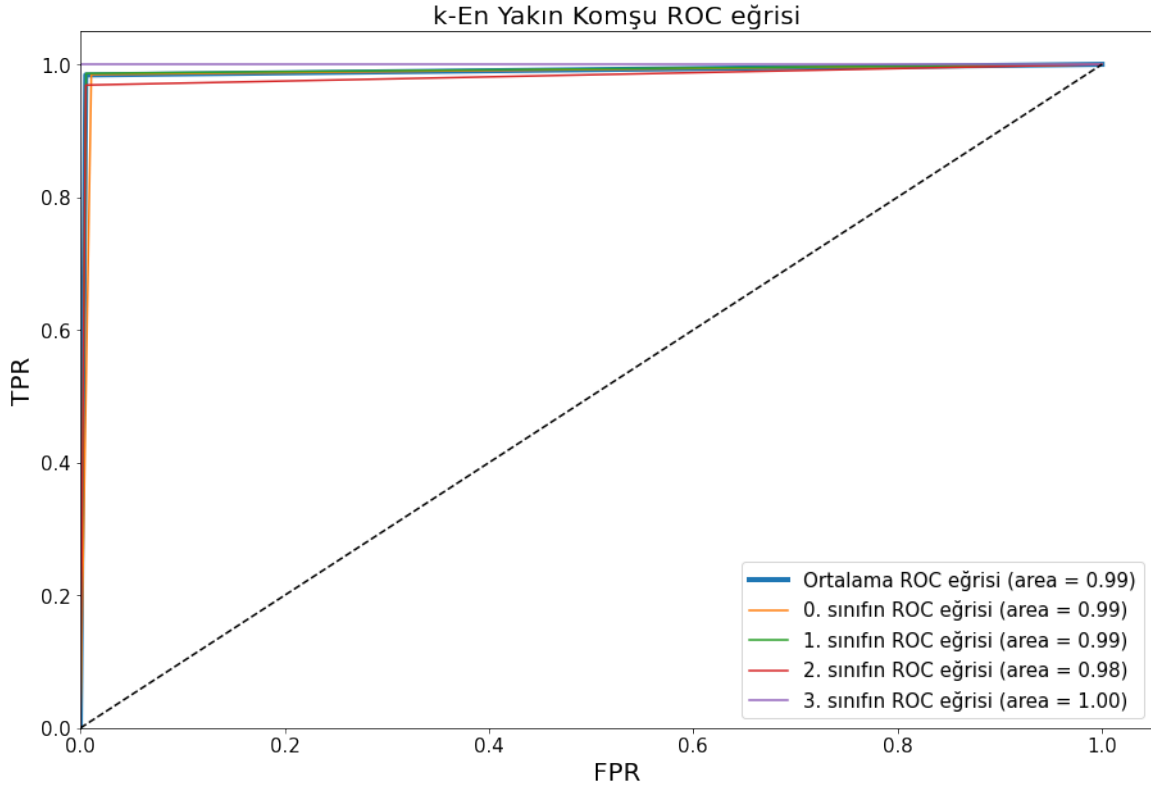
Şekil 6.4 Hata payı olmayan veri ile yapay sinir ağı ROC eğrisi.

Grafikte yer alan etiketlerden *0. sınıf* eMBB dilimini, *1. sınıf* MlO T dilimini, *2. sınıf* V2X dilimini ve *3. sınıf* URLLC dilimini temsil etmektedir.

ROC eğrisinde görüldüğü üzere, ANN modeli hata payı olmayan verilerin her sınıfında üstün başarılı bir sınıflandırma başarımı vermiştir.

6.1.2. k-En Yakın Komşu

Oluşturulan k-NN modeli ile hata payı içermeyen veri ile elde edilen başarı sonuçları Şekil 6.5'te ve Tablo 6.1'de görülmektedir.

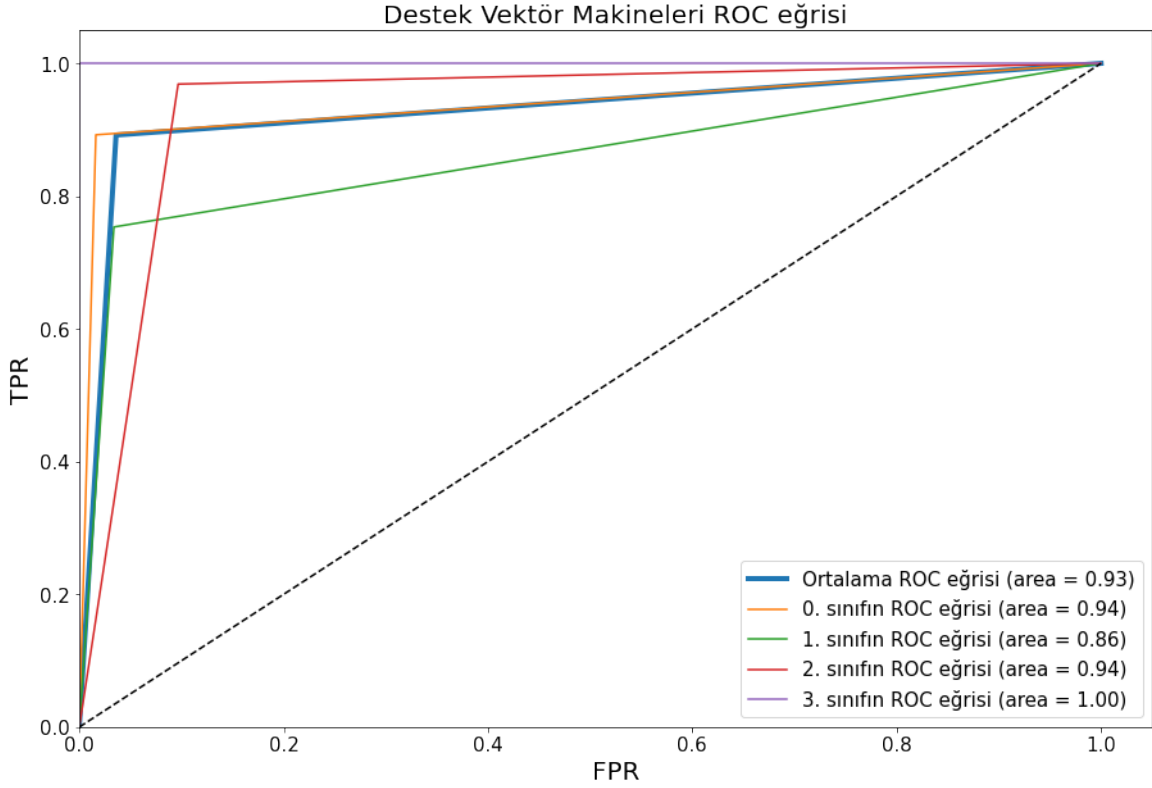


Şekil 6.5 Hata payı olmayan veri ile k-En yakın komşu ROC eğrisi.

Grafikte yer alan etiketlerden *0. sınıf* eMBB dilimini, *1. sınıf* MlO T dilimini, *2. sınıf* V2X dilimini ve *3. sınıf* URLLC dilimini temsil etmektedir. ROC eğrisinden görüldüğü üzere, k-NN modeli sınırları belli olan ve hata payı içermeyen çoklu sınıflandırma görevini başarıyla gerçekleştirmiştir.

6.1.3. Destek Vektör Makineleri

Oluşturulan SVM modeli ile hata payı içermeyen veri ile elde edilen başarı sonuçları Şekil 6.6'da ve Tablo 6.1'de görülmektedir.



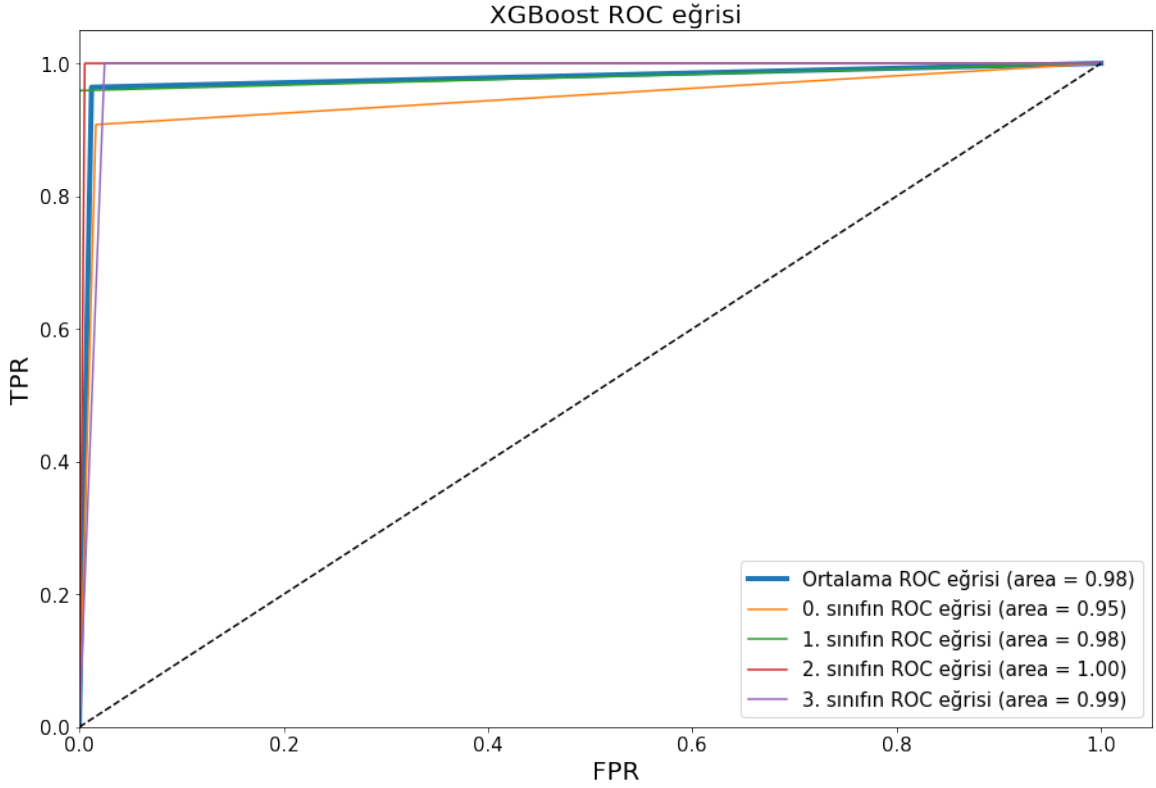
Şekil 6.6 Hata payı olmayan veri ile destek vektör makineleri ROC eğrisi.

Grafikte yer alan etiketlerden *0. sınıf* eMBB dilimini, *1. sınıf* MIoT dilimini, *2. sınıf* V2X dilimini ve *3. sınıf* URLLC dilimini temsil etmektedir.

ROC eğrisi incelendiğinde, SVM modeli ile başarılı bir sınıflandırma başarımı elde edildiği görülmektedir. SVM modelleri sınıflandırma görevleri için sıklıkla kullanılırsa da veri içerisindeki sınıf sayısının artmasıyla başarımları düşmektedir. Kullanılan veri içerisindeki sınıf sayısı çok yüksek olmadığından SVM modeli başarı göstermiş olsa da ikili sınıflandırma gibi düşük sayılı bir sınıflandırma yapılmadığından sonuç etkilenmiştir.

6.1.4. XGBoost - eXtreme Gradient Boosting (Aşırı Gradyan Artırma)

Oluşturulan XGBoost modeli ile hata payı içermeyen veri ile elde edilen başarı sonuçları Şekil 6.7’de ve Tablo 6.1’de görülmektedir.



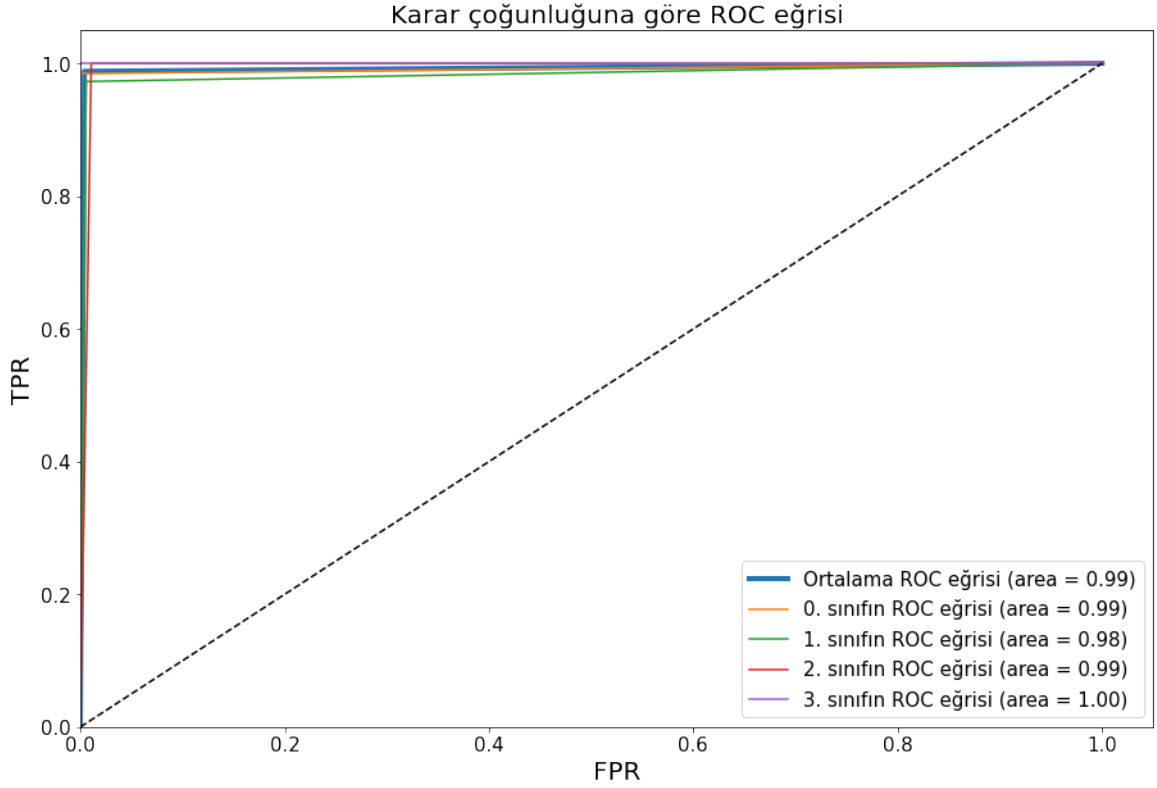
Şekil 6.7 Hata payı olmayan veri ile XGBoost ROC eğrisi.

Grafikte yer alan etiketlerden *0. sınıf* eMBB dilimini, *1. sınıf* MIoT dilimini, *2. sınıf* V2X dilimini ve *3. sınıf* URLLC dilimini temsil etmektedir.

XGBoost, bu gibi çalışmalarda kullanılmakta olan çizelge şeklindeki veriler ile üstün başarı gösterdiğinden dolayı bu çalışmada tercih edilmiştir. ROC eğrisinde de görüldüğü üzere başarılı bir sınıflandırma başarımı elde etmiştir.

6.1.5. Karar Çoğunluğu Mekanizması

Oluşturulan karar çoğunluğu modeli ile hata payı içermeyen veri ile elde edilen başarı sonuçları Şekil 6.8’de ve Tablo 6.1’de görülmektedir.



Şekil 6.8 Hata payı olmayan veri ile karar çoğunluğu ROC eğrisi.

Grafikte yer alan etiketlerden *0. sınıf* eMBB dilimini, *1. sınıf* MIoT dilimini, *2. sınıf* V2X dilimini ve *3. sınıf* URLLC dilimini temsil etmektedir.

Karar çoğunluğu modeli oluşturulurken belirlenen hedef en başarılı sınıflandırma modelini elde etmek olmuştur. Bunun için tüm modellerin her bir kullanıcı için çıktısının çoğunluğuna bakılıp, son bir karar verilmiştir. Hata payı içermeyen kullanıcı veri senaryosunda, istenilen hedefe ulaşılmıştır. Modeller tek başlarına da yüksek başarımlar göstermiş olsalar da karar çoğunluğu mekanizması en yüksek başarımları göstermeyi başarmıştır.

Tablo 6.1 Hata payı içermeyen veri ile model başarımları.

	Ağırlıklı Ortalama			Makro Ortalama		
	Duyarlılık	Kesinlik	F1-Skoru	Duyarlılık	Kesinlik	F1-Skoru
Yapay Sinir Ağı	0.98	0.98	0.98	0.98	0.98	0.98
k-En Yakın Komşu	0.98	0.98	0.98	0.98	0.99	0.98
Destek Vektör Makineleri	0.89	0.90	0.89	0.90	0.91	0.90
XGBoost	0.96	0.97	0.96	0.97	0.96	0.96
Karar Çoğunluğu	0.99	0.99	0.99	0.99	0.99	0.99

Hata payı içermeyen kullanıcı verisi, modellerin eğitildiği veri ile yakın benzerlik göstermektedir. Bu sebeple modellerin başarılarının yüksek olması beklenmiştir. Tablo 6.1'de görüldüğü üzere, tekli modeller içerisindeki en yüksek başarıları ANN ve k-NN modelleri elde etmişlerdir. Ayrıca modellerin karar çoğunluğu ile oluşturulan mekanizmanın en başarılı sınıflandırma modeli olduğu görülmüştür.

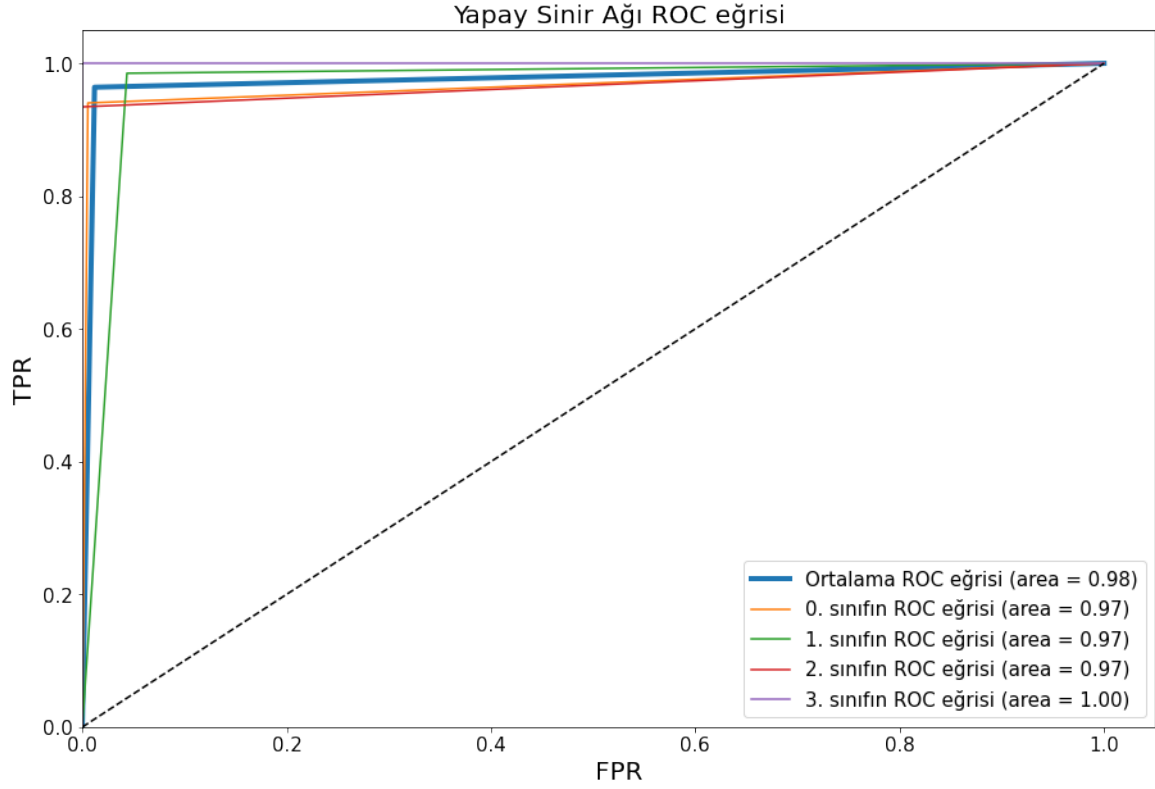
6.2. Hata Payı İçeren Kullanıcı Verileri

Gerçeğe yakın senaryolarda kullanıcı verisi elde edilirken hata payı söz konusu olduğundan dolayı, bu kısımda hata payı eklenerek oluşturulmuş kullanıcı verileri ile model başarımları değerlendirilmiştir. Vodafone Telekomünikasyon A.Ş. ile yapılan görüşmeler ışığında, bahsedilen hata payı kaynak tipi *gecikme kritik* olan kullanıcılar için %5 olarak, kaynak tipi *gecikme kritik* olmayan kullanıcılar için %20 olarak belirlenmiştir.

Kullanıcı verisine hata payı eklemek, oluşturulan modelleri gerçeğe yakın senaryolarda değerlendirme imkanı sağlarken, ML modellerinin veri üzerindeki değişimlere de nasıl tepki vereceklerinin görülmesini sağlamaktadır.

6.2.1. Yapay Sinir Ağı

Oluşturulan ANN modeli ile hata payı içeren veri ile elde edilen başarı sonuçları Şekil 6.9'da ve Tablo 6.2'de görülmektedir.



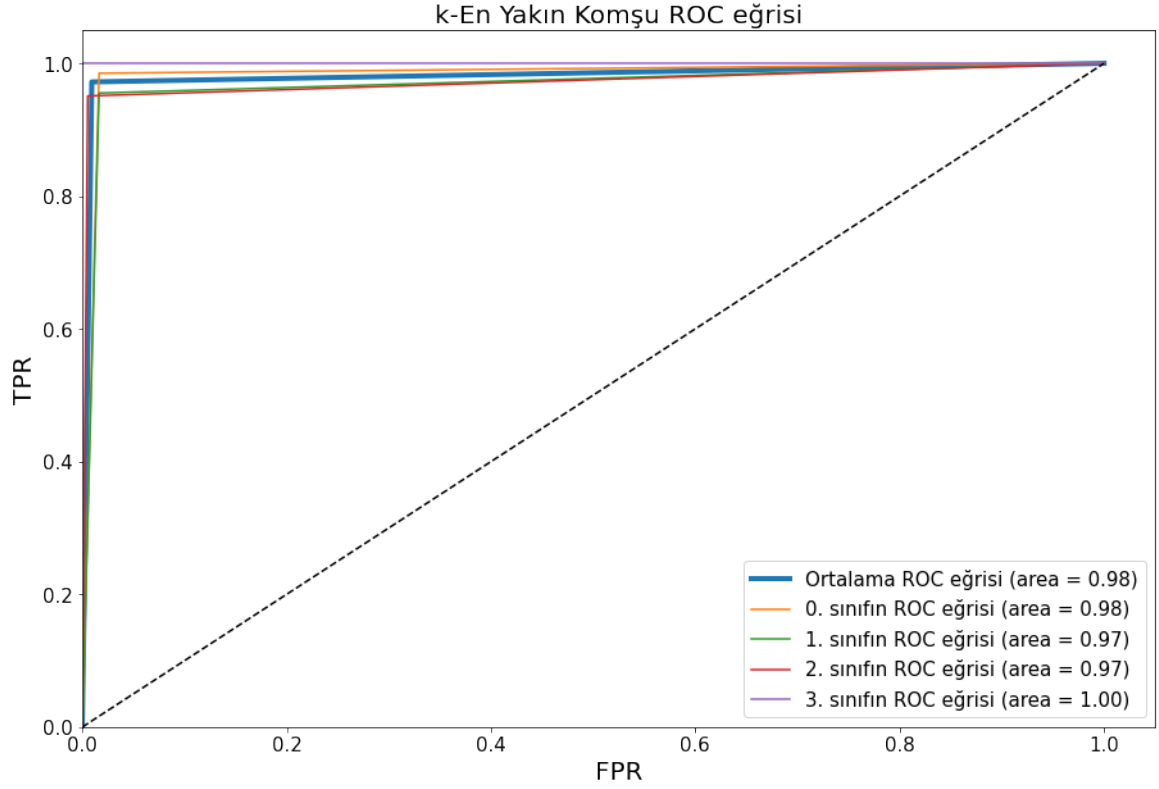
Şekil 6.9 Hata paylı veri ile yapay sinir ağı ROC eğrisi.

Grafikte yer alan etiketlerden *0. sınıf* eMBB dilimini, *1. sınıf* MIoT dilimini, *2. sınıf* V2X dilimini ve *3. sınıf* URLLC dilimini temsil etmektedir.

ANN modeli esnek ve kolay adapte olabilen yapıya sahip olduğundan, verideki değişime başarılı bir şekilde tepki vermiştir.

6.2.2. k-En Yakın Komşu

Oluşturulan k-NN modeli ile hata payı içeren veri ile elde edilen başarı sonuçları Şekil 6.10'da ve Tablo 6.2'de görülmektedir.



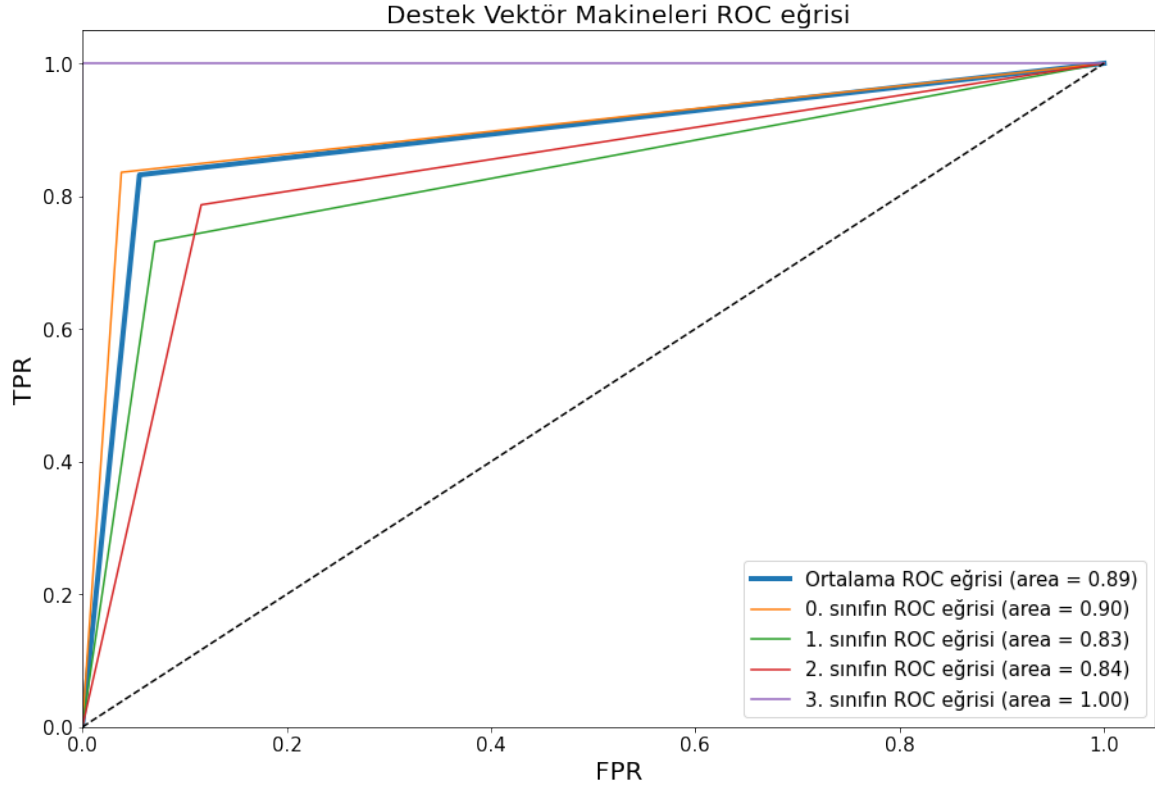
Şekil 6.10 Hata paylı veri ile k-En yakın komşu ROC eğrisi.

Grafikte yer alan etiketlerden *0. sınıf* eMBB dilimini, *1. sınıf* MIoT dilimini, *2. sınıf* V2X dilimini ve *3. sınıf* URLLC dilimini temsil etmektedir.

k-NN modeli sınırlar bazından çalışmaktadır. Bu sebeple, hata payının kabul edilebilir olduğu senaryolarda yüksek başarıyı vermeye devam edebilmektedir.

6.2.3. Destek Vektör Makineleri

Oluşturulan SVM modeli ile hata payı içeren veri ile elde edilen başarı sonuçları Şekil 6.11'de ve Tablo 6.2'de görülmektedir.



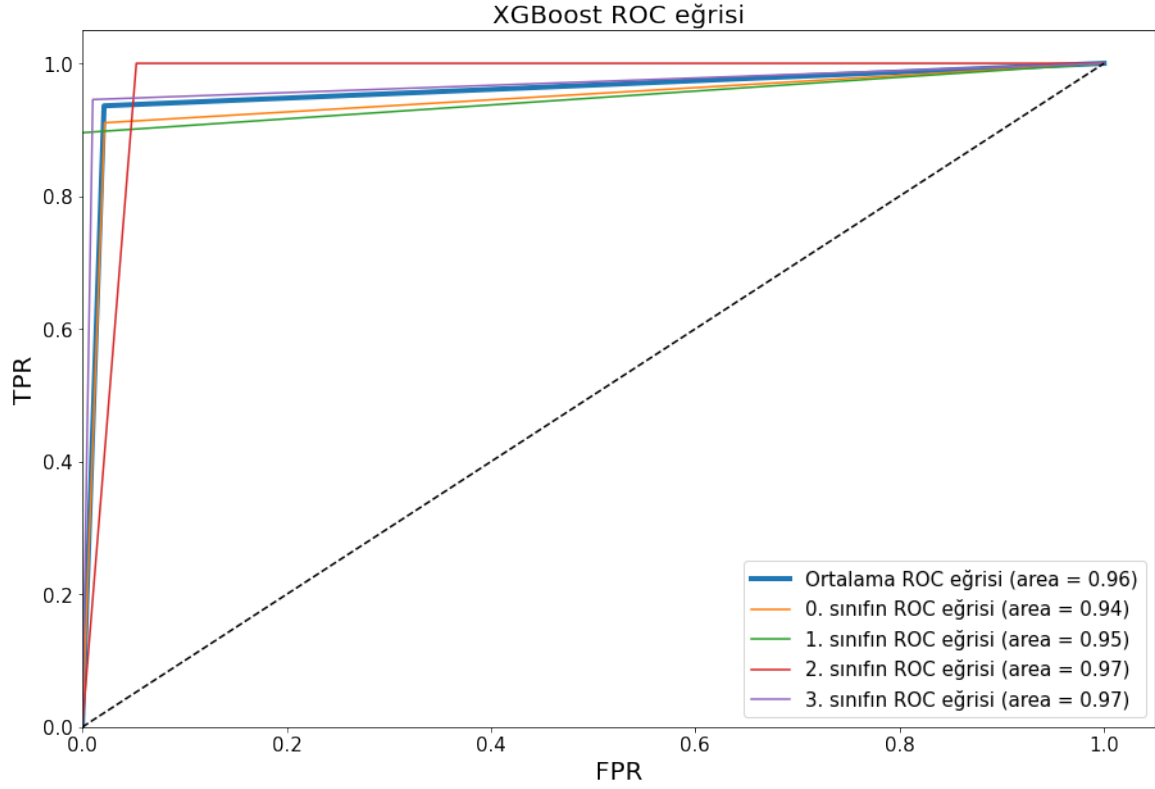
Şekil 6.11 Hata paylı veri ile destek vektör makineleri ROC eğrisi.

Grafikte yer alan etiketlerden *0. sınıf* eMBB dilimini, *1. sınıf* MIoT dilimini, *2. sınıf* V2X dilimini ve *3. sınıf* URLLC dilimini temsil etmektedir.

SVM modeli düşük sayıda sınıfa sahip olmayan veriler üzerinde daha düşük başarı gösterdiğinden, hata payı artışı en çok SVM modelini etkilemektedir.

6.2.4. XGBoost - eXtreme Gradient Boosting (Aşırı Gradyan Artırma)

Oluşturulan XGBoost modeli ile hata payı içeren veri ile elde edilen başarı sonuçları Şekil 6.12'de ve Tablo 6.2'de görülmektedir.



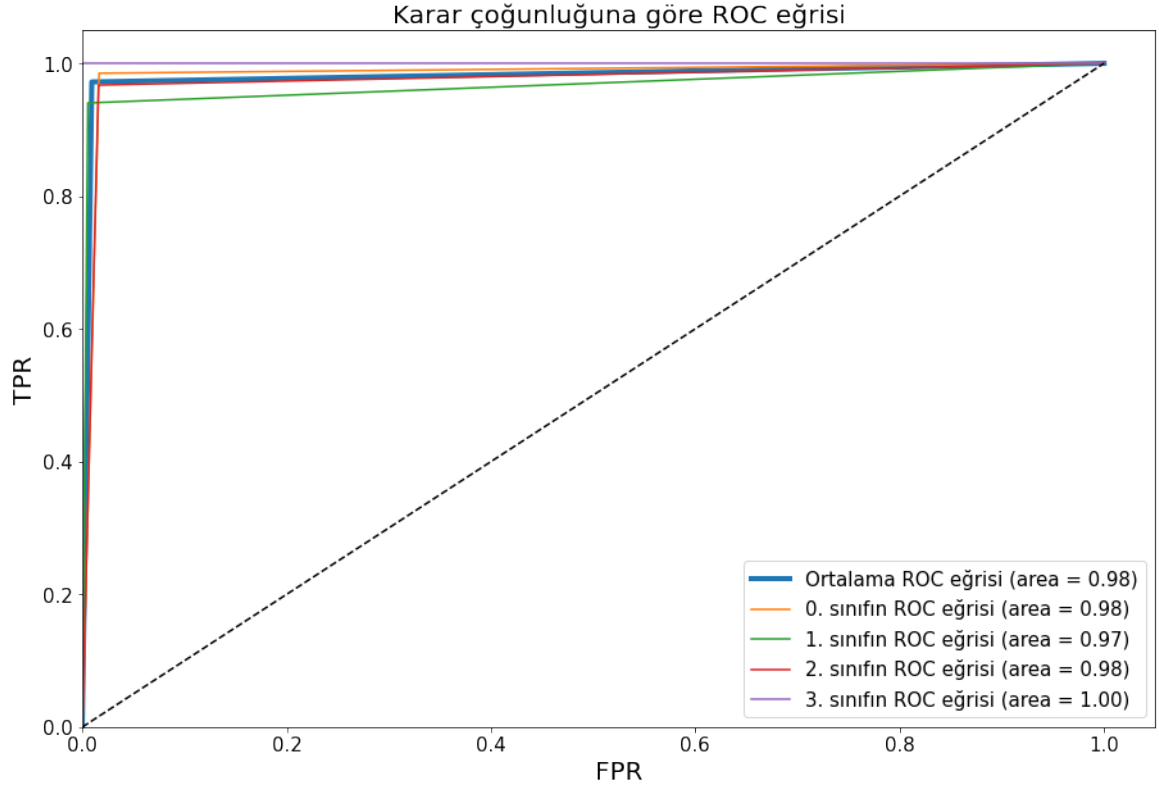
Şekil 6.12 Hata paylı veri ile XGBoost ROC eğrisi.

Grafikte yer alan etiketlerden *0. sınıf* eMBB dilimini, *1. sınıf* MIoT dilimini, *2. sınıf* V2X dilimini ve *3. sınıf* URLLC dilimini temsil etmektedir.

Çizelge veriler üzerinde yüksek başarımlı gösterebilen XGBoost modeli, hata paylı kullanıcı verisi karşısında başarımlı değerlerini en başarılı koruyan ML modeli olmuştur.

6.2.5. Karar Çoğunluğu Mekanizması

Oluşturulan karar çoğunluğu modeli ile hata payı içermeyen veri ile elde edilen başarı sonuçları Şekil 6.13'te ve Tablo 6.1'de görülmektedir.



Şekil 6.13 Hata paylı veri ile karar çoğunluđu ROC eđrisi.

Grafikte yer alan etiketlerden *0. sınıf* eMBB dilimini, *1. sınıf* MIoT dilimini, *2. sınıf* V2X dilimini ve *3. sınıf* URLLC dilimini temsil etmektedir.

Hata payı içermeyen kullanıcı verisinde olduđu gibi, hata paylı kullanıcı verisi senaryosunda da karar çoğunluđu mekanizması en yüksek başarıyı göstermeyi başarmıştır. Tekli ML modellerinin başarısı yüksek olsa da mekanizmanın hızlı çalışabiliyor olması karar çoğunluđu mekanizmasının kullanılmasını mantıklı kılmaktadır.

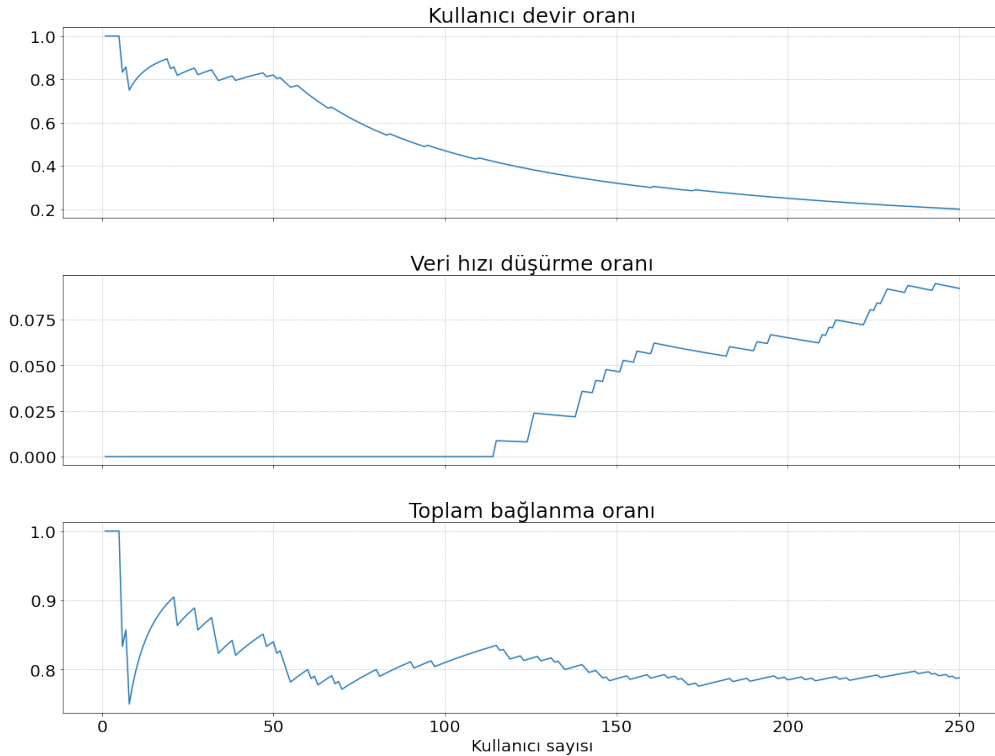
Tablo 6.2 Hata payı içeren veri ile model başarımları.

	Ađırlıklı Ortalama			Makro Ortalama		
	Duyarlılık	Kesinlik	F1-Skoru	Duyarlılık	Kesinlik	F1-Skoru
Yapay Sinir Ađı	0.96	0.97	0.96	0.96	0.97	0.97
k-En Yakın Komşu	0.97	0.97	0.97	0.97	0.97	0.97
Destek Vektör Makineleri	0.83	0.84	0.83	0.84	0.84	0.84
XGBoost	0.97	0.97	0.97	0.97	0.97	0.97
Karar Çođunluđu	0.98	0.98	0.98	0.98	0.98	0.98

Hata payı içeren kullanıcı verisi, modellerin eğitildiği veri ile farklılıklar içermektedir. Bu sebeple, modellerin hata payı olmayan kullanıcılara göre daha düşük bir başarıml göstermesi beklenmiştir. Tablo 6.2’de görüldüğü üzere, tekli modeller içerisindeki ANN, k-NN ve XGBoost yeni veriye başarılı bir şekilde adapte olurken, SVM diğer modeller kadar adapte olamamıştır. Ayrıca modellerin karar çoğunluğu ile oluşturulan mekanizmanın önceki senaryoda da olduğu gibi en başarılı sınıflandırma modeli olduğu görülmüştür.

6.3. Benzetim Ortamı İstatistikleri

Benzetim çalışması süresince ağ dilimlemenin yanı sıra sisteme dahil olan diğer mekanizmalar da çalışmıştır. Kullanıcılara kesintisiz hizmet verilebilmesi için kullanıcı devri ve veri hızı düşürme mekanizmaları geliştirilmiştir. Bu mekanizmaların aktifliğini denetleyebilmek amacıyla, benzetim süresince kullanıcı istatistikleri tutulmuştur. Şekil 6.14’te benzetim boyunca gerçekleşen kullanıcı devri, veri hızı düşürme ve kullanıcı bağlanma değerlerinin toplam kullanıcı sayısına olan oranları görülmektedir.



Şekil 6.14 Gerçekleşen ek mekanizmaların istatistik değerleri.

Şekil 6.14'te görüldüğü gibi kullanıcı devir oranı benzetim başladığında yüksek bir oranda iken, sonrasında rastgele olan verilerin sayısının artmasıyla birlikte beklenen oranlara gelmiştir.

Veri hızı düşürme mekanizması baz istasyonlarının kapasitesi belli bir değerin altına düştüğünde devreye girdiğinden dolayı benzetimin ilk yarısında devreye girmemiştir. Sonrasında devreye girip, kullanım süresi dolan ve yeniden bağlanan kullanıcı dolaşımından etkilenerek dalgalı bir değişim göstermiştir.

Toplam bağlanma ise benzetim boyunca %80-%85 aralığında olup, toplam kullanıcı sayısına oranla beklenen bir oran göstermektedir.

7. SONUÇ VE TARTIŞMA

Bu tez çalışmasında, içerisinde kullanıcılar ve baz istasyonları bulunan, gerçekçi bir 5G benzetim ortamı oluşturulması ve ML kullanılarak kullanıcıların en uygun 5G ağ dilimine yerleştirilmesi amaçlanmıştır. Ağ dilimleme işlemi için çeşitli ML modelleri oluşturup bu modellerin başarımlarını karşılaştırılması yapılmasının yanı sıra, benzetim ortamının işleyişini daha sağlıklı hale getirmek için kullanıcı devri ve kullanıcı veri hızı düşürülmesi gibi ek mekanizmalar da geliştirilmesi hedeflenmiştir.

Bu çalışmada 3GPP kaynakları baz alınarak, benzetim ortamında yer alacak olan ağ dilimleme sınıflandırılması yapılacak kullanıcılar ve baz istasyonları oluşturulmuştur. Sonrasında benzetim çalışması içerisinde yer alan kullanıcı-baz istasyonu ilişkilendirilme mekanizması, kullanıcı devir mekanizması, kullanıcı veri hızı düşürme mekanizması ve ağ dilimleme mekanizmaları oluşturulmuştur.

Veri hızı düşürme mekanizması ve ağ dilimleme mekanizmaları ortak eğitim verisini kullanan ML modelleri içermektedirler. Veri hızı düşürme mekanizmasında RF modeli ile kullanım tipi sınıflandırılması yapılması, ağ dilimleme mekanizmasında ise dört farklı ML modeli kullanılıp modellerin karar çoğunluğuna göre sınıflandırma yapılması ve ML modellerinin başarımlarını karşılaştırılması yapılması hedeflenmiştir.

Çalışmanın ana hedefi ağ dilimlemesi olduğundan, benzetim çalışmalarında ağ dilimlemesine odaklanılmıştır. Bu noktada, çalışmanın gerçekçi olması adına hem ideal hem de gerçeğe yakın kullanıcılar oluşturulup iki farklı senaryo ile ML modelleri değerlendirilmiştir. İdeal senaryoda modeller eğitim verisine benzer kullanıcı verisi ile, gerçeğe yakın senaryoda ise modeller hata payı eklenmiş kullanıcı verisi ile test edilmiştir. Kullanıcı verisine hata payı eklenmiş olması, çalışmayı gerçeğe yakın kılarken aynı zamanda ML modellerinin başarımlarının da daha detaylı şekilde karşılaştırılmasını sağlamıştır. Benzetim çalışması kapsamında modellerin kesinlik, duyarlılık ve F1-Skorları incelenmiştir. Bu kapsamda modeller genel olarak başarılı sonuçlar elde etmiş, ancak aralarında yüksek başarı gösteren modeller olduğu görülmüştür. Çalışmada kullanılan

çizelge tipi verilere daha uygun olan ML modelleri, hata payına sahip veri ile test edilirken düşük başarımları kayıplarını yaşarlarken, yapısı daha farklı olan modeller farklı başarımları kayıplarını yaşamışlardır. Bu farklılık, çalışma içerisindeki ML modellerinin yapısına ve çalışma şekillerine örnek olmaktadır.

Bu tez kapsamında yapılan katkılar; 3GPP standartlarına uygun ve gerçeğe yakın durumları dikkate alan bir benzetim ortamı oluşturulması, dört farklı ML modeli ile ağ dilimlemesi yapılarak, karar yoğunluğuna bakılarak bir ana karar mekanizması oluşturulması, kullanıcı hareketlerine göre değişen kanal parametresi ile gereken durumlarda kullanıcı-baz istasyonu devri gerçekleştirilmesi, baz istasyonlarının kapasitesinin belli bir eşiğin altına düşmesiyle birlikte RF modeli içeren kullanıcı veri hızı düşürme mekanizması ile veri hızı düşürülmesi şeklinde özetlenebilmektedir.

Son olarak, bu çalışmayı geliştirebilecek gelecek çalışmalar düşünülmektedir. Baz istasyonlarının özelliklerinin belirlenmesinde kullanılan geçmiş kullanıcı verileri, farklı amaçlarla da kullanılabilir. Geçmiş kullanıcı verileri yardımıyla hangi zaman diliminde hangi ağ dilimi için ne kadar kapasite gerekeceği tahmini mekanizması ile, benzetim çalışmasının daha sorunsuz geçirilmesi gerçekleştirilebilir. Bunun için hafızalı ML modelleri kullanılması ve isabetli tahminler yapılması gerekmektedir, ancak yapılan çalışmanın devamı niteliğinde olabilecek bir çalışma potansiyelindedir.

KAYNAKLAR

- [1] Ericsson mobility report. ericsson.com/mobility-report, **2021**.
- [2] 3GPP TS 23.501 V17.0.0. System architecture for the 5G system. *3GPP*, **2021**.
- [3] Jon Matias, Jokin Garay, Nerea Toledo, Juanjo Unzilla, and Eduardo Jacob. Toward an SDN-enabled NFV architecture. *IEEE Communications Magazine*, 53(4):187–193, **2015**.
- [4] Eugen Borcoci, Ana-Maria Drăgulescu, Frank Y Li, Marius-Constantin Vochin, and Kjetil Kjellstadli. An overview of 5G slicing operational business models for internet of vehicles, maritime IoT applications and connectivity solutions. *IEEE Access*, **2021**.
- [5] Anju Uttam Gawas. An overview on evolution of mobile wireless communication networks: 1G-6G. *International Journal on Recent and Innovation Trends in Computing and Communication*, 3(5):3130–3133, **2015**.
- [6] Hyung G Myung, Junsung Lim, and David J Goodman. Single carrier FDMA for uplink wireless transmission. *IEEE vehicular technology magazine*, 1(3):30–38, **2006**.
- [7] Randolph Nelson and Leonard Kleinrock. Spatial TDMA: A collision-free multihop channel access protocol. *IEEE Transactions on communications*, 33(9):934–944, **1985**.
- [8] Shinsuke Hara and Ramjee Prasad. Overview of multicarrier CDMA. *IEEE communications Magazine*, 35(12):126–133, **1997**.
- [9] Hujun Yin and Siavash Alamouti. OFDMA: A broadband wireless access technology. In *2006 IEEE sarnoff symposium*, pages 1–4. IEEE, **2006**.
- [10] Jerry R Hampton. *Introduction to MIMO communications*. Cambridge university press, **2013**.

- [11] *Everything you need to know about 5G*. Qualcomm.
- [12] Trinh Van Chien and Emil Björnson. Massive MIMO communications. In *5G mobile communications*, pages 77–116. Springer, **2017**.
- [13] Xenofon Foukas, Georgios Patounas, Ahmed Elmokashfi, and Mahesh K Marina. Network slicing in 5G: Survey and challenges. *IEEE communications magazine*, 55(5):94–100, **2017**.
- [14] Jose Ordonez-Lucena, Pablo Ameigeiras, Diego Lopez, Juan J Ramos-Munoz, Javier Lorca, and Jesus Folgueira. Network slicing for 5G with SDN/NFV: Concepts, architectures, and challenges. *IEEE Communications Magazine*, 55(5):80–87, **2017**.
- [15] Asvin Gohil, Hardik Modi, and Shobhit K Patel. 5G technology of mobile communication: A survey. In *2013 international conference on intelligent systems and signal processing (ISSP)*, pages 288–292. IEEE, **2013**.
- [16] Rupendra Nath Mitra and Dharma P Agrawal. 5G mobile technology: A survey. *Ict Express*, 1(3):132–137, **2015**.
- [17] Peter Rost, Albert Banchs, Ignacio Berberana, Markus Breitbach, Mark Doll, Heinz Droste, Christian Mannweiler, Miguel A Puente, Konstantinos Samdanis, and Bessem Sayadi. Mobile network architecture evolution toward 5G. *IEEE Communications Magazine*, 54(5):84–91, **2016**.
- [18] Harri Holma, Antti Toskala, and Takehiro Nakamura. *5G technology: 3GPP new radio*. John Wiley & Sons, **2020**.
- [19] Diego Kreutz, Fernando MV Ramos, Paulo Esteves Verissimo, Christian Esteve Rothenberg, Siamak Azodolmolky, and Steve Uhlig. Software-defined networking: A comprehensive survey. *Proceedings of the IEEE*, 103(1):14–76, **2014**.
- [20] *What is Software-Defined Networking SDN?* vmvare.

- [21] Bo Han, Vijay Gopalakrishnan, Lusheng Ji, and Seungjoon Lee. Network function virtualization: Challenges and opportunities for innovations. *IEEE communications magazine*, 53(2):90–97, **2015**.
- [22] *What is network functions virtualization?* vmware.
- [23] *5G Network Slicing*. Viavi Solutions.
- [24] Anurag Thantharate, Rahul Paropkari, Vijay Walunj, and Cory Beard. Deepslice: A deep learning approach towards an efficient and reliable network slicing in 5G networks. In *2019 IEEE 10th Annual Ubiquitous Computing, Electronics & Mobile Communication Conference (UEMCON)*, pages 0762–0767. IEEE, **2019**.
- [25] Guolin Sun, Gebrekidan Tesfay Zemuy, and Kun Xiong. Dynamic reservation and deep reinforcement learning based autonomous resource management for wireless virtual networks. In *2018 IEEE 37th International Performance Computing and Communications Conference (IPCCC)*, pages 1–4. IEEE, **2018**.
- [26] Marcos Toscano, Federico Grunwald, Matías Richart, Javier Baliosian, Eduardo Grampín, and Alberto Castro. Machine learning aided network slicing. In *2019 21st International Conference on Transparent Optical Networks (ICTON)*, pages 1–4. IEEE, **2019**.
- [27] Mustufa Haider Abidi, Hisham Alkhalefah, Khaja Moiduddin, Mamoun Alazab, Muneer Khan Mohammed, Wadea Ameen, and Thippa Reddy Gadekallu. Optimal 5G network slicing using machine learning and deep learning concepts. *Computer Standards & Interfaces*, 76:103518, **2021**.
- [28] Jie Mei, Xianbin Wang, and Kan Zheng. Intelligent network slicing for V2X services toward 5G. *IEEE Network*, 33(6):196–204, **2019**.
- [29] Akihiro Nakao and Ping Du. Toward in-network deep machine learning for identifying mobile applications and enabling application specific network slicing. *IEICE Transactions on Communications*, page 2017CQI0002, **2018**.

- [30] Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, et al. Scikit-learn: Machine learning in python. *the Journal of machine Learning research*, 12:2825–2830, **2011**.
- [31] François Chollet et al. Keras. <https://keras.io>, **2015**.
- [32] Fatih Isik, Gurkan Ozden, and Mehmet Kuntalp. Importance of data preprocessing for neural networks modeling: The case of estimating the compaction parameters of soils. *Energy Educ Sci Technol Part A: Energy Sci Res*, 29:463–74, **2012**.
- [33] Tanay Agrawal. Hyperparameter optimization using scikit-learn. In *Hyperparameter Optimization in Machine Learning*, pages 31–51. Springer, **2021**.
- [34] Payam Refaeilzadeh, Lei Tang, and Huan Liu. Cross-validation. *Encyclopedia of database systems*, 5:532–538, **2009**.
- [35] Salvador García, Julián Luengo, and Francisco Herrera. *Data preprocessing in data mining*, volume 72. Springer, **2015**.
- [36] Mauricio A Hernández and Salvatore J Stolfo. Real-world data is dirty: Data cleansing and the merge/purge problem. *Data mining and knowledge discovery*, 2(1):9–37, **1998**.
- [37] S Manikandan. Data transformation. *Journal of Pharmacology and Pharmacotherapeutics*, 1(2):126, **2010**.
- [38] Philip R Bevington and D Keith Robinson. Data reduction and error analysis. *McGraw-Hill, New York*, **2003**.
- [39] John Quackenbush. Microarray data normalization and transformation. *Nature genetics*, 32(4):496–501, **2002**.

- [40] Patricio Cerda, Gaël Varoquaux, and Balázs Kégl. Similarity encoding for learning with dirty categorical variables. *Machine Learning*, 107(8):1477–1494, **2018**.
- [41] Aristidis Likas, Nikos Vlassis, and Jakob J Verbeek. The global k-means clustering algorithm. *Pattern recognition*, 36(2):451–461, **2003**.
- [42] IBM. Neural networks. **Aug. 2020**.
- [43] Guoqiang Peter Zhang. Neural networks for classification: a survey. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 30(4):451–462, **2000**.
- [44] Simon Haykin. *Neural networks and learning machines, 3/E*. Pearson Education India, **2010**.
- [45] Sagar Sharma, Simone Sharma, and Anidhya Athaiya. Activation functions in neural networks. *towards data science*, 6(12):310–316, **2017**.
- [46] Abien Fred Agarap. Deep learning using rectified linear units (relu). *arXiv preprint arXiv:1803.08375*, **2018**.
- [47] Kaibo Duan, S Sathiya Keerthi, Wei Chu, Shirish Krishnaji Shevade, and Aun Neow Poo. Multi-category classification by soft-max combination of binary classifiers. In *International Workshop on Multiple Classifier Systems*, pages 125–134. Springer, **2003**.
- [48] Jiang Wang, Yi Yang, Junhua Mao, Zhiheng Huang, Chang Huang, and Wei Xu. CNN-RNN: A unified framework for multi-label image classification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2285–2294. **2016**.
- [49] Ji Young Lee and Franck Dernoncourt. Sequential short-text classification with recurrent and convolutional neural networks. *arXiv preprint arXiv:1603.03827*, **2016**.

- [50] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: a simple way to prevent neural networks from overfitting. *The journal of machine learning research*, 15(1):1929–1958, **2014**.
- [51] Dimitris Stathakis. How many hidden layers and nodes? *International Journal of Remote Sensing*, 30(8):2133–2147, **2009**.
- [52] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: a simple way to prevent neural networks from overfitting. *The journal of machine learning research*, 15(1):1929–1958, **2014**.
- [53] Zijun Zhang. Improved adam optimizer for deep neural networks. In *2018 IEEE/ACM 26th International Symposium on Quality of Service (IWQoS)*, pages 1–2. IEEE, **2018**.
- [54] Leif E Peterson. K-nearest neighbor. *Scholarpedia*, 4(2):1883, **2009**.
- [55] William S Noble. What is a support vector machine? *Nature biotechnology*, 24(12):1565–1567, **2006**.
- [56] Thomas G Dietterich et al. Ensemble learning. *The handbook of brain theory and neural networks*, 2(1):110–125, **2002**.
- [57] Vladimir Svetnik, Andy Liaw, Christopher Tong, J Christopher Culberson, Robert P Sheridan, and Bradley P Feuston. Random forest: a classification and regression tool for compound classification and QSAR modeling. *Journal of chemical information and computer sciences*, 43(6):1947–1958, **2003**.
- [58] J Ross Quinlan. Learning decision tree classifiers. *ACM Computing Surveys (CSUR)*, 28(1):71–72, **1996**.
- [59] Robert E Schapire. The boosting approach to machine learning: An overview. *Nonlinear estimation and classification*, pages 149–171, **2003**.
- [60] Alexey Natekin and Alois Knoll. Gradient boosting machines, a tutorial. *Frontiers in neurorobotics*, 7:21, **2013**.

- [61] Tianqi Chen, Tong He, Michael Benesty, Vadim Khotilovich, Yuan Tang, Hyunsu Cho, Kailong Chen, et al. Xgboost: extreme gradient boosting. *R package version 0.4-2*, 1(4):1–4, **2015**.
- [62] 3GPP TS 23.203 V17.2.0. Policy and charging control architecture. *3GPP*, **2021**.
- [63] 3GPP TS 22.186 V16.2.0. Enhancement of 3GPP support for V2X scenarios. *3GPP*, **2019**.
- [64] 3GPP TS 22.261 V18.5.0. Service requirements for the 5G system. *3GPP*, **2021**.
- [65] 5G global launches & statistics. **2020**.
- [66] Michel F Sanner et al. Python: a programming language for software integration and development. *J Mol Graph Model*, 17(1):57–61, **1999**.
- [67] Dajie Jiang and Guangyi Liu. An overview of 5G requirements. *5G Mobile Communications*, pages 3–26, **2017**.
- [68] Macrocell vs. small cell vs. femtocell: A 5G introduction. **2022**.
- [69] 3GPP TR 138 901 V16.1.0. Study on channel model for frequencies from 0.5 to 100 GHz. *ETSI*, **2020**.
- [70] Maad M Mijwel. Artificial neural networks advantages and disadvantages. Retrieved from LinkedIn [https://www.linkedin.com/pulse/artificial-neuralnet Work](https://www.linkedin.com/pulse/artificial-neuralnet-work), **2018**.
- [71] Gongde Guo, Hui Wang, David Bell, Yaxin Bi, and Kieran Greer. KNN model-based approach in classification. In *OTM Confederated International Conferences” On the Move to Meaningful Internet Systems”*, pages 986–996. Springer, **2003**.
- [72] Laura Auria and Rouslan A Moro. Support vector machines (SVM) as a technique for solvency analysis. **2008**.

- [73] Tianqi Chen, Tong He, Michael Benesty, Vadim Khotilovich, Yuan Tang, Hyunsu Cho, Kailong Chen, et al. Xgboost: extreme gradient boosting. *R package version 0.4-2*, 1(4):1–4, **2015**.
- [74] Jesse Davis and Mark Goadrich. The relationship between precision-recall and ROC curves. In *Proceedings of the 23rd international conference on Machine learning*, pages 233–240. **2006**.
- [75] Marina Sokolova, Nathalie Japkowicz, and Stan Szpakowicz. Beyond accuracy, F-score and ROC: a family of discriminant measures for performance evaluation. In *Australasian joint conference on artificial intelligence*, pages 1015–1021. Springer, **2006**.
- [76] Kenneth Leung. Micro, macro & weighted averages of F1 score, clearly explained. **2022**.
- [77] Zhe Hui Hoo, Jane Candlish, and Dawn Teare. What is a ROC curve?, **2017**.